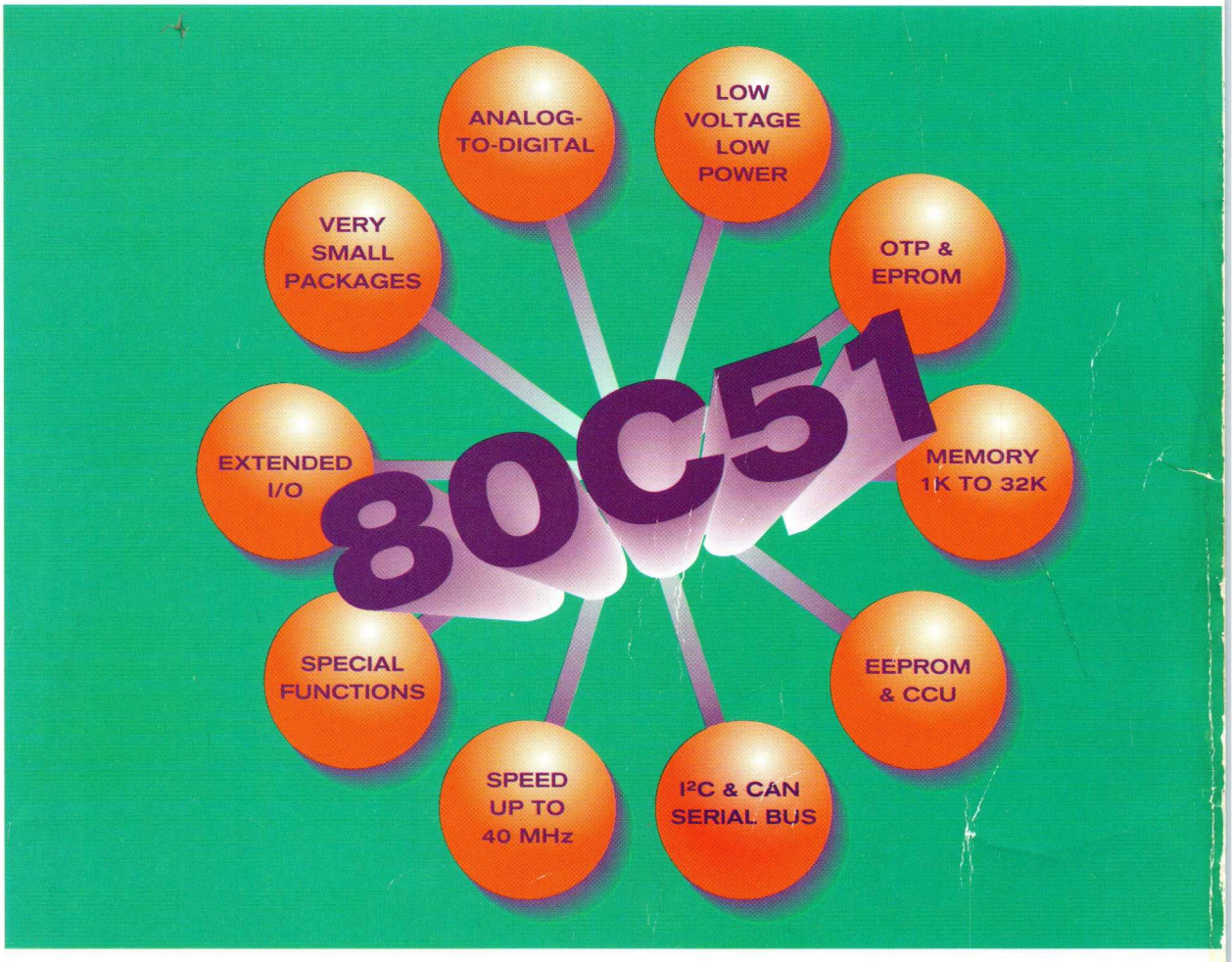
INTEGRATED CIRCUITS

# 80C51-Based 8-Bit Microcontrollers




1997

Data Handbook IC20

Philips Semiconductors

Let's make things better.

PHILIPS

## QUALITY ASSURED

Our quality system focuses on the continuing high quality of our components and the best possible service for our customers. We have a three-sided quality strategy: we apply a system of total quality control and assurance; we operate customer-oriented dynamic improvement programmes; and we promote a partnering relationship with our customers and suppliers.

## PRODUCT SAFETY

In striving for state-of-the-art perfection, we continuously improve components and processes with respect to environmental demands. Our components offer no hazard to the environment in normal use when operated or stored within the limits specified in the data sheet.

Some components unavoidably contain substances that, if exposed by accident or misuse, are potentially hazardous to health. Users of these components are informed of the danger by warning notices in the data sheets supporting the components. Where necessary the warning notices also indicate safety precautions to be taken and disposal instructions to be followed. Obviously users of these components, in general the set-making industry, assume responsibility towards the consumer with respect to safety matters and environmental demands.

All used or obsolete components should be disposed of according to the regulations applying at the disposal location. Depending on the location, electronic components are considered to be 'chemical', 'special' or sometimes 'industrial' waste. Disposal as domestic waste is usually not permitted.

# 80C51-Based 8-Bit Microcontrollers

## CONTENTS


| | | page |
|---|---|---|
| SECTION 1 | GENERAL INFORMATION | 1-1 |
| SECTION 2 | 80C51 TECHNICAL DESCRIPTION | 2-1 |
| SECTION 3 | 80C51 FAMILY DERIVATIVES | 3-1 |
| SECTION 4 | HIGH PERFORMANCE 16-BIT 80C51 XA (eXtended Architecture) | 4-1 |
| SECTION 5 | PACKAGE OUTLINES | 5-1 |
| SECTION 6 | DATA HANDBOOK SYSTEM | 6-1 |
| APPENDIX A | PIN CONFIGURATIONS | A-1 |

Philips Semiconductors and Philips Electronics North America Corporation reserve the right to make changes, without notice, in the products, including circuits, standard cells, and/or software, described or contained herein in order to improve design and/or performance. Philips Semiconductors assumes no responsibility or liability for the use of any of these products, conveys no license or title under any patent, copyright, or mask work right to these products, and makes no representations or warranties that these products are free from patent, copyright, or mask work right infringement, unless otherwise specified. Applications that are described herein for any of these products are for illustrative purposes only. Philips Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

LIFE SUPPORT APPLICATIONS

Philips Semiconductors and Philips Electronics North America Corporation Products are not designed for use in life support appliances, devices, or systems where malfunction of a Philips Semiconductors and Philips Electronics North America Corporation Product can reasonably be expected to result in a personal injury. Philips Semiconductors and Philips Electronics North America Corporation customers using or selling Philips Semiconductors and Philips Electronics North America Corporation Products for use in such applications do so at their own risk and agree to fully indemnify Philips Semiconductors and Philips Electronics North America Corporation for any damages resulting from such improper use or sale.

Philips Semiconductors and Philips Electronics North America Corporation register eligible circuits under the Semiconductor Chip Protection Act.

© Copyright Philips Electronics North America Corporation, 1996

All rights reserved.
Printed in U.S.A.

## PRODUCT STATUS

| DEFINITIONS | | |
|---|---|---|
| **DATA SHEET IDENTIFICATION** | **PRODUCT STATUS** | **DEFINITION** |
| ***Objective Specification*** | **Formative or in Design** | This data sheet contains the design target or goal specifications for product development. Specifications may change in any manner without notice. |
| ***Preliminary Specification*** | **Preproduction Product** | This data sheet contains preliminary data, and supplementary data will be published at a later date. Philips Semiconductors reserves the right to make changes at any time without notice in order to improve design and supply the best possible product. |
| ***Product Specification*** | **Full Production** | This data sheet contains Final Specifications. Philips Semiconductors reserves the right to make changes at any time without notice, in order to improve design and supply the best possible product. |




Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# Preface

**80C51-Based**
**8-Bit Microcontrollers**

## Microcontrollers from Philips Semiconductors

Philips Semiconductors 8 and 16-bit microcontrollers are based on the widely-accepted 8048, 8051 and XA architectures. We offer most of the 'industry standard' products in these architectures as well as a large selection of powerful derivative products. These derivatives offer a wide assortment of features, including: additional memory, A/D, PWM, additional timers, DTMF, OSD, OTP, EMC and EMI, plus many others. The variety of product derivatives allows Philips Semiconductors to support a broad range of functions in consumer, telecom, EDP, multi media, automotive and industrial applications.

For details, see:

- 8048 'industry standard' architecture types (PCF84CXXX family) in *"Data Handbook IC14"*.
  The PCD33XX family covers telecom terminal family devices based on the 8048 core and instruction set, in *"Data Handbook IC03"*.
- 8051 'industry standard' architecture types in *"Data Handbook IC20"*.
- XA types in *"Data Handbook IC25"*.

The Low Power 80CL51 family of derivatives can be found in *"Data Handbook IC20"*. These devices operate over the wide voltage range of 1.8 to 6.0V and are ideal for portable and battery operations.

Many of Philips Semiconductors ICs offer on-board UART serial ports and $I^2C$-bus. The $I^2C$-bus allows easy connection to over 100 other devices, thereby increasing system capabilities even further. For automotive and industrial applications, we also offer the CAN and the VAN serial bus. The CAN standard, developed by Bosch, and VAN concepts offer high noise immunity and error correction.

Philips Semiconductors 16-bit microcontroller family is based on the XA architecture. The XA is upwards compatible with the 80C51 and offers users an easy migration path to higher performance. While compatible with the 80C51, this compatibility has in no way limited the performance of the XA, which is one of the highest performance 16-bit microcontrollers available.

Philips Semiconductors is developing a family of 32-bit microcontrollers that will be based on the MIPS core. This family of microcontrollers will offer advanced performance for those applications that are computation and memory intensive in an embedded control environment.

# Section 1
## General Information

**80C51-Based**
**8-Bit Microcontrollers**

**CONTENTS**


Contents ..... 1-3
Quality ..... 1-9
80C51 microcontroller family features guide ..... 1-10
8051 microcontroller cross-reference guide ..... 1-14
Low power / low voltage microcontroller family ..... 1-15
CMOS and NMOS 8-bit microcontroller family ..... 1-16
CMOS 16-bit microcontroller family ..... 1-20
Ordering Information ..... 1-21
FAX-on-DEMAND System ..... 1-23
Microcontroller internet and bulletin board access ..... 1-24
80C51 microcontroller development system support ..... 1-26
8-bit microcontroller demonstration and evaluation boards ..... 1-28

# CONTENTS

## IC20: 80C51-BASED 8-BIT MICROCONTROLLERS

Preface ..... iii

**Section 1 – General Information**

Contents ..... 1-3
Quality ..... 1-9
80C51 microcontroller family features guide ..... 1-10
8051 microcontroller cross-reference guide ..... 1-14
Low power / low voltage microcontroller family ..... 1-15
CMOS and NMOS 8-bit microcontroller family ..... 1-16
CMOS 16-bit microcontroller family ..... 1-20
Ordering Information ..... 1-21
FAX-on-DEMAND System ..... 1-23
Microcontroller internet and bulletin board access ..... 1-24
80C51 microcontroller development system support ..... 1-26
8-bit microcontroller demonstration and evaluation boards ..... 1-28

**Section 2 – 80C51 Technical Description**

80C51 family architecture ..... 2-3
80C51 family hardware description ..... 2-18
80C51 family programmer's guide and instruction set ..... 2-43
80C51 family EPROM products ..... 2-98

**Section 3 – 80C51 Family Derivatives**

| | | |
|---|---|---|
| **80C31/80C51/87C51** | CMOS single-chip 8-bit microcontrollers | 3-3 |
| **83C51FA/83C51FB/ 83C51FC/80C51FA** | CMOS single-chip 8-bit microcontrollers | 3-25 |
| **87C51FA/87C51FB** | CMOS single-chip 8-bit microcontrollers | 3-61 |
| **87C51FC** | CMOS single-chip 8-bit microcontroller | 3-89 |
| 80CL31/80CL51 | Low-voltage single-chip 8-bit microcontrollers | 3-117 |
| **87L51FA/87L51FB** | CMOS single-chip 3.0V 8-bit microcontrollers | 3-150 |
| **80C32/87C52** | CMOS single-chip 8-bit microcontrollers | 3-166 |
| **80C52/80C54/80C58** | CMOS single-chip 8-bit microcontrollers | 3-188 |
| **87C54/87C58** | CMOS single-chip 8-bit microcontrollers | 3-215 |
| **83C145; 83C845 83C055; 87C055** | Microcontrollers for TV and video (MTV) | 3-236 |
| 80CL410/83CL410 | Low voltage/low power single-chip 8-bit microcontroller with $I^2C$ | 3-270 |
| **80C451/83C451/87C451** | CMOS single-chip 8-bit microcontrollers | 3-292 |
| **80C453/83C453/87C453** | CMOS single-chip 8-bit microcontrollers | 3-311 |
| **83C508/87C508** | CMOS single-chip 8-bit microcontrollers | 3-334 |
| **P83C524** | 8-bit microcontroller | 3-351 |
| 87C524 | CMOS single-chip 8-bit microcontroller | 3-372 |
| 80C528/83C528 | CMOS single-chip 8-bit microcontrollers | 3-392 |
| **87C528** | CMOS single-chip 8-bit microcontroller | 3-410 |
| **P8xCE528** | 8-bit microcontroller with EMC | 3-430 |
| 80C550/83C550/87C550 | CMOS single-chip 8-bit microcontroller with A/D and watchdog timer | 3-450 |
| **8XC552/562 overview** | | 3-473 |
| 8XC552 OVERVIEW | | 3-473 |
| 83C562 OVERVIEW | | 3-473 |
| Differences From the 80C51 | | 3-473 |
| Program Memory | | 3-473 |

# CONTENTS


Data Memory ... 3-473
Special Function Registers ... 3-473
Timer T2 ... 3-474
Timer T3, The Watchdog Timer ... 3-480
Serial I/O ... 3-481
Reset Circuitry ... 3-512
Interrupts ... 3-513
I/O Port Structure ... 3-517
Port 1 Operation ... 3-517
Port 5 Operation ... 3-517
Pulse Width Modulated Outputs ... 3-517
Analog-to-Digital Converter ... 3-517
Power Reduction Modes ... 3-523
Memory Organization ... 3-525

**80C552/83C552** Single-chip 8-bit microcontroller with 10-bit A/D, capture/compare timer, high-speed outputs, PWM . 3-530
87C552 Single-chip 8-bit microcontroller with 10-bit A/D, capture/compare timer, high-speed outputs, PWM . 3-550
**P83CE558/P80CE558/ P89CE558** Single-chip 8-bit microcontroller ... 3-571
**P83CE559/P80CE559** Single-chip 8-bit microcontroller ... 3-640
80C562/83C562 Single-chip 8-bit microcontroller with 8-bit A/D, capture/compare timer, high-speed outputs, PWM .. 3-708
**80C575/83C575/87C575** CMOS single-chip 8-bit microcontrollers ... 3-721
**83C576/87C576** CMOS single-chip 8-bit microcontrollers ... 3-756
**P80CL580; P83CL580** Low voltage 8-bit microcontrollers ... 3-798
**P8xC592** 8-bit microcontroller with on-chip CAN ... 3-869
**P8xCE598** 8-bit microcontroller with on-chip CAN ... 3-969
**80C652/83C652** CMOS single-chip 8-bit microcontrollers ... 3-992
**87C652** CMOS single-chip 8-bit microcontroller ... 3-1010
**83C654** CMOS single-chip 8-bit microcontroller ... 3-1030
**87C654** CMOS single-chip 8-bit microcontroller ... 3–1048
**83CE654** CMOS single-chip 8-bit microcontroller with Electromagnetic Compatibility improvements ... 3-1069
**83C748/87C748** CMOS single-chip 8-bit microcontrollers ... 3-1083
**83C749/87C749** CMOS single-chip 8-bit microcontrollers ... 3-1095
**TPM749** Microcontroller with TrackPoint™ microcode from IBM ... 3-1110
**83C750/87C750** CMOS single-chip 8-bit microcontrollers ... 3-1119
**83C751/87C751** CMOS single-chip 8-bit microcontrollers ... 3-1129
**83C752/87C752** CMOS single-chip 8-bit microcontroller with A/D, PWM ... 3-1147
**83C754/87C754** CMOS single-chip 8-bit microcontrollers ... 3-1165
**TPM754** Microcontroller with TrackPoint™ microcode from IBM ... 3-1187
**P83CL781; P83CL782** Low voltage 8-bit microcontrollers ... 3-1195
80C851/83C851 CMOS single-chip 8-bit microcontroller with on-chip EEPROM ... 3-1251

**Section 4 – High Performance 16-bit 80C51 XA (eXtended Architecture)**

80C51XA Architectural overview ... 4-3
**XA-G1** CMOS single-chip 16-bit microcontroller ... 4-13
**XA-G2** CMOS single-chip 16-bit microcontroller ... 4-14
**XA-G3** CMOS single-chip 16-bit microcontroller ... 4-15
**XA-C3** CMOS single-chip 16-bit microcontroller with CAN/DeviceNet controller ... 4-16
**XA-S3** Single-chip 16-bit microcontroller ... 4-17

# CONTENTS

**Section 5 – Package Outlines**

Soldering Package information ... 5-2

**Plastic Dual In-Line Package**

DIP8: plastic dual in-line package; 8 leads (300 mil) ... SOT97-1 ... 5-4
DIP24: plastic dual in-line package; 24 leads (300 mil) ... SOT222-1 ... 5-5
DIP28: plastic dual in-line package; 28 leads (600 mil); long body ... SOT117-2 ... 5-6
DIP40: plastic dual in-line package; 40 leads (600 mil) ... SOT129-1 ... 5-7

**Plastic Shrink Dual In-Line Package**

SDIP42: plastic shrink dual in-line package; 42 leads (600 mil) ... SOT270-1 ... 5-8
SDIP64: plastic shrink dual in-line package; 64 leads (750 mil) ... SOT274-1 ... 5-9

**Ceramic Dual In-Line Package**

24-Pin (300 mils wide) Ceramic Dual In-line (F) Package (with Window (FA) Package) ... 0586B ... 5-10
28-Pin (600 mils wide) Ceramic Dual In-line (F) Package (with Window (FA) Package) ... 0589B ... 5-11
40-Pin (600 mils wide) Ceramic Dual In-line (F) Package (with Window (FA) Package) ... 0590B ... 5-12

**Plastic Leaded Chip Carrier**

PLCC28: plastic leaded chip carrer; 28 leads; pedestal ... SOT261-3 ... 5-13
PLCC44: plastic leaded chip carrier; 44 leads ... SOT187-2 ... 5-14
PLCC68: plastic leaded chip carrier; 68 leads ... SOT188-2 ... 5-15
PLCC68: plastic leaded chip carrier; 68 leads; pedestal ... SOT188-3 ... 5-16

**Ceramic Leaded Chip Carrier**

68-Pin Chip Carrier, J-Bend (L) Package ... 1240C ... 5-17
Ceramic leaded chip carrier (window); 68 leads ... NO330 ... 5-18

**Plastic Quad Flat Package**

QFP44: plastic quad flat package; 44 leads (lead length 1.3 mm); body 10 x 10 x 1.75 mm ... SOT307-2 ... 5-19
QFP44: plastic quad flat package; 44 leads (lead length 2.35 mm); body 14 x 14 x 2.2 mm ... SOT205-1 ... 5-20
LQFP44: plastic low profile quad flat package; 44 leads; body 10 x 10 x 1.4 mm ... SOT389-1 ... 5-21
QFP64: plastic quad flat package; 64 leads (lead length 2.35 mm); body 14 x 20 x 2.75 mm ... SOT208-1 ... 5-22
QFP64: plastic quad flat package; 64 leads (lead length 1.95mm); body 14 x 20 x 2.8 mm ... SOT319-2 ... 5-23
QFP80: plastic quad flat package; 80 leads (lead length 1.6 mm); body 14 x 20 x 3.0 mm ... SOT310-1 ... 5-24
QFP80: plastic quad flat package; 80 leads (lead length 1.95 mm); body 14 x 20 x 2.7 mm; high stand-off height ... SOT318-1 ... 5-25
QFP80: plastic quad flat package; 80 leads (lead length 1.95 mm); body 14 x 20 x 2.8 mm ... SOT318-2 ... 5-26

**Ceramic Quad Flat Package**

44-pin CerQuad J-Bend (K) Package ... 1472A ... 5-27
68-pin CerQuad J-Bend (K) Package ... 1473A ... 5-28

**Plastic Small Outline Package**

SO8: plastic small outline package; 8 leads; body width 3.9mm ... SOT96-1 ... 5-29
SO28: plastic small outline package; 28 leads; body width 7.5mm ... SOT136-1 ... 5-30
VSO40: plastic very small outline package; 40 leads ... SOT158-1 ... 5-31
VSO56: plastic very small outline package; 56 leads ... SOT190-1 ... 5-32

**Plastic Shrink Small Outline Package**

SSOP24: plastic shrink small outline package; 24 leads; body width 5.3 mm ... SOT340-1 ... 5-33
SSOP28: plastic shrink small outline package; 28 leads; body width 5.3mm ... SOT341-1 ... 5-34

**Section 6 – Data Handbook System** ... 6-1

Data handbook system ... 6-2

**Appendix A – Pin Configurations** ... A-1

# CONTENTS

## APPLICATION NOTES AND DEVELOPMENT TOOLS FOR 80C51 MICROCONTROLLERS


Preface ..... iii

**Section 1 – General Information**

Contents ..... 1-3
Quality ..... 1-9
80C51 microcontroller family features guide ..... 1-10
8051 microcontroller cross-reference guide ..... 1-14
Low power / low voltage microcontroller family ..... 1-15
CMOS and NMOS 8-bit microcontroller family ..... 1-16
CMOS 16-bit microcontroller family ..... 1-20
Ordering Information ..... 1-21
FAX-on-DEMAND System ..... 1-23
Microcontroller internet and bulletin board access ..... 1-24
80C51 microcontroller development system support ..... 1-26
8-bit microcontroller demonstration and evaluation boards ..... 1-28

**Section 2 – 80C51 Support Chips**

**PCF1252-X family** Threshold detector and reset generator ..... 2-3

**Section 3 – Inter-Integrated Circuit ($I^2C$) Bus**

The $I^2C$-bus and how to use it ..... 3-3
$I^2C$ peripheral selection guide ..... 3-27
82B715 $I^2C$ bus extender ..... 3-29

**Section 4 – $I^2C$ Serial Bus Application Notes & Articles**

AN422 Using the 8XC751 microcontroller as an $I^2C$ bus master ..... 4-3
AN425 Interfacing the PCF8584 $I^2C$-bus controller to 80C51 family microcontrollers ..... 4-21
AN430 Using the 8XC751/752 in multimaster $I^2C$ applications ..... 4-40
AN433 $I^2C$ slave routines for the 83C751 ..... 4-76
AN434 Connecting a PC keyboard to the $I^2C$-bus ..... 4-82
AN438 $I^2C$ routines for 8XC528 ..... 4-100
AN444 Using the P82B715 $I^2C$ extender on long cables ..... 4-122
AN452 One mile long $I^2C$ communication using the P82B715 ..... 4-142
ETV/AN89004 PLM51 $I^2C$ software interface IIC51 (version 0.5) ..... 4-149
EIE/AN91007 $I^2C$ driver routines for 8XC751/2 microcontrollers ..... 4-158
Exploring $I^2C$ ..... 4-212
Programming the $I^2C$ interface ..... 4-215

**Section 5 – 87C750, 8XC751, 8XC752 Application Notes**

AN422 Using the 8XC751 microcontroller as an $I^2C$ bus master ..... See Section 4
AN423 Software driven serial communication routines for the 83C751 and 83C752 microcontrollers ..... 5-3
AN426 Controlling air core meters with the 87C751 and SA5775 ..... 5-8
AN427 Timer I for the 83/87C748/749 and the 83/87C751/752 (non-$I^2C$ applications) microcontrollers ..... 5-22
AN428 Using the ADC and PWM of the 83C752/87C752 ..... 5-28
AN429 Airflow measurement using the 83/87C752 and "C" ..... 5-35
AN430 Using the 8XC751/752 in multimaster $I^2C$ applications ..... See Section 4
AN433 $I^2C$ slave routines for the 83C751 ..... See Section 4
AN436 "Opti-Mizer" power management for notebook computers using the 8XC752 microcontroller ..... 5-54
AN439 87C751 fast NiCad charger ..... 5-64

# CONTENTS


| | | |
|---|---|---|
| AN442 | (BCM) 87C751 Specification for a bus-controlled monitor | 5-76 |
| AN446 | A software duplex UART for the 751/752 | 5-93 |
| AN453 | Using the 87C751 microcontroller to gang program PCF8582/PCF8581 EEPROMs | 5-101 |
| AN454 | Interfacing the 83C576/87C576 to the ISA bus | 5-120 |
| EIE/AN91007 | $I^2C$ driver routines for 8XC751/2 microcontrollers | See Section 4 |

**Section 6 – 80C51 Application Notes & Articles**

| | | |
|---|---|---|
| AN408 | 80C451 operation of port 6 | 6-3 |
| AN417 | 256k Centronics printer buffer using the 87C451 microcontroller | 6-14 |
| AN418 | Counter/timer 2 of the 83C552 microcontroller | 6-27 |
| AN420 | Using up to 5 external interrupts on 80C51 family microcontrollers | 6-34 |
| AN424 | 8051 family warm boot determinations | 6-36 |
| AN440 | RAM loader program for 80C51 family applications | 6-38 |
| AN443 | IEEE Micro Mouse using the 87C751 microcontroller | 6-47 |
| AN447 | Automatic baud rate detection for the 80C51 | 6-68 |
| AN448 | Determining baud rates for 8051 UARTs and other UART issues | 6-71 |
| AN449 | Measure resistance and capacitance without an A/D | 6-74 |
| AN455 | Incircuit programming of the Philips 87C576 microcontroller | 6-87 |
| AN456 | Using LC oscillator circuits with Philips microcontrollers | 6-92 |
| AN457 | 80C51 External Memory Interfacing | 6-94 |
| ESG89001 | Electro magnetic compatibility and printed circuit board (PCB) constraints | 6-112 |
| EIE/AN91001 | Workbench EMC evaluation method | 6-131 |
| EIE/AN91006 | A/D conversion with P83CL410 PCF1252-x | 6-148 |
| EIE/AN91009 | Driver for 8xC851 E2PROM | 6-164 |
| EIE/AN92001 | Low RF-emission applications with a P83CE654 microcontroller | 6-179 |
| EIE/AN93017 | Using the analog-to-digital converter of the 8XC552 microcontroller | 6-191 |
| Chips push CAN bus into embedded world | | 6-209 |
| Add Text Overlay to Any Video Display | | 6-211 |

**Section 7 – Control Area Network (CAN) Bus**

| | | |
|---|---|---|
| Control Area Network (CAN) overview | | 7-3 |
| **P82C150** | CAN Serial Linked I/O device (SLIO) with digital and analog port functions | 7-4 |
| 82C200 | Stand-alone CAN-controller | 7-33 |
| **PCA82C250** | CAN controller interface | 7-69 |

**Section 8 – Development Support Tools**

| | | |
|---|---|---|
| Development support tools | | 8-3 |
| Advin Systems, Inc.: PILOT-MVP Universal Programmer | | 8-16 |
| Aisys Ltd.: DriveWay™–51 Device Drivers Code Generation Tool | | 8-17 |
| Archimedes Software, Inc.: IDE-8051 Integrated Development Environment | | 8-18 |
| Ashling: CTS51 Microprocessor Development Systems for Philips Microcontrollers | | 8-19 |
| Avocet Systems, Inc.: AvCase51 Complete Embedded Software Development | | 8-24 |
| BSO/Tasking: The total development solution for the 8051 family | | 8-25 |
| Cactus Logic: Integrated Debugging System/LC (IDS/LC™) | | 8-34 |
| CEIBO: | DS-51 Microprocessor Development System | 8-35 |
| | EB-51 Emulation Board | 8-43 |
| | DB-51 Development Board | 8-47 |
| | DS-750 Microcontroller Development Tool | 8-49 |
| | DS-752 In Circuit Emulator | 8-51 |
| | MP-51 Programmer | 8-53 |
| ChipTools, Inc.: | ChipView®-51 High-Level/Low-Level Debugger | 8-55 |
| CMX Company: | CMX-RTX™, CMX-TINY™, CMX-TINY+™ Real-Time Multi-Tasking Operating System for Microprocessors and Microcomputers | 8-56 |

# CONTENTS


| | | |
|---|---|---|
| Emulation Technology, Inc.: | ET-iC2000 8051 Real-Timer In-Circuit OpenEmulator | 8-57 |
| | 8051 Emulator Adapters, Test Clips and Accessories | 8-58 |
| Franklin Software, Inc.: | 8051 Software Development Tool Suites | 8-59 |
| | ProView—8051 Turbo Tools | 8-61 |
| | DS51 Source Level Debugger/Simulator | 8-63 |
| | A51 Macro Assembler Kit | 8-64 |
| | C51 C Language Compiler Kit for the Professional User | 8-65 |
| | DK51—Complete 8051 Developers Kit | 8-66 |
| | PK51—The Professional 8051 Developers Kit | 8-67 |
| HiTech Equipment Corp: | DryICE 8051 Family In-Circuit Emulators | 8-68 |
| Hitex: | *teletest 51* In-Circuit Emulators | 8-69 |
| IAR Systems: | 8051 Embedded Workbench for Windows | 8-70 |
| | C-SPY 8051 for Windows | 8-71 |
| Lauterbach: | TRACE32 ICE-51 | 8-72 |
| Logical Systems: | 8051 Family Programming Adapters and Socket Adapters | 8-73 |
| MetaLink Corporation: | iceMASTER-PE™ 8051 Family In-Circuit Emulator | 8-74 |
| | iceMASTER™ 8051 In-Circuit Emulators | 8-79 |
| Microtek International, Inc.: | EasyPack 8051 Family In-Circuit Emulator | 8-92 |
| | EasyPack 8052F In-Circuit Emulator | 8-95 |
| Nohau Corporation: | EMUL51-PC | 8-97 |
| | EMUL51-PC – PC-based in-circuit emulator | 8-99 |
| Raisonance: | PCE-51 Real-Time In-Circuit Emulator | 8-110 |
| Ramtex A/S: | STIMGATE® Target Controller for ANSI-C | 8-111 |
| Signum Systems: | USP-51 In-Circuit Emulator | 8-112 |
| Systronix®, Inc.: | BCI51 Pro/Dallas Development Board | 8-113 |
| PDS51 | Development System for 8XC51 Microcontroller Derivatives | 8-114 |
| LCP Family of Programmers for 87C51 and Derivatives | | 8-118 |
| S87C00KSD | 8XC51 and $I^2C$ Bus Evaluation Board | 8-121 |
| OM4130 | CAN evaluation board with the 8XC552 and 82C200 CAN controller | 8-122 |
| OM4239 | CAN evaluation board with the 8XC592 microcontroller | 8-123 |
| OM4240 | Evaluation board for the 8XCE598 microcontroller | 8-124 |
| OM4272 | SLIO evaluation board with the 82C150 and 82C250 CAN ICs | 8-125 |
| OM4280 | P83C852 Smart Card crypto-controller demonstration kit | 8-126 |
| PEB552 | Evaluation board | 8-127 |

**Section 9 – Data Handbook System** 9-2

## TOTAL QUALITY MANAGEMENT

Philips Semiconductors is a Quality Company, renowned for the high quality of our products and service. We keep alive this tradition by constantly aiming towards one ultimate standard, that of zero defects. This aim is guided by our Total Quality Management (TQM) system, the basis of which is described in the following paragraphs.

### Quality assurance

Based on ISO 9000 standards, customer standards such as Ford TQE and IBM MDQ. Our factories are certified to ISO 9000 by external inspectorates.

### Partnerships with customers

PPM co-operations, design-in agreements, ship-to-stock, just-in-time and self-qualification programmes, and application support.

### Partnerships with suppliers

Ship-to-stock, statistical process control and ISO 9000 audits.

### Quality improvement programme

Continuous process and system improvement, design improvement, complete use of statistical process control, realization of our final objective of zero defects, and logistics improvement by ship-to-stock and just-in-time agreements.

## ADVANCED QUALITY PLANNING

During the design and development of new products and processes, quality is built-in by advanced quality planning. Through failure-mode-and-effect analysis the critical parameters are detected and measures taken to ensure good performance on these parameters. The capability of process steps is also planned in this phase.

## PRODUCT CONFORMANCE

The assurance of product conformance is an integral part of our quality assurance (QA) practice. This is achieved by:

- Incoming material management through partnerships with suppliers.
- In-line quality assurance to monitor process reproducibility during manufacture and initiate any necessary corrective action. Critical process steps are 100% under statistical process control.
- Acceptance tests on finished products to verify conformance with the device specification. The test results are used for quality feedback and corrective actions. The inspection and test requirements are detailed in the general quality specifications.
- Periodic inspections to monitor and measure the conformance of products.

## PRODUCT RELIABILITY

With the increasing complexity of Original Equipment Manufacturer (OEM) equipment, components reliability must be extremely high. Our research laboratories and development departments study the failure mechanisms of semiconductors. Their studies result in design rules and process optimization for the highest built-in product reliability. Highly accelerated tests are applied to the product reliability evaluation. Rejects from reliability tests and from customer complaints are submitted to failure analysis, to result in corrective action.

## CUSTOMER RESPONSES

Our quality improvement depends on joint action with our customer. We need our customer's inputs and we invite constructive comments on all aspects of our performance. Please contact our local sales representative.

## RECOGNITION

The high quality of our products and services is demonstrated by many Quality Awards granted by major customers and international organizations.

# 80C51 microcontroller family features guide

| Part Number (ROMless) | | Memory | | | Counter Timers | I/O Port | Serial Interfaces | External Interrupt | Comments/ Special Features |
|---|---|---|---|---|---|---|---|---|---|
| | | ROM | EPRM | RAM | | | | | |
| P | 83C750 | 1K | | 64 | 1 (16-bit) | 2-3/8 | - | 2 | 40 MHz, Lowest cost, SSOP |
| P | 87C750 | | 1K | 64 | 1 (16-bit) | 2-3/8 | - | 2 | 40 MHz, Lowest cost, SSOP |
| P | 83C748 | 2K | | 64 | 1 (16-bit) | 2-3/8 | - | 2 | 8XC751 w/o $I^2C$, SSOP |
| P | 87C748 | | 2K | 64 | 1 (16-bit) | 2-3/8 | - | 2 | 8XC751 w/o $I^2C$, SSOP |
| S | 83C751 | 2K | | 64 | 1 (16-bit) | 2-3/8 | $I^2C$ (bit) | 2 | 24-pin Skinny DIP, SSOP |
| S | 87C751 | | 2K | 64 | 1 (16-bit) | 2-3/8 | $I^2C$ (bit) | 2 | 24-pin Skinny DIP, SSOP |
| P | 83C749 | 2K | | 64 | 1 (16-bit) | 2-5/8 | - | 2 | 8XC752 w/o $I^2C$, SSOP |
| P | 87C749 | | 2K | 64 | 1 (16-bit) | 2-5/8 | - | 2 | 8XC752 w/o $I^2C$, SSOP |
| S | 83C752 | 2K | | 64 | 1 (16-bit) | 2-5/8 | $I^2C$ (bit) | 2 | 5 Channel 8-bit A/D, PWM Output, SSOP |
| S | 87C752 | | 2K | 64 | 1 (16-bit) | 2-5/8 | $I^2C$ (bit) | 2 | 5 Channel 8-bit A/D, PWM Output, SSOP |
| SC | 80C51 (80C31) | 4K | | 128 | 2 | 4 | UART | 2 | CMOS (Sunnyvale) |
| PCx | 80C51 (80C31) | 4K | | 128 | 2 | 4 | UART | 2 | CMOS (Hamburg) |
| SC | 87C51 | | 4K | 128 | 2 | 4 | UART | 2 | CMOS |
| P | 80CL51 (80CL31) | 4K | | 128 | 2 | 4 | UART | 10 | Low Voltage (1.8V to 6V), Low Power |
| P | 83CL410 (80CL410) | 4K | | 128 | 2 | 4 | $I^2C$ | 10 | Low Voltage (1.8V to 6V), Low Power |
| SC | 83C451 (80C451) | 4K | | 128 | 2 | 7 | UART | 2 | Extended I/O, Processor Bus Interface |
| SC | 87C451 | | 4K | 128 | 2 | 7 | UART | 2 | Extended I/O, Processor Bus Interface |
| P | 83C550 (80C550) | 4K | | 128 | 2 + Watchdog | 4 | UART | 2 | 8 Channel 8-bit A/D |
| P | 87C550 | | 4K | 128 | 2 + Watchdog | 4 | UART | 2 | 8 Channel 8-bit A/D |
| P | 83C851 (80C851) | 4K | | 128 | 2 | 4 | UART | 2 | 256B EEPROM, 80C51 Pin compatible |
| P | 83C852 | 6K | | 256 | 2 (16-bit) | 2/8 | - | 1 | Smartcard Controller with 2K EEPROM (Data, Code) Cryptographic Calc Unit |
| P | 83CL580 (80CL580) | 6K | | 256 | 3 + Watchdog | 5 | UART, $I^2C$ | 9 | 4 Channel 8-bit A/D, PWM Output, Low Voltage (2.5V to 6V), Low Power |
| P | 80C52 (80C32) | 8K | | 256 | 3 | 4 | UART | 2 | 80C51 Pin Compatible |
| P | 87C52 | | 8K | 256 | 3 | 4 | UART | 2 | (see above) |
| P | 83C652 (80C652) | 8K | | 256 | 2 | 4 | UART, $I^2C$ | 2 | 80C51 Pin Compatible |
| S | 87C652 | | 8K | 256 | 2 | 4 | UART, $I^2C$ | 2 | (see above) |
| P | 83C453 (80C453) | 8K | | 256 | 2 | 7 | UART | 2 | Extended I/O, Processor Bus Interface |
| P | 87C453 | | 8K | 256 | 2 | 7 | UART | 2 | Extended I/O, Processor Bus Interface |
| S | 83C51FA (80C51FA) | 8K | | 256 | 3 + PCA | 4 | UART | 2 | Enhanced UART, 3 timers + PCA |
| S | 87C51FA | | 8K | 256 | 3 + PCA | 4 | UART | 2 | Enhanced UART, 3 timers + PCA |
| S | 83L51FA | 8K | | 256 | 3 + PCA | 4 | UART | 2 | Low Voltage 83C51FA (3V @ 20MHz) |
| S | 87L51FA | | 8K | 256 | 3 + PCA | 4 | UART | 2 | Low Voltage OTP 87C51FA (3V @ 20MHz) |
| P | 83C575 (80C575) | 8K | | 256 | 3 + PCA+ Watchdog | 4 | UART | 2 | High Reliability, with Low Voltage Detect, OSC Fail Detect, Analog Comparators, PCA |
| P | 87C575 | | 8K | 256 | (see above) | 4 | UART | 2 | (see above) |
| P | 83C576 (80C576) | 8K | | 256 | 3 + PCA+ Watchdog | 4 | UART | 2 | Same as 8XC575 plus UPI and 10-bit A/D |
| P | 87C576 | | 8K | 256 | (see above) | 4 | UART | 2 | (see above) |
| PC | 83C562 (80C562) | 8K | | 256 | 3 + Watchdog | 6 | UART | 2 | 8 Channel 8-bit A/D, 2 PWM Outputs, Capture/Compare Timer |
| PCx | 83C552 (80C552) | 8K | | 256 | 3 + Watchdog | 6 | UART, $I^2C$ | 2 | 8 Channel 10-bit A/D, 2 PWM Outputs, Capture/Compare Timer |
| S | 87C552 | | 8K | 256 | 3 + Watchdog | 6 | UART, $I^2C$ | 2 | (see above) |

Notes: Part number prefixes are noted in the first column.
All combinations of part type, speed, temperature and package may not be available.

# 80C51 microcontroller family features guide

| Part Number (ROMless) | | Program Security? | Clock Freq (MHz) | Temperature Ranges (°C) | | | Package | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 0 to 70 | –40 to +85 | –55 to +125 | PDIP | CDIP | PLCC | CLCC | PQFP/SSOP |
| 83C750 | S | N | 3.5 to 40 | X | X | | N24 | F24 | A28 | | DB24 (0–70F) |
| 87C750 | S | Y | 3.5 to 40 | X | X | | N24 | F24 | A28 | | DB24 (0–70F) |
| 83C748 | S | N | 3.5 to16 | X | X | | N24 | | A28 | | DB24 (0–70F) |
| 87C748 | S | Y | 3.5 to16 | X | X | | N24 | F24 | A28 | | DB24 (0–70F) |
| 83C751 | S | N | 3.5 to16 | X | X | | N24 | | A28 | | DB24 (0–70F) |
| 87C751 | S | Y | 3.5 to16 | X | X | | N24 | F24 | A28 | | DB24 (0–70F) |
| 83C749 | S | N | 3.5 to 16 | X | X | | N28 | | A28 | | DB28 (0–70F) |
| 87C749 | S | Y | 3.5 to 16 | X | X | | N28 | F28 | A28 | | DB28 (0–70F) |
| 83C752 | S | N | 3.5 to 16 | X | X | X | N28 | | A28 | | DB28 (0–70F) |
| 87C752 | S | Y | 3.5 to 16 | X | X | X | N28 | F28 | A28 | | DB28 (0–70F) |
| SC80C51 (80C31) | S | Y | 3.5 to 33 | X | X | X | N40 | | A44 | | B44 (5) |
| PCx80C51 (80C31) | H | N | 1.2 to 30 | X | X | X | P (40) | | WP (44) | | H (44) |
| 87C51 | S | Y | 3.5 to 33 | X | X | X | N40 | F40 | A44 | K44 | B44 (5) |
| 80CL51 (80CL31) | Z | N | 0 to 16 (1) | | X | | N40 (2) | | | | B44 |
| 83CL410(80CL410) | Z | N | 0 to 12 (1) | | X | | N40 (2) | | | | B44 |
| 83C451 (80C451) | S | N | 3.5 to 16 | X | X | X | N64 (4) | | A68 | | |
| 87C451 | S | Y | 3.5 to16 | X | X | X | N64 (4) | | A68 | | |
| 83C550 (80C550) | S | Y | 3.5 to 16 | X | X | | N40 | | A44 | | |
| 87C550 | S | Y | 3.5 to 16 | X | X | –40 to +125 | N40 | F40 | A44 | K44 | |
| 83C851 (80C851) | H | Y | 1.2 to 16 | X | X | | N40 | | A44 | | B44 |
| 83C852 | H | Y | 1 to 12 | X | | | SO28 or die | | | | |
| 83CL580 (80CL580) | Z | N | 0 to 12 (1) | | X | | (3) | | | | B64 |
| 80C52 (80C32) | S | Y | 3.5 to 24 | X | X | | N40 | | A44 | | B44 (5) |
| 87C52 | S | Y | 3.5 to 24 | X | X | X | N40 | F40 | A44 | K44 | B44 (5) |
| 83C652 (80C652) | H | Y | 1.2 to 24 | X | X | –40 to +125 | N40 | | A44 | | B44 |
| 87C652 | S | Y | 1.2 to 20 | X | X | X | N40 | F40 | A44 | K44 | |
| 83C453 (80C453) | S | N | 3.5 to 16 | X | X | | | | A68 | | |
| 87C453 | S | Y | 3.5 to16 | X | X | | | | A68 | | |
| 83C51FA (80C51FA) | S | Y | 3.5 to 24 | X | X | | N40 | | A44 | | B44 |
| 87C51FA | S | Y | 3.5 to 24 | X | X | | N40 | F40 | A44 | K44 | B44 |
| 83L51FA | S | Y | 3.5 to 20 | X | X | | N40 | | A44 | | B44 |
| 87L51FA | S | Y | 3.5 to 20 | X | X | | N40 | F40 | A44 | K44 | B44 |
| 83C575 (80C575) | S | Y | 4 to 16 | X | | X | N40 | | A44 | | B44 |
| 87C575 | S | Y | 4 to 16 | X | | X | N40 | F40 | A44 | K44 | B44 |
| 83C576 (80C576) | S | Y | 4 to 16 | X | | X | N40 | | A44 | | B44 |
| 87C576 | S | Y | 4 to 16 | X | | X | N40 | F40 | A44 | K44 | B44 |
| 83C562 (80C562) | H | N | 1.2 to 16 | X | X | –40 to +125 | | | A68 | | B80 |
| 83C552 (80C552) | H | N | 1.2 to 30 | X | X | –40 to +125 | | | A68 | | B80 |
| 87C552 | S | Y | 1.2 to 16 | X | | | | | A68 | K68 | |

Notes: Production Centers are indicated in the second column: H – Hamburg, S – Sunnyvale, Z – Zurich.
All combinations of part type, speed, temperature and package may not be available.
1) Oscillator options start from 32kHz.
2) Also available in VSO40 package.
3) Also available in VSO56 Package.
4) Not recommended for new design.
5) Package available up to 16 MHz only.

# 80C51 microcontroller family features guide

| Part Number (ROMless) | | Memory | | | Counter Timers | I/O Port | Serial Interfaces | External Interrupt | Comments/ Special Features |
|---|---|---|---|---|---|---|---|---|---|
| | | ROM | EPRM | RAM | | | | | |
| P | 83C055 | 16K | | 256 | 2 (16-bit) | 3 1/2 | – | 2 | On-Screen Display, 9 PWM Outputs, 3 Software A/D Inputs |
| P | 87C055 | | 16K | 256 | 2 (16-bit) | 3 1/2 | – | 2 | (see above) |
| P | 80C54 | 16K | | 256 | 3 | 4 | UART | 2 | Standard; 80C51 compatible |
| P | 87C54 | | 16K | 256 | 3 | 4 | UART | 2 | Standard; 87C51 compatible |
| P | 83C654 | 16K | | 256 | 2 | 4 | UART, I²C | 2 | 80C51 Pin Compatible |
| S | 87C654 | | 16K | 256 | 2 | 4 | UART, I²C | 2 | (see above) |
| P | 83CE654 | 16K | | 256 | 2 | 4 | UART, I²C | 2 | 83C654 with Reduced EMI |
| P | 83CL781 | 16K | | 256 | 3 | 4 | UART, I²C | 10 | Low Voltage (1.8V to 6V), Low Power |
| P | 83CL782 | 16K | | 256 | 3 | 4 | UART, I²C | 10 | 83CL781 Optimized 12MHz @ 3.1V |
| S | 83C51FB | 16K | | 256 | 3 + PCA | 4 | UART | 2 | Enhanced UART, 3 timers + PCA |
| S | 87C51FB | | 16K | 256 | 3 + PCA | 4 | UART | 2 | Enhanced UART, 3 timers + PCA |
| S | 83L51FB | 16K | | 256 | 3 + PCA | 4 | UART | 2 | Low Voltage 83C51FB (3V @ 20MHz) |
| S | 87L51FB | | 16K | 256 | 3 + PCA | 4 | UART | 2 | Low Voltage OTP 87C51FB (3V @ 20MHz) |
| P | 83C524 | 16K | | 512 | 3 + Watchdog | 4 | UART, I²C-bit | 2 | 512 RAM |
| P | 87C524 | | 16K | 512 | 3 + Watchdog | 4 | UART, I²C-bit | 2 | 512 RAM |
| P | 83C592 (80C592) | 16K | | 512 | 3 + Watchdog | 6 | UART, CAN | 6 | CAN Bus Controller with 8 x 10-bit A/D, 2 PWM outputs, Capture/Compare Timer |
| P | 87C592 | | 16K | 512 | 3 + Watchdog | 6 | UART, CAN | 6 | (see above) |
| P | 80C58 | 32K | | 256 | 3 | 4 | UART | 2 | Standard; 80C51 compatible |
| P | 87C58 | | 32K | 256 | 3 | 4 | UART | 2 | Standard; 87C51 compatible |
| S | 83C51FC | 32K | | 256 | 3 + PCA | 4 | UART | 2 | Enhanced UART, 3 timers + PCA |
| S | 87C51FC | | 32K | 256 | 3 + PCA | 4 | UART | 2 | Enhanced UART, 3 timers + PCA |
| P | 83C528 (80C528) | 32K | | 512 | 3 + Watchdog | 4 | UART, I²C-bit | 2 | Large Memory for High Level Languages |
| P | 87C528 | | 32K | 512 | 3 + Watchdog | 4 | UART, I²C-bit | 2 | Large Memory for High Level Languages |
| P | 83CE528 (80CE528) | 32K | | 512 | 3 + Watchdog | 4 | UART, I²C-bit | 2 | 8XC528 with Reduced EMI |
| P | 83CE598 (80CE598) | 32K | | 512 | 3 + Watchdog | 6 | UART, CAN | 6 | CAN Bus Controller, 8 x 10-bit A/D, 2 PWM outputs, WD, T2, Reduced EMI |
| P | 87CE598 | | 32K | 512 | 3 + Watchdog | 6 | UART, CAN | 6 | (see above) |
| P | 83CE558(80CE558) | 32K | | 1024 | 3 + Watchdog | 6 | UART, I²C | 2 | Low EMI, 8 Channel 10-bit A/D, 2 PWM Outputs, Capture/Compare Timer |
| P | 89CE558 | | 32K | 1024 | 3 + Watchdog | 6 | UART, I²C | 2 | 32K FLash EEPROM plus above |

Notes: Part number prefixes are noted in the first column.
All combinations of part type, speed, temperature and package may not be available.

# 80C51 microcontroller family features guide

| Part Number (ROMless) | | Program Security? | Clock Freq (MHz) | Temperature Ranges (°C) | | | Package | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 0 to 70 | –40 to +85 | –55 to +125 | PDIP | CDIP | PLCC | CLCC | PQFP/SSOP |
| 83C055 | S | N | 3.5 to 20 | X | | | NB42 | | | | |
| 87C055 | S | N | 3.5 to 20 | X | | | NB42 | | | | |
| 80C54 | S | Y | 3.5 to 24 | X | X | | N40 | | A44 | | B44 |
| 87C54 | S | Y | 3.5 to 24 | X | X | | N40 | F40 | A44 | K44 | B44 |
| 83C654 (80C654) | H | Y | 1.2 to 24 | X | X | –40 to +125 | R42, N40 | | A44 | | B44 |
| 87C654 | S | Y | 1.2 to 20 | X | X | X | N40 | F40 | A44 | K44 | B44 |
| 83CE654 | H | Y | 1.2 to 16 | X | X | | | | | | B44 |
| 83CL781 | Z | N | 0 to 12 (1) | | X | | N40 | | | | B44 |
| 83CL782 | Z | N | 0 to 12 (1) | | –25 to +55 | | N40 | | | | B44 |
| 83C51FB | S | Y | 3.5 to 24 | X | X | | N40 | | A44 | | B44 |
| 87C51FB | S | Y | 3.5 to 24 | X | X | | N40 | F40 | A44 | K44 | B44 |
| 83L51FB | S | Y | 3.5 to 20 | X | | | N40 | | A44 | | B44 |
| 87L51FB | S | Y | 3.5 to 20 | X | | | N40 | F40 | A44 | K44 | B44 |
| 83C524 | H | Y | 1.2 to 16 | X | X | | N40 | | A44 | | B44 |
| 87C524 | S | Y | 3.5 to 20 | X | X | | N40 | F40 | A44 | K44 | B44 |
| 83C592 (80C592) | H | Y | 1.2 to 16 | | X | –40 to +125 | | | A68 | K68 | |
| 87C592 | H | Y | 1.2 to 16 | X | | | R42 | | A68 | K68 | |
| 80C58 | S | Y | 3.5 to 16 | X | X | | N40 | | A44 | | B44 |
| 87C58 | S | Y | 3.5 to 16 | X | X | | N40 | F40 | A44 | K44 | B44 |
| 83C51FC | S | Y | 3.5 to 24 | X | X | | N40 | | A44 | | B44 |
| 87C51FC | S | Y | 3.5 to 24 | X | X | | N40 | F40 | A44 | K44 | B44 |
| 83C528 (80C528) | H | Y | 1.2 to 16 | X | X | –40 to +125 | N40 | | A44 | | B44 |
| 87C528 | S | Y | 3.5 to 20 | X | X | | N40 | F40 | A44 | K44 | B44 |
| 83CE528 (80CE528) | H | Y | 1.2 to 16 | X | X | –40 to +125 | | | A44 | | B44 |
| 83CE598 (80CE598) | H | Y | 1.2 to 16 | | X | –40 to +125 | | | | | B80 |
| 87CE598 | H | Y | 3.5 to 16 | X | X | | | | | | B80 |
| 83CE558 80CE558 | H | Y | 1.2 to 16 | X | X | –40 to +125 | | | | | B80 |
| 89CE558 | H | Y | 1.2 to 16 | X | X | | | | | Q80 | B80 |

Notes: Production Centers are indicated in the second column: H – Hamburg, S – Sunnyvale, Z – Zurich.
All combinations of part type, speed, temperature and package may not be available.
1) Oscillator options start from 32kHz.
2) Also available in VSO40 package.
3) Also available in VSO56 Package.
4) Not recommended for new design.
5) Package available up to 16 MHz only.

# 8051 microcontroller cross-reference guide

| | INTEL | SIEMENS | OKI | MATRA/HARRIS | PHILIPS SEMICONDUCTORS |
|---|---|---|---|---|---|
| CMOS | 80C31BH | SAB 80C31 | MSM80C31 | 80C31 | PCB80C31BH-2/SC80C31BCC |
| | 80C31BH-1 | | | 80C31-1 | PCB80C31BH-3/SC80C31BCG |
| | 80C31BH-2 | | MSM80C31 | 80C3151 | /SC80C31BCB |
| | 80C51BH | SAB 80C51 | MSM80C51 | 80C51 | PCB80C51BH-2/SC80C51BCC |
| | 80C51BH-1 | | | 80C51-1 | PCB80C51BH-3/SC80C51BCG |
| | 80C51BH-2 | | MSM80C51 | 80C51 | /SC80C51BCB |
| | 87C51 | | | | SC87C51CC |
| | 87C51-1 | | | | SC87C51CG |
| | 87C51-2 | | | | SC87C51CB |
| | 80C32 | SAB80C32 | | | P80C32EB |
| | 80C32-1 | | | 80C32-25 | P80C32GB |
| | 80C52 | SAB80C52 | | | P80C52EB |
| | 80C52-1 | | | 80C52-25 | P80C52GB |
| | 80C54 | | | | 80C54 |
| | 83C54 | | | | 83C54 |
| | 87C54 | | | | 87C54 |
| | 80C58 | | | | 80C58 |
| | 83C58 | | | | 83C58 |
| | 87C58 | | | | 87C58 |
| CMOS | 83C51FA | | | | S83C51FA |
| | 87C51FA | | | | S87C51FA |
| | 83C51FB | | | | S83C51FB |
| | 87C51FB | | | | S87C51FB |
| | 83C51FC | | | | S83C51FC |
| | 87C51FC | | | | S87C51FC |

**NOTES:**
1. 80XXAHL = 80XX with low power standby pin; H = HMOS.

# Low power / low voltage microcontroller family

## 80C51 LOW POWER FAMILY

| Type | Available | ROM | RAM | I/O | I²C | UART | Features | Package |
|---|---|---|---|---|---|---|---|---|
| 80CL51 | Yes | 4k | 128 | 32 | No | Yes | Low Voltage 80C51 | 40-Pin Dual In-Line<br>40-Pin Very Small Outline<br>44-Pin Quad Flat Pack |
| 80CL31 | Yes | – | 128 | 32 | No | Yes | Low Voltage 80C31 | 40-Pin Dual In-Line<br>40-Pin Very Small Outline<br>44-Pin Quad Flat Pack |
| 83CL410 | Yes | 4k | 128 | 32 | Yes | No | 80CL51 with I²C-bus | 40-Pin Dual In-Line<br>40-Pin Very Small Outline<br>44-Pin Quad Flat Pack |
| 80CL410 | Yes | – | 128 | 32 | Yes | No | 80CL51 with I²C-bus | 40-Pin Dual In-Line<br>40-Pin Very Small Outline<br>44-Pin Quad Flat Pack |
| 83CL580 | Yes | 6k | 256 | 40 | Yes | Yes | ADC, PWM, Watchdog, T2 | 50–Pin Very Small Outline<br>64-Pin Quad Flat Pack |
| 80CL580 | Yes | – | 256 | 40 | Yes | Yes | ADC, PWM, Watchdog, T2 | 50–Pin Very Small Outline<br>64-Pin Quad Flat Pack |
| 83CL781 | Yes | 16k | 256 | 32 | Yes | Yes | Low voltage 83C654, T2 | 40-Pin Dual In-Line<br>44-Pin Quad Flat Pack |
| 83CL782 | Yes | 16k | 256 | 32 | Yes | Yes | Fast 83CL781: 12MHz/3V | 40-Pin Dual In-Line<br>44-Pin Quad Flat Pack |
| 85CL000 | Yes | – | 256 | 32 | Yes | Yes | For SW development | Piggyback |
| 85CL580 | Yes | – | 256 | 40 | Yes | Yes | For SW development | Piggyback |
| 85CL782 | Yes | – | 256 | 32 | Yes | Yes | For SW development | Piggyback |

## LOW VOLTAGE DEVICES

| Type | Available | ROM | RAM | I/O | I²C | UART | Features | Package |
|---|---|---|---|---|---|---|---|---|
| 83L51FA | Yes | 8k | 256 | 32 | No | Yes | PCA, Enhanced UART<br>3.0V to 4.5V | 40-Pin Dual In-Line<br>44-Pin PLCC<br>44-Pin Quad Flat Pack |
| 87L51FA | Yes | 8k<br>EPROM/<br>OTP | 256 | 32 | No | Yes | PCA, Enhanced UART<br>3.0V to 4.5V | 40-Pin Dual In-Line<br>44-Pin PLCC<br>44-Pin Quad Flat Pack |
| 83L51FB | Yes | 16k | 256 | 32 | No | Yes | PCA, Enhanced UART<br>3.0V to 4.5V | 40-Pin Dual In-Line<br>44-Pin PLCC<br>44-Pin Quad Flat Pack |
| 83L51FB | Yes | 16k<br>EPROM/<br>OTP | 256 | 32 | No | Yes | PCA, Enhanced UART<br>3.0V to 4.5V | 40-Pin Dual In-Line<br>44-Pin PLCC<br>44-Pin Quad Flat Pack |

# CMOS and NMOS 8-bit microcontroller family

## 8400 FAMILY CMOS

| TYPE | ROM | RAM | SPEED (MHz) | PACKAGE | FUNCTIONS | REMARKS | PROBE SDS | REMARKS |
|---|---|---|---|---|---|---|---|---|
| 84C21A<br>84C41A<br>84C81A | 2k<br>4k<br>8k | 64<br>128<br>256 | 10<br>10<br>10 | DIL28/SO28<br>DIL28/SO28<br>DIL28/SO28 | 20 I/O lines<br>8-bit timer<br>Byte $I^2C$ | | OM1083 | OM1025<br>(LSDS) |
| 84C22A<br>84C42A<br>84C12A | 2k<br>4k<br>1k | 64<br>64<br>64 | 10<br>10<br>16 | DIL20/SO20<br>DIL20/SO20<br>DIL20/SO20<br>DIL20/SO20 | 13 I/O lines<br>8-bit timer | | OM1083 +<br>Adapter_1 | OM1025<br>(LSDS) |
| 84C00B<br><br><br>84C00T | 0<br><br><br>0 | 256<br><br><br>256 | 10<br><br><br>10 | 28 pins<br><br><br>VSO-56 | 20 I/O lines<br>8-bit timer<br>Byte $I^2C$ | Piggyback<br><br><br>ROMless | OM1080<br><br><br>OM1080 | |
| 84C121<br><br><br>84C121B | 1k<br><br><br>0 | 64<br><br><br>64 | 10<br><br><br>10 | DIL20/SO20 | 13 I/O lines<br>2 8-bit timers<br>8 bytes<br>EEPROM | <br><br><br>Piggyback | OM1073 | OM1025(LEDS)<br><br><br>OM1027 |
| 84C122A<br>84C122B<br>84C422A<br>84C422B<br>84C822A<br>84C822B<br>84C822C | 1k<br><br>4K<br><br>8K | 32<br><br>32<br><br>32 | 10 | A: SO20<br>B: SO24<br>C: SO28 | Controller for<br>remote control<br>A: 12 I/O<br>B: 16 I/O<br>C: 20 I/O | | OM4830 | |
| 84C230 | 2I | 64 | 10 | DIL40/VSO40 | 12 I/O lines<br>8-bit timer<br>16*4 LCD drive | | OM1072 | |
| 84C430<br><br><br>84C430BH | 4k<br><br><br>0 | 128<br><br><br>128 | 10<br><br><br>10 | QFP64 | 24 I/O lines<br>8-bit timer<br>Byte $I^2C$<br>24*4 LCD drive | <br><br><br>Piggyback for C230<br>and C430 | OM1072 | |
| 84C633<br><br><br><br><br>84C633B | 6k<br><br><br><br><br>0 | 256<br><br><br><br><br>256 | 16<br><br><br><br><br>16 | VSO56 | 28 I/O lines<br>8-bit timer<br>16-bit up/down<br>counter<br>16-bit timer<br>with compare<br>and capture<br>16*4 LCD drive | | OM1086 | |
| 84C440<br>84C441<br>84C443<br>84C444<br>84C640<br>84C641<br>84C643<br>84C644<br>84C840<br>84C841<br>84C843<br>84C844 | 4k<br>4k<br>4k<br>4k<br>6k<br>6k<br>6k<br>6k<br>8k<br>8k<br>8k<br>8k | 128<br>128<br>128<br>128<br>128<br>128<br>128<br>128<br>192<br>192<br>192<br>192 | 10<br>10<br>10<br>10<br>10<br>10<br>10<br>10<br>10<br>10<br>10<br>10 | DIP42 shrunk | RC: 29 I/O lines<br>LC: 28 I/O lines<br>8-bit timer<br>1 14-bit PWM<br>5 6-bit PWM<br>3-bit ADC<br>OSD 2L-16 | $I^2C$, RC<br>$I^2C$, LC<br>RC<br>LC<br>$I^2C$, RC<br>$I^2C$, LC<br>RC<br>LC<br>$I^2C$, RC<br>$I^2C$, LC<br>RC<br>LC | OM1074 | For emulation of<br>LC versions,<br>use OM1074 +<br>adapter_3 +<br>2 adapter_5<br><br><br><br><br>Baud for LCDS<br>OM4831 |

# CMOS and NMOS 8-bit microcontroller family

## 8400 FAMILY CMOS (Continued)

| TYPE | ROM | RAM | SPEED (MHz) | PACKAGE | FUNCTIONS | REMARKS | PROBE SDS | REMARKS |
|---|---|---|---|---|---|---|---|---|
| 84C646<br>84C846 | 6k<br>8k | 192<br>192 | 10<br>10 | DIP42 shrunk | 30 I/O lines<br>DOS clock = PLL<br>8 bit timer<br>1-14 bit PWM<br>4-6 bit PWM<br>4-7 bit PWM<br>3-4 bit ADC<br>DOS: 64 disp. RAM<br>62 char. fonts<br>Char. blinking<br>Shadow modes<br>8 foreground colors/char.<br>8 background colors/word<br>DOS: clock: 8 . . 20MHz | $I^2C$, RC<br>$I^2C$, RC | OM4829 +<br>OM4832 | OM4833 for LCD584 |
| 84C85<br><br>84C85B | 8k<br><br>0 | 256<br><br>256 | 10<br><br>10 | DIL40/VSO40 | 32 I/O lines<br>8-bit timer<br>Byte $I^2C$ | <br><br>Piggyback for C85 | OM1070 | |
| 84C853<br><br><br><br>84C853B | 8k<br><br><br><br>0 | 256<br><br><br><br>256 | 16<br><br><br><br>16 | DIL40/VSO40 | 33 I/O lines<br>8-bit timer<br>16-bit up/down counter<br>16-bit timer with compare and capture | <br><br><br><br>Piggyback for C853 | OM1081 | |
| 84C270<br>84C470<br><br>84C270B<br><br>84C470B | 2k<br>4k<br><br>0<br><br>0 | 128<br>128<br><br>128<br><br>128 | 10<br>10<br><br>10<br><br>10 | DIL40/VSO40<br>DIL40/VSO40 | 8 I/O lines<br>16*8 capture keyboard matrix<br>8-bit timer<br><br>470 also handles mech. keys | <br><br><br>Piggyback for C270<br><br>Piggyback for C470 | OM1077 | |
| 84C271 | 2k | 128 | 10 | DIL40 | 8 I/O lines<br>16*8 mech. keyboard matrix<br>8-bit timer | | OM1078 | |

## 8400 FAMILY NMOS

| TYPE | ROM | RAM | SPEED (MHz) | PACKAGE | FUNCTIONS | REMARKS | EMULATOR TOOLS | REMARKS |
|---|---|---|---|---|---|---|---|---|
| 8411<br>8421<br>8441<br>8461 | 1k<br>2k<br>4k<br>6k | 64<br>64<br>128<br>128 | 6<br>6<br>6<br>6 | DIL28/SO28<br>DIL28/SO28<br>DIL28/SO28<br>DIL28/SO28 | 20 I/O lines<br>8-bit timer<br>Byte $I^2C$ | | | OM1025 (LCDS) + OM1026 |
| 8422<br>8442 | 2k<br>4k | 64<br>128 | 6<br>6 | DIL20<br>DIL20 | 13 I/O lines<br>8-bit timer<br>Bit $I^2C$ | | | |
| 8401B | 0 | 128 | 6 | 28-pin | | Piggyback for 84X1 | | |

## 3300 FAMILY CMOS

| TYPE | ROM | RAM | SPEED (MHz) | PACKAGE | FUNCTIONS | REMARKS | PROBE SDS | REMARKS |
|---|---|---|---|---|---|---|---|---|
| 3315A | 1.5k | 160 | 10 | DIL28/SO28 | 20 I/O lines<br>8-bit timer<br>$V_{DD}$ > 1.8V | | OM1083 | OM1025(LCDS) |
| 3343 | 3k | 224 | 10 | DIL28/SO28 | 20 I/O lines<br>8-bit timer<br>$V_{DD}$ > 1.8V<br>Byte $I^2C$ | | OM1083 | OM1025(LCDS) |
| 3344A | 2k | 224 | 3.58 | DIL28/SO28 | 20 I/O lines<br>8-bit timer<br>DTMF generator | | OM1071 | OM1025(LCDS)<br>+ OM1028 |
| 3346A | 4k | 128 | 10 | DIL28/SO28 | 20 I/O lines<br>8-bit timer<br>Byte $I^2C$<br>256 bytes EEPROM<br>$V_{DD}$ < 1.8V | | OM1076 | |
| 3347 | 1.5k | 64 | 3.58 | DIL20/SO20 | 12 I/O lines<br>8-bit timer<br>DTMF generator | | OM1071 +<br>Adapter_2 | OM1025(LCDS)<br>+ OM1028 |
| 3348A | 8k | 256 | 10 | DIL28/SO28 | 20 I/O lines<br>8-bit timer<br>Byte $I^2C$<br>$V_{DD}$ < 1.8V | | OM1083 | OM1025(LCDS) |
| 3349A | 4k | 224 | 3.58 | DIL28/SO28 | 20 I/O lines<br>8-bit timer<br>DTMF generator | | OM1071 | OM1025(LCDS)<br>+ OM1028 |
| 3350A | 8k | 128 | 3.58 | VSO64 | 30 I/O lines<br>8-bit timer<br>DTMF generator<br>256 bytes EEPROM | | | |
| 3351A | 2k | 64 | 3.58 | DIL28/SO28 | 20 I/O lines<br>8-bit timer<br>DTMF generator<br>128 bytes EEPROM | | OM5000 | |
| 3352A | 6k | 128 | 3.58 | DIL28/SO28 | 20 I/O lines<br>8-bit timer<br>DTMF generator<br>128 byte EEPROM | | OM5000 | |
| 3353A | 6k | 128 | 16 | DIL28/SO28 | 20 I/O lines<br>8-bit timer<br>DTMF generator<br>Ringer out<br>128 bytes EEPROM | March '92 | OM5000 | |
| 3354A | 8k | 256 | 16 | QFP64 | 36 I/O lines<br>8-bit timer<br>DTMF generator<br>Ringer out<br>256 bytes EEPROM | June '92 | OM4829 +<br>OM5003 | OM4829: Probe base |
| 8755A | 0 | 128 | 16 | DIL28/SO28 | 8k OTP<br>20 I/O lines<br>8-bit timer<br>DTMF generator<br>Melody output<br>128 bytes EEPROM | In Development | | |
| 3301B | | | | | | Piggyback for 3315, 3343, 3348 | OM1083 | |
| 3344B | | | | | | Piggyback for 3344, 3347, 3349 | OM1071 | |
| 3346B | | | | | | Piggyback for 3346 | OM1076 | |

**3300 FAMILY CMOS** (Continued)

| TYPE | ROM | RAM | SPEED (MHz) | PACKAGE | FUNCTIONS | REMARKS | PROBE SDS | REMARKS |
|---|---|---|---|---|---|---|---|---|
| 3350B | | | | | | Piggyback for 3350A | OM4829+ OM5003 | |
| 3351B | | | | | | Piggyback for 3351A, 3352A, 3353A | OM5000 | |
| 3354B | | | | | | Piggyback for 3354A | OM4829+ OM5010 | |

# CMOS 16-bit microcontroller family

## 16-BIT CONTROLLERS (XA ARCHITECTURE)

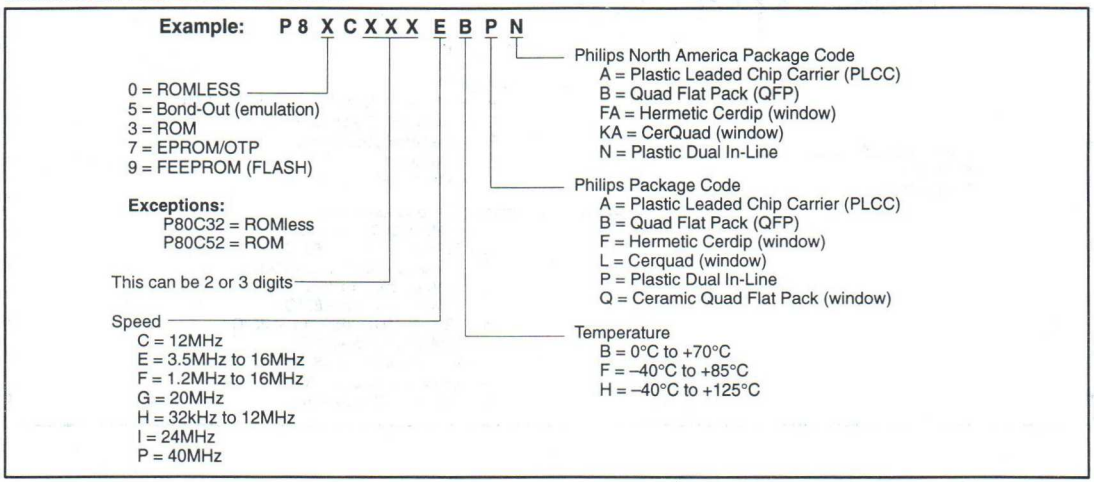
| TYPE | (EP)ROM | RAM | SPEED (MHz) | FUNCTIONS | REMARKS | DEVELOPMENT TOOLS |
|---|---|---|---|---|---|---|
| XA-G1 | 8k | 512 | 30 | 3 timers, watchdog, 2 UARTs | –40 to +125°C | Nohau<br>Ceibo<br>MacCraigor Systems |
| XA-G2 | 16k | 512 | 30 | 3 timers, watchdog, 2 UARTs | –40 to +125°C | Nohau<br>Ceibo<br>MacCraigor Systems |
| XA-G3 | 32k | 512 | 30 | 3 timers, watchdog, 2 UARTs | –40 to +125°C | Nohau<br>Ceibo<br>MacCraigor Systems |

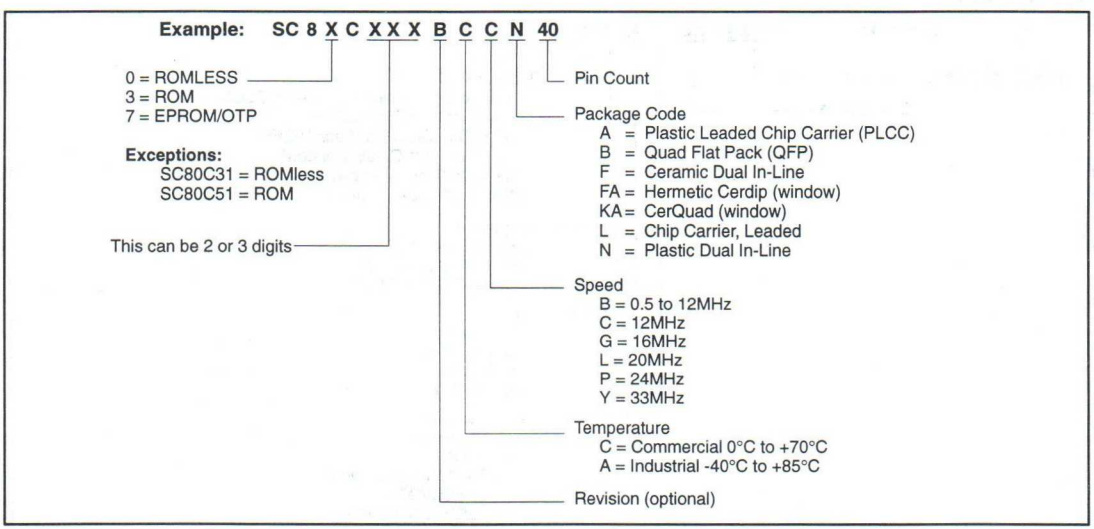
## 16-BIT CONTROLLERS (68000 ARCHITECTURE)

| TYPE | (EP)ROM | RAM | SPEED (MHz) | FUNCTIONS | REMARKS | PHILIPS TOOLS | THIRD-PARTY TOOLS |
|---|---|---|---|---|---|---|---|
| 68070 | – | – | 17.5 | 2 DMA channels, MMU, UART, 16-bit timer, $I^2C$, 68000 bus interface, 16Mb address range | | OM4160 Microcore 1<br>OM4160/2 Microcore 2<br>OM4161 (SBE68070)<br>OM4767/2 XRAY68070SBE high level symbolic debugger<br>OM4222 68070DS development system<br>OM4226 XRAY68070DS high level symbolic debugger | TRACE32-ICE68070 (Lauterbach) |
| 93C101 | 34k | 512 | 15 | Derivative with low power modes | Not for new design | | |
| 90CE201 | 16MB external ROM | 16MB external RAM | 24 | UART, fast $I^2C$, 3 timers (16 bit), Watchdog timer. 68000 software compatible, EMC, QFP64 | –25 to +85°C | OM4162 Microcore 4 | TRACE32 – (Lauterbach) |

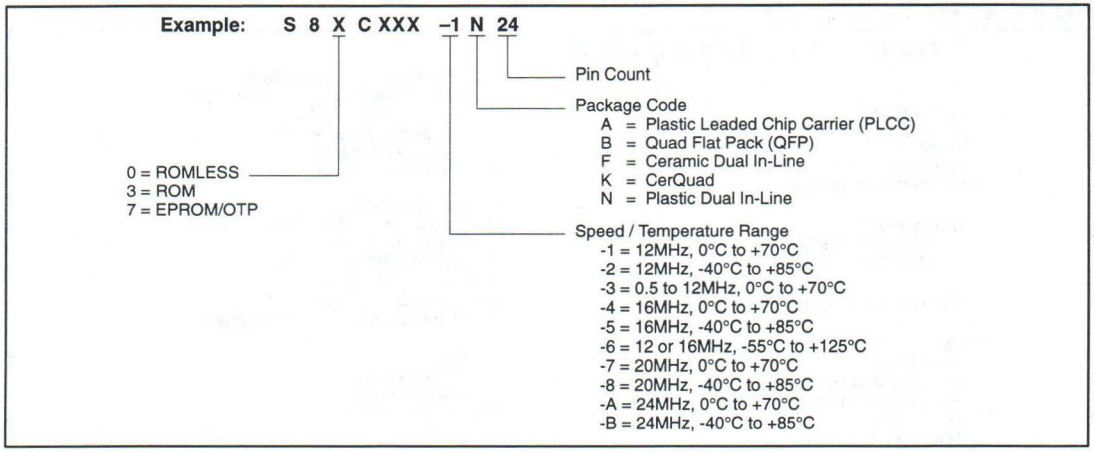
# Ordering Information

## MICROCONTROLLER PRODUCTS

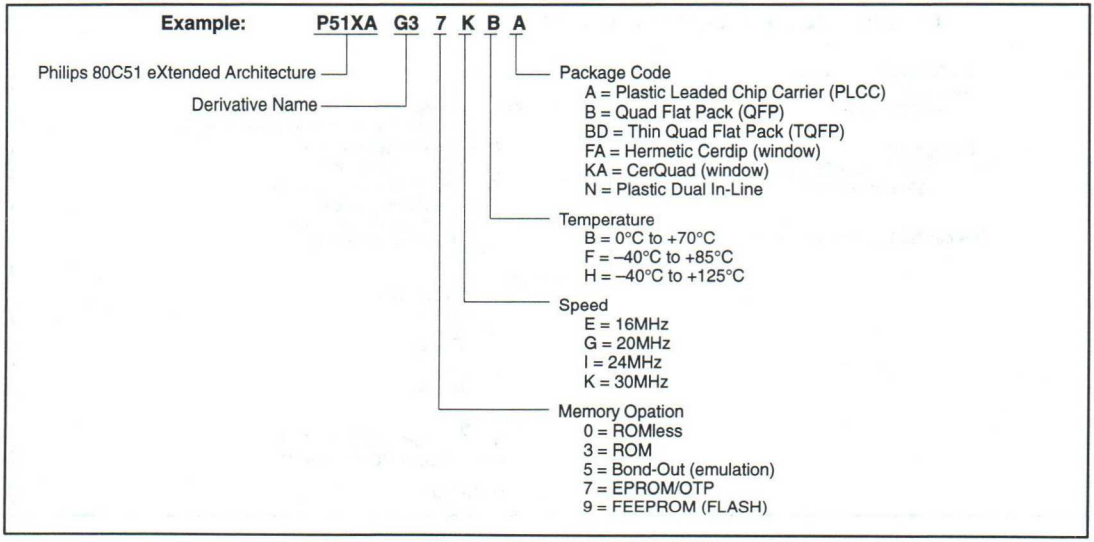


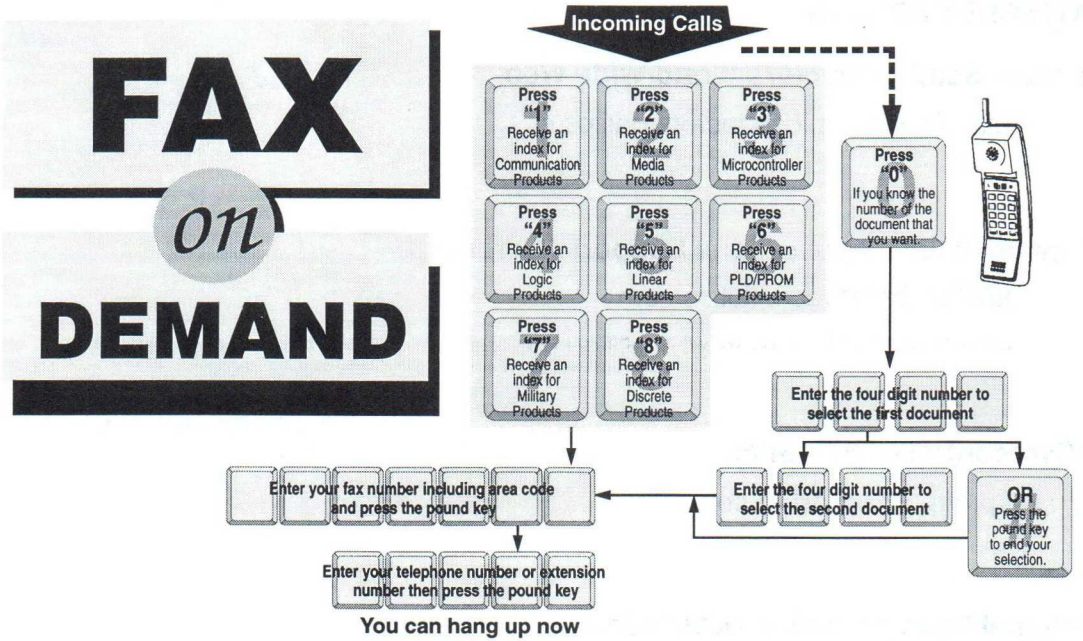
# Ordering Information




## XA PRODUCTS



# FAX-on-DEMAND System




## What is it?

The FAX-on-DEMAND system is a computer facsimile system that allows customers to receive selected documents by fax automatically.

## How does it work?

To order a document, you simply enter the document number. This number can be obtained by asking for an index of available documents to be faxed to you the first time you call the system.

Our system has a selection of the latest product data sheets from Philips with varying page counts. As you know, it takes approximately one minute to FAX one page. This isn't bad if the number of pages is less than 10. But if the document is 37 pages long, be ready for a long transmission!

Philips Semiconductors also maintains product information on the World-Wide Web. Our home page can be located at:
**http://www.semiconductors.philips.com**

## Who do I contact if I have a question about FAX-on-DEMAND?

Contact your local Philips sales office.

## FAX-on-DEMAND phone numbers:

| | |
|---|---|
| England (United Kingdom, Ireland) | 44-181-730-5020 |
| France | 33-1-40-99-60-60 |
| Germany (Austria, Switzerland) | 49-40-23536-357 |
| Italy | 39-167-295502 |
| North America | 1-800-282-200 |

## Locations soon to be in operation:

Hong Kong
Japan
The Netherlands

# Microcontroller internet and bulletin board access

## INTERNET ACCESS

### Philips Semiconductors World Wide Web:

http://www.semiconductors.philips.com

### Internet 80C51 Applications Support Address:

80C51_help@scs.philips.com

Send us your questions and we will respond quickly.

### Microcontroller FTP Site:

ftp://ftp.PhilipsMCU.com

### Internet Microcontroller Newsletter:

To subscribe, send email to:

News-Request@PhilipsMCU.com

### Internet 80C51 Discussion Forum:

Forum-Request@PhilipsMCU.com

### Internet XA 16-bit 80C51 Support Address:

XA_help@scs.philips.com

# Microcontroller internet and bulletin board access

## BULLETIN BOARDS

To better serve our customers, Philips maintains two microcontroller bulletin boards. These computer bulletin board systems feature microcontroller newsletters, application and demonstration programs for download, and the ability to send messages to microcontroller application engineers.

The telephone numbers are:

**North American Bulletin Board**
**MAX 14.400 baud 8-N-1**
**(800) 451-6644 (in the U.S.)**
**or**
**(408) 991-2406**

**European Bulletin Board**
**MAX 14.400 baud**
**Standards V32/V42/V42.bis/HST**
**+31 40 2721102**

---

## Sunnyvale ROMcode Bulletin Board

We also have a ROM code bulletin board through which you can submit ROM codes. This is a closed bulletin board for security reasons. To get an ID, contact your local sales office. The system can be accessed with a 2400, 1200, or 300 baud modem, and is available 24 hours a day.

The telephone number is:

**(408) 991-3459**

---

All code for application notes in this databook are available on the Philips BBS, as well as on the world-wide web.

# 80C51 microcontroller development system support

## DEVELOPMENT SYSTEM CONTACTS

| COMPANY | ADDRESS | TELEPHONE |
|---|---|---|
| Ashling Microsystems Limited | Plassey Technological Park<br>Limerick, Ireland<br><br>Eastern Systems Inc.<br>160 East Main Street<br>Westboro, MA 01581 | (353) 61 334 466<br><br>(508) 366-3220 |
| BSO Tasking | Norfolk Place<br>333 Elm Street<br>Dedham, MA 02026-4530 | (800) 458-8276 |
| Ceibo | 7 Edgestone Ct.<br>Florissant, MO 63033<br><br>Merkazim Building, Industrial Zone<br>P.O. Box 2106<br>Herzelia 46120, ISRAEL | (314) 830-4084<br><br>972-59-555387 |
| Lauterbach Datentechnik GmbH | Fichtenstrasse 27<br>85649 Hofolding<br>Germany<br><br>945 Concord Street<br>Framingham, MA 01701 | 49 8104 894 328<br><br>(508) 620-4521 |
| MetaLink Corp. | 325 E. Elliot Road, Suite 23<br>Chandler, AZ 85225 | (602) 926–0797 |
| Nohau Corp. | 51 E. Campbell Ave.<br>Campbell, CA 95008-2053 | (408) 866–1820 |
| Philips Semiconductors | Corporate Centre<br>Building BAE-2<br>P.O. Box 218<br>5600 MD Eindhoven<br>The Netherlands | 31-40-724223 |
| SIGNUM Systems | 171 E. Thousand Oaks Blvd.,<br>#202<br>Thousand Oaks, CA 91360 | (805) 371-4608 |

## EPROM PROGRAMMING SUPPORT CONTACTS

| | | |
|---|---|---|
| Advin Systems<br>1050-L East Duane Ave.<br>Sunnyvale, CA 94086<br>(408) 243-7000<br>(800) 627-2456 | Logical Devices, Inc.<br>1201 Northwest 65th Place<br>Ft. Lauderdale, FL 33309<br>(305) 974-0967 | North Valley Products<br>P.O. Box 32899<br>San Jose, CA 95152<br>(408) 929-5345 |
| BP Microsystems<br>10681 Haddington #190<br>Houston, TX 77043<br>(800) 225-2102, (713) 461-9430 | Logical Systems<br>P. O. Box 6184<br>Syracuse, NY 13217-6184<br>(315) 478-0722 | Strebor Data Communications<br>1008 N. Nob Hill<br>American Fork, UT 84003<br>(801) 756-3605 |
| Data I/O Corp.<br>10525 Willows Road N.E.<br>P.O. Box 97046<br>Redmond, WA 98073-9746<br>(206) 881-6444 | Needham's Electronics<br>4535 Orange Grove Ave.<br>Sacramento, CA 95841<br>(916) 924-8037 | |

# 80C51 microcontroller development system support

## SOFTWARE SUPPORT CONTACTS

| COMPANY | ADDRESS | TELEPHONE |
|---|---|---|
| Archimedes Software, Inc. | 2159 Union St.<br>San Francisco, CA 94123 | (415) 567–4010 |
| BSO/Tasking | Tasking Software BV<br>P.O. Box 899<br>3800 AW Amersfoort<br>The Netherlands<br><br>BSO Tasking<br>128 Technology Center<br>P.O. Box 9164<br>Waltham, MA 02254-9164 | 31-33-55-85-84 (Telephone)<br>31-33-55-00-33 (Fax)<br><br>(617) 894-7800 (Telephone)<br>(617) 894-0551 (Fax)<br>(710) 324-0760 (Telex)<br>(800) 458-8276 (Toll Free) |
| Franklin Software, Inc. | 888 Saratoga Ave. #2<br>San Jose, CA 95129 | (408) 296–8051 |
| Keil Software | Bretonischer Ring 15<br>85630 Grasbrunn<br>Germany | 49-89-46-50-57 (Telephone)<br>49-89-46-81-62 (Fax) |

**NOTE:**
For more information on Development Support, see Section 9, Vol. 2, IC20.

# 8-bit microcontroller demonstration and evaluation boards

| PRODUCT | DESCRIPTION |
|---|---|
| OM4151, S87C00K | $I^2C$ demonstration board based on 80C51 derivatives |
| OM4238, P8051DB | 8051 family demonstration board |
| OM4128 | 8XC552 evaluation board PEB552 |
| OM4130, PCAN-EVAL | CAN controller evaluation board |
| OM4239 | 8XC592 evaluation board PEB592 |
| OM4240 | 8XCE598 evaluation board PEB598 |
| OM4241 | 8XCE598 evaluation board PDB598 |
| OM4160, SM68070 | 68070 and 66470 demonstration and evaluation board Microcore 1 |
| OM4160/2 | 68070 evaluation board Microcore 2 |
| OM4162 | 9XCE201 evaluation board Microcore 4 |
| OM4280, P83C852DEM | 83C852 demonstration kit |
| OM4281 [1] | 83C852 software evaluation kit |
| P8051DB | 80C51 family development board |
| OM4717 | 83CL410 solar powered demonstration board |
| OM5005, P80CLEVAL | 80CL51 evaluation board |
| DS750 | 8XC750 microcontroller in-circuit emulation development tool |

**NOTE:**
1. The OM4281 is now available only from Ashling Microsystems Ltd. as type SCPC4281.

# Section 2
## 80C51 Technical Description

**80C51-Based**
**8-Bit Microcontrollers**

CONTENTS

80C51 family architecture ..... 2-3
80C51 family hardware description ..... 2-18
80C51 family programmer's guide and instruction set ..... 2-43
80C51 family EPROM products ..... 2-98

## 80C51 ARCHITECTURE

### MEMORY ORGANIZATION

All 80C51 devices have separate address spaces for program and data memory, as shown in Figures 1 and 2. The logical separation of program and data memory allows the data memory to be accessed by 8-bit addresses, which can be quickly stored and manipulated by an 8-bit CPU. Nevertheless, 16-bit data memory addresses can also be generated through the DPTR register.

Program memory (ROM, EPROM) can only be read, not written to. There can be up to 64k bytes of program memory. In the 80C51, the lowest 4k bytes of program are on-chip. In the ROMless versions, all program memory is external. The read strobe for external program memory is the $\overline{PSEN}$ (program store enable).

Data Memory (RAM) occupies a separate address space from Program Memory. In the 80C51, the lowest 128 bytes of data memory are on-chip. Up to 64k bytes of external RAM can be addressed in the external Data Memory space. In the ROMless version, the lowest 128 bytes are on-chip. The CPU generates read and write signals, $\overline{RD}$ and $\overline{WR}$, as needed during external Data Memory accesses.

External Program Memory and external Data Memory may be combined if desired by applying the RD and PSEN signals to the inputs of an AND gate and using the output of the gate as the read strobe to the external Program/Data memory.

### Program Memory

Figure 3 shows a map of the lower part of the Program Memory. After reset, the CPU begins execution from location 0000H. As shown in Figure 3, each interrupt is assigned a fixed location in Program Memory. The interrupt causes the CPU to jump to that location, where it commences execution of the service routine. External Interrupt 0, for example, is assigned to location 0003H. If External Interrupt 0 is going to be used, its service routine must begin at location 0003H. If the interrupt is not going to be used, its service location is available as general purpose Program Memory.

The interrupt service locations are spaced at 8-byte intervals: 0003H for External Interrupt 0, 000BH for Timer 0, 0013H for External Interrupt 1, 001BH for Timer 1, etc. If an interrupt service routine is short enough (as is often the case in control applications), it can reside entirely within that 8-byte interval. Longer service routines can use a jump instruction to skip over subsequent interrupt locations, if other interrupts are in use.

The lowest 4k bytes of Program Memory can either be in the on-chip ROM or in an external ROM. This selection is made by strapping the $\overline{EA}$ (External Access) pin to either $V_{CC}$, or $V_{SS}$. In the 80C51, if the $\overline{EA}$ pin is strapped to $V_{CC}$, then the program fetches to addresses 0000H through 0FFFH are directed to the internal ROM. Program fetches to addresses 1000H through FFFFH are directed to external ROM.

If the $\overline{EA}$ pin is strapped to $V_{SS}$, then all program fetches are directed to external ROM. The ROMless parts (8031, 80C31, etc.) must have this pin externally strapped to $V_{SS}$ to enable them to execute from external Program Memory.

The read strobe to external ROM, $\overline{PSEN}$, is used for all external program fetches. $\overline{PSEN}$ is not activated for internal program fetches.

The hardware configuration for external program execution is shown in Figure 4. Note that 16 I/O lines (Ports 0 and 2) are dedicated to bus functions during external Program Memory fetches. Port 0 (P0 in Figure 4) serves as a multiplexed address/data bus. It emits the low byte of the Program Counter (PCL) as an address, and then goes into a float state awaiting the arrival of the code byte from the Program Memory. During the time that the low byte of the Program Counter is valid on Port 0, the signal ALE (Address Latch Enable) clocks this byte into an address latch. Meanwhile, Port 2 (P2 in Figure 4) emits the high byte of the Program Counter (PCH). Then $\overline{PSEN}$ strobes the EPROM and the code byte is read into the microcontroller.

Program Memory addresses are always 16 bits wide, even though the actual amount of Program Memory used may be less than 64k bytes. External program execution sacrifices two of the 8-bit ports, P0 and P2, to the function of addressing the Program Memory.

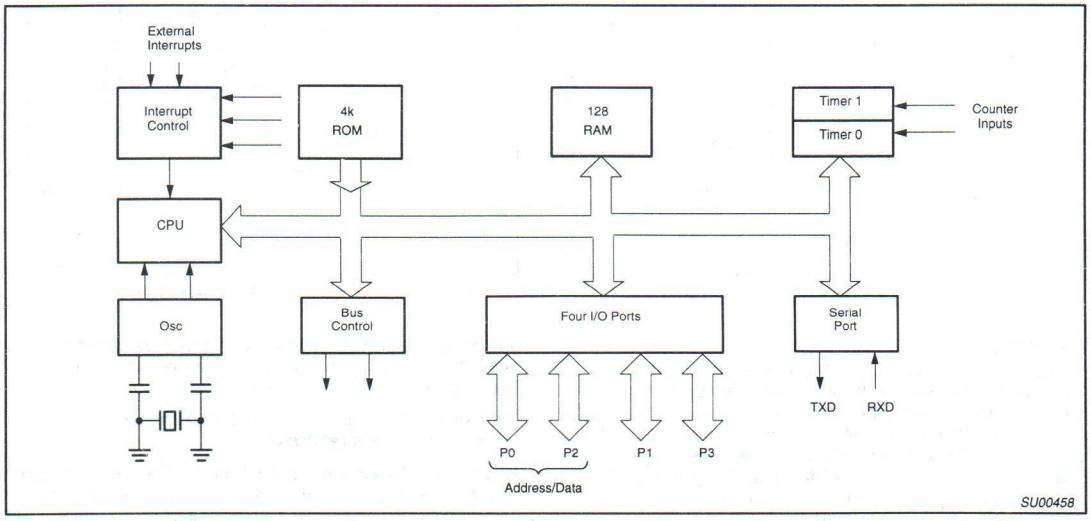



**Figure 1. 80C51 Block Diagram**

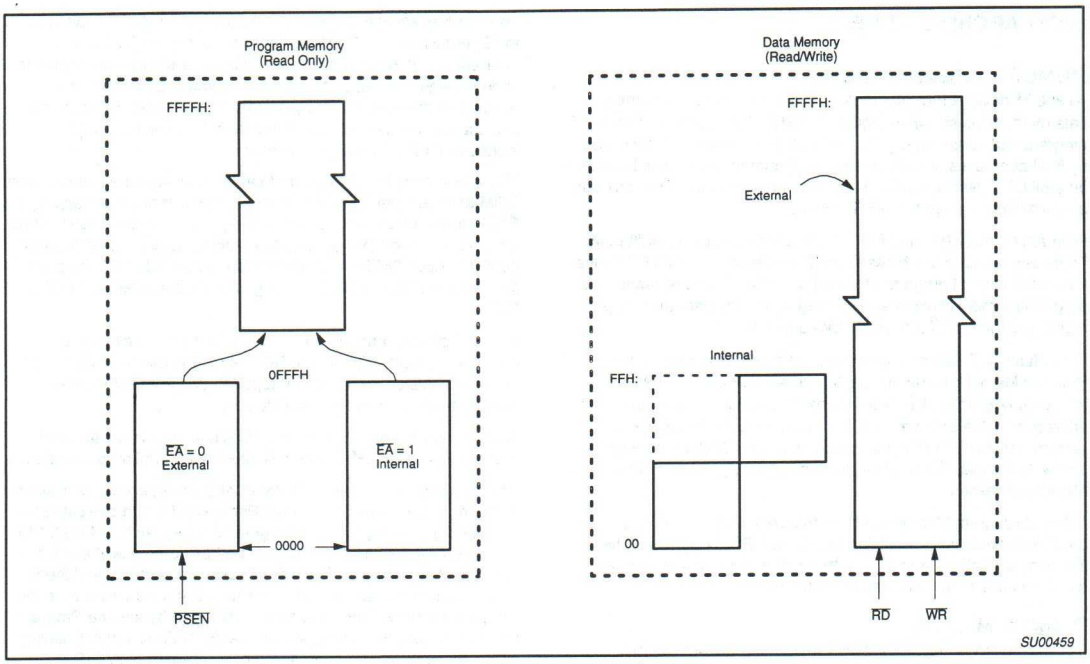



**Figure 2. 80C51 Memory Structure**

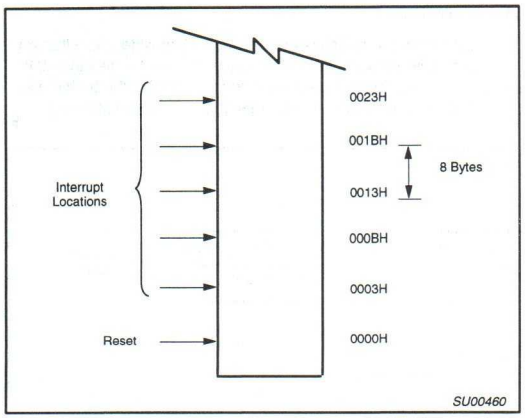



**Figure 3. 80C51 Program Memory**

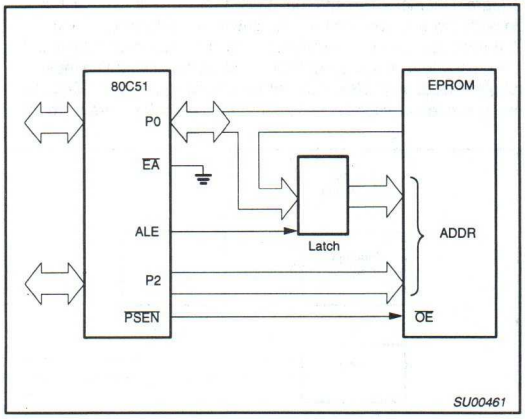



**Figure 4. Executing from External Program Memory**

## Data Memory

The right half of Figure 2 shows the internal and external Data Memory spaces available to the 80C51 user. Figure 5 shows a hardware configuration for accessing up to 2k bytes of external RAM. The CPU in this case is executing from internal ROM. Port 0 serves as a multiplexed address/data bus to the RAM, and 3 lines of Port 2 are being used to page the RAM. The CPU generates $\overline{RD}$ and $\overline{WR}$ signals as needed during external RAM accesses. There can be up to 64k bytes of external Data Memory. External Data Memory addresses can be either 1 or 2 bytes wide. One-byte addresses are often used in conjunction with one or more other I/O lines to page the RAM, as shown in Figure 5.

Two-byte addresses can also be used, in which case the high address byte is emitted at Port 2.

Internal Data Memory is mapped in Figure 6. The memory space is shown divided into three blocks, which are generally referred to as the Lower 128, the Upper 128, and SFR space.

Internal Data Memory addresses are always one byte wide, which implies an address space of only 256 bytes. However, the addressing modes for internal RAM can in fact accommodate 384 bytes, using a simple trick. Direct addresses higher than 7FH access one memory space, and indirect addresses higher than 7FH access a different memory space. Thus Figure 6 shows the Upper 128 and SFR space occupying the same block of addresses, 80H through FFH, although they are physically separate entities.

The Lower 128 bytes of RAM are present in all 80C51 devices as mapped in Figure 7. The lowest 32 bytes are grouped into 4 banks of 8 registers. Program instructions call out these registers as R0 through R7. Two bits in the Program Status Word (PSW) select which register bank is in use. This allows more efficient use of code space, since register instructions are shorter than instructions that use direct addressing.

The next 16 bytes above the register banks form a block of bit-addressable memory space. The 80C51 instruction set includes a wide selection of single-bit instructions, and the 128 bits in this area can be directly addressed by these instructions. The bit addresses in this area are 00H through 7FH.

All of the bytes in the Lower 128 can be accessed by either direct or indirect addressing. The Upper 128 (Figure 8) can only be accessed by indirect addressing.

Figure 9 gives a brief look at the Special Function Register (SFR) space. SFRs include the Port latches, timers, peripheral controls, etc. These registers can only be accessed by direct addressing. Sixteen addresses in SFR space are both byte- and bit-addressable. The bit-addressable SFRs are those whose address ends in 0H or 8H.

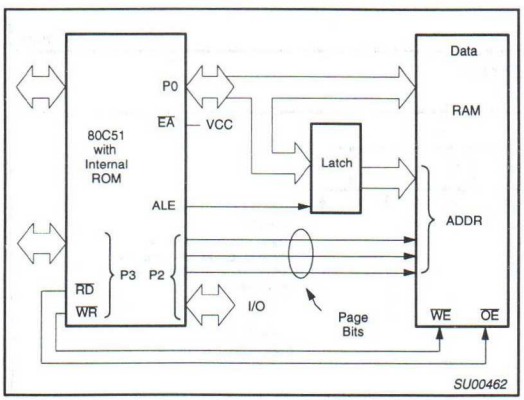



**Figure 5. Accessing External Data Memory If the Program Memory Is Internal, the Other Bits of P2 Are Available as I/O**

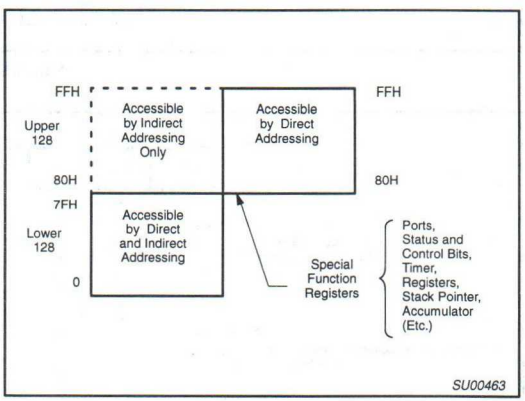



**Figure 6. Internal Data Memory**

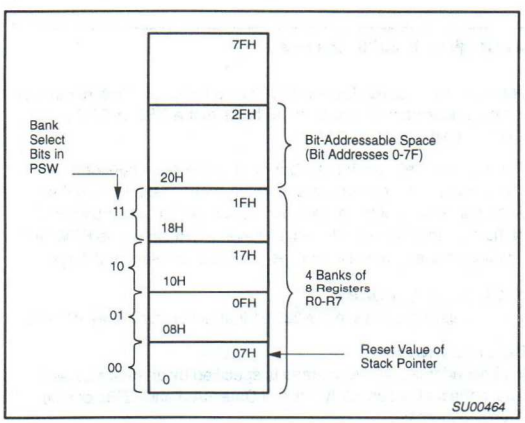



**Figure 7. Lower 128 Bytes of Internal RAM**

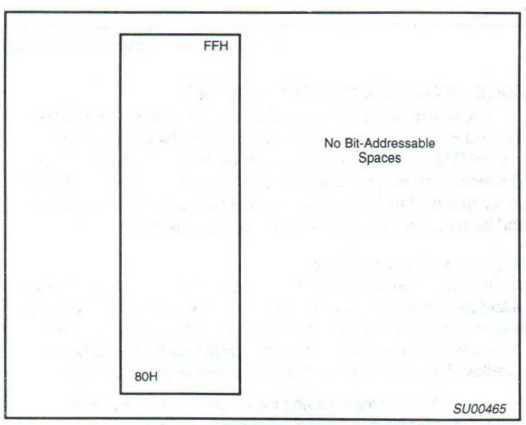



**Figure 8. Upper 128 Bytes of Internal RAM**

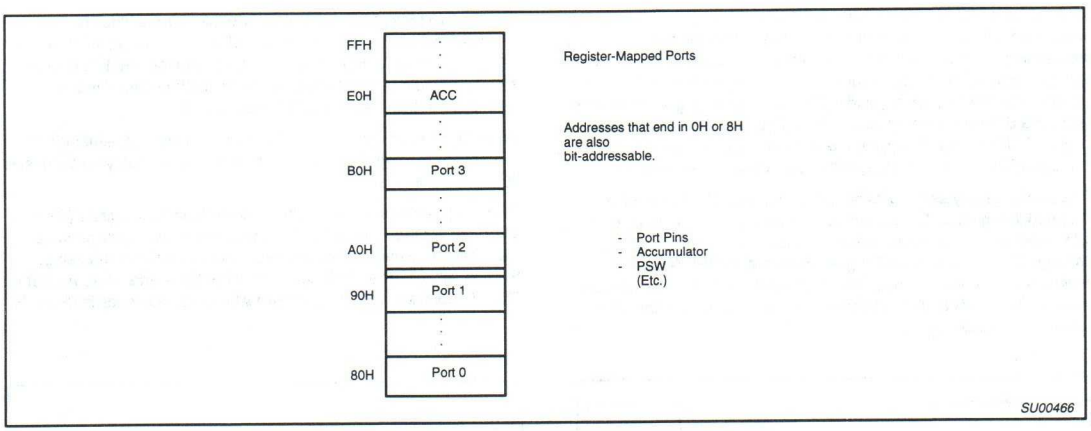



**Figure 9. SFR Space**

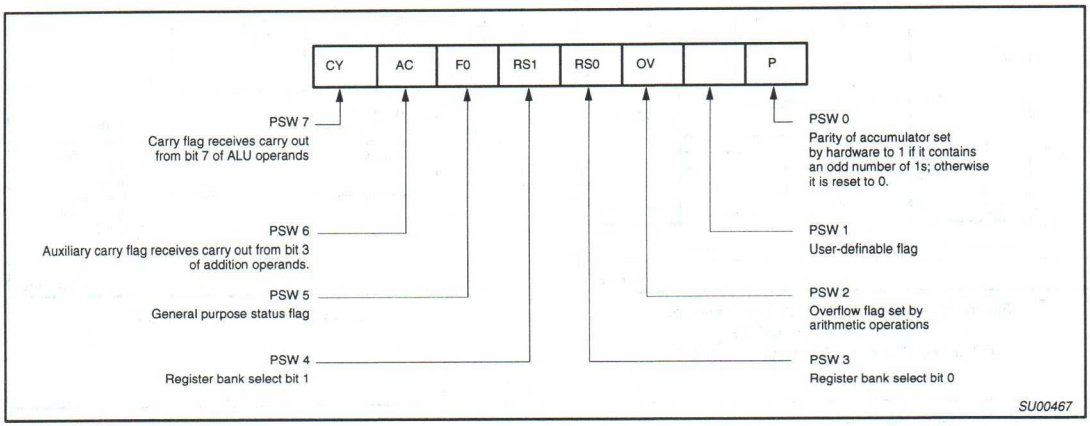



**Figure 10. PSW (Program Status Word) Register in 80C51 Devices**

## 80C51 FAMILY INSTRUCTION SET

The 80C51 instruction set is optimized for 8-bit control applications. It provides a variety of fast addressing modes for accessing the internal RAM to facilitate byte operations on small data structures. The instruction set provides extensive support for one-bit variables as a separate data type, allowing direct bit manipulation in control and logic systems that require Boolean processing.

### Program Status Word

The Program Status Word (PSW) contains several status bits that reflect the current state of the CPU. The PSW, shown in Figure 10, resides in the SFR space. It contains the Carry bit, the Auxiliary Carry (for BCD operations), the two register bank select bits, the Overflow flag, a Parity bit, and two user-definable status flags.

The Carry bit, other than serving the function of a Carry bit in arithmetic operations, also serves as the "Accumulator" for a number of Boolean operations.

The bits RS0 and RS1 are used to select one of the four register banks shown in Figure 7. A number of instructions refer to these RAM locations as R0 through R7. The selection of which of the four is being referred to is made on the basis of the RS0 and RS1 at execution time.

The Parity bit reflects the number of 1s in the Accumulator: P = 1 if the Accumulator contains an odd number of 1s, and P = 0 if the Accumulator contains an even number of 1s. Thus the number of 1s in the Accumulator plus P is always even. Two bits in the PSW are uncommitted and may be used as general purpose status flags.

### Addressing Modes

The addressing modes in the 80C51 instruction set are as follows:

#### Direct Addressing

In direct addressing the operand is specified by an 8-bit address field in the instruction. Only internal Data RAM and SFRs can be directly addressed.

**Indirect Addressing**
In indirect addressing the instruction specifies a register which contains the address of the operand. Both internal and external RAM can be indirectly addressed.

The address register for 8-bit addresses can be R0 or R1 of the selected bank, or the Stack Pointer. The address register for 16-bit addresses can only be the 16-bit "data pointer" register, DPTR.

**Register Instructions**
The register banks, containing registers R0 through R7, can be accessed by certain instructions which carry a 3-bit register specification within the opcode of the instruction. Instructions that access the registers this way are code efficient, since this mode eliminates an address byte. When the instruction is executed, one of the eight registers in the selected bank is accessed. One of four banks is selected at execution time by the two bank select bits in the PSW.

**Register-Specific Instructions**
Some instructions are specific to a certain register. For example, some instructions always operate on the Accumulator, or Data Pointer, etc., so no address byte is needed to point to it. The opcode itself does that. Instructions that refer to the Accumulator as A assemble as accumulator specific opcodes.

**Immediate Constants**
The value of a constant can follow the opcode in Program Memory. For example,

```
MOV  A, #100
```

loads the Accumulator with the decimal number 100. The same number could be specified in hex digits as 64H.

**Indexed Addressing**
Only program Memory can be accessed with indexed addressing, and it can only be read. This addressing mode is intended for reading look-up tables in Program Memory A 16-bit base register (either DPTR or the Program Counter) points to the base of the table, and the Accumulator is set up with the table entry number.

The address of the table entry in Program Memory is formed by adding the Accumulator data to the base pointer.

Another type of indexed addressing is used in the "case jump" instruction. In this case the destination address of a jump instruction is computed as the sum of the base pointer and the Accumulator data.

## Arithmetic Instructions

The menu of arithmetic instructions is listed in Table 1. The table indicates the addressing modes that can be used with each instruction to access the <byte> operand. For example, the ADD A,<byte> instruction can be written as:

```
ADD   a, 7FH (direct addressing)
ADD   A, @R0 (indirect addressing)
ADD   a, R7 (register addressing)
ADD   A, #127 (immediate constant)
```

The execution times listed in Table 1 assume a 12MHz clock frequency. All of the arithmetic instructions execute in 1µs except the INC DPTR instruction, which takes 2µs, and the Multiply and Divide instructions, which take 4µs.

Note that any byte in the internal Data Memory space can be incremented without going through the Accumulator.

One of the INC instructions operates on the 16-bit Data Pointer. The Data Pointer is used to generate 16-bit addresses for external memory, so being able to increment it in one 16-bit operation is a useful feature.

The MUL AB instruction multiplies the Accumulator by the data in the B register and puts the 16-bit product into the concatenated B and Accumulator registers.

The DIV AB instruction divides the Accumulator by the data in the B register and leaves the 8-bit quotient in the Accumulator, and the 8-bit remainder in the B register.

Oddly enough, DIV AB finds less use in arithmetic "divide" routines than in radix conversions and programmable shift operations. An example of the use of DIV AB in a radix conversion will be given later. In shift operations, dividing a number by 2n shifts its n bits to the right. Using DIV AB to perform the division completes the shift in 4µs and leaves the B register holding the bits that were shifted out. The DA A instruction is for BCD arithmetic operations. In BCD arithmetic, ADD and ADDC instructions should always be followed by a DA A operation, to ensure that the result is also in BCD. Note that DA A will not convert a binary number to BCD. The DA A operation produces a meaningful result only as the second step in the addition of two BCD bytes.

**Table 1. 80C51 Arithmetic Instructions**

| MNEMONIC | OPERATION | ADDRESSING MODES | | | | EXECUTION |
|---|---|---|---|---|---|---|
| | | DIR | IND | REG | IMM | TIME (µs) |
| ADD A,<byte> | A = A + <byte> | X | X | X | X | 1 |
| ADDC A,<byte> | A = A + <byte> + C | X | X | X | X | 1 |
| SUBB A,<byte> | A = A – <byte> – C | X | X | X | X | 1 |
| INC A | A = A + 1 | Accumulator only | | | | 1 |
| INC <byte> | <byte> = <byte> + 1 | X | X | X | | 1 |
| INC DPTR | DPTR = DPTR + 1 | Data Pointer only | | | | 2 |
| DEC A | A = A – 1 | Accumulator only | | | | 1 |
| DEC <byte> | <byte> = <byte> – 1 | X | X | X | | 1 |
| MUL AB | B:A = B x A | ACC and B only | | | | 4 |
| DIV AB | A = Int[A/B]<br>B = Mod[A/B] | ACC and B only | | | | 4 |
| DA A | Decimal Adjust | Accumulator only | | | | 1 |

## Logical Instructions

Table 2 shows the list of 80C51 logical instructions. The instructions that perform Boolean operations (AND, OR, Exclusive OR, NOT) on bytes perform the operation on a bit-by-bit basis. That is, if the Accumulator contains 00110101B and byte contains 01010011B, then:

```
ANL   A, <byte>
```

will leave the Accumulator holding 00010001B.

The addressing modes that can be used to access the <byte> operand are listed in Table 2.

The ANL A, <byte> instruction may take any of the forms:

```
ANL   A,7FH (direct addressing)
ANL   A,@R1 (indirect addressing)
ANL   A,R6 (register addressing)
ANL   A,#53H (immediate constant)
```

All of the logical instructions that are Accumulator-specific execute in 1µs (using a 12MHz clock). The others take 2µs.

Note that Boolean operations can be performed on any byte in the internal Data Memory space without going through the Accumulator. The XRL <byte>, #data instruction, for example, offers a quick and easy way to invert port bits, as in XRL P1, #OFFH.

If the operation is in response to an interrupt, not using the Accumulator saves the time and effort to push it onto the stack in the service routine.

The Rotate instructions (RL, A, RLC A, etc.) shift the Accumulator 1 bit to the left or right. For a left rotation, the MSB rolls into the LSB position. For a right rotation, the LSB rolls into the MSB position.

The SWAP A instruction interchanges the high and low nibbles within the Accumulator. This is a useful operation in BCD manipulations. For example, if the Accumulator contains a binary number which is known to be less than 100, it can be quickly converted to BCD by the following code:

```
MOVE  B,#10
DIV   AB
SWAP  A
ADD   A,B
```

Dividing the number by 10 leaves the tens digit in the low nibble of the Accumulator, and the ones digit in the B register. The SWAP and ADD instructions move the tens digit to the high nibble of the Accumulator, and the ones digit to the low nibble.

## Data Transfers

### Internal RAM

Table 3 shows the menu of instructions that are available for moving data around within the internal memory spaces, and the addressing modes that can be used with each one. With a 12MHz clock, all of these instructions execute in either 1 or 2µs.

The MOV <dest>, <src> instruction allows data to be transferred between any two internal RAM or SFR locations without going through the Accumulator. Remember, the Upper 128 bytes of data RAM can be accessed only by indirect addressing, and SFR space only by direct addressing.

Note that in 80C51 devices, the stack resides in on-chip RAM, and grows upwards. The PUSH instruction first increments the Stack Pointer (SP), then copies the byte into the stack. PUSH and POP use only direct addressing to identify the byte being saved or restored, but the stack itself is accessed by indirect addressing using the SP register. This means the stack can go into the Upper 128 bytes of RAM, if they are implemented, but not into SFR space.

The Upper 128 bytes of RAM are not implemented in the 80C51 nor in its ROMless or EPROM counterparts. With these devices, if the SP points to the Upper 128, PUSHed bytes are lost, and POPed bytes are indeterminate.

The Data Transfer instructions include a 16-bit MOV that can be used to initialize the Data Pointer (DPTR) for look-up tables in Program Memory, or for 16-bit external Data Memory accesses.

**Table 2. 80C51 Logical Instructions**

| MNEMONIC | OPERATION | ADDRESSING MODES | | | | EXECUTION |
|---|---|---|---|---|---|---|
| | | DIR | IND | REG | IMM | TIME (µs) |
| ANL A,<byte> | A = A.AND. <byte> | X | X | X | X | 1 |
| ANL <byte>,A | <byte> = <byte> .AND.A | X | | | | 1 |
| ANL <byte>,#data | <byte> = <byte> .AND.#data | X | | | | 2 |
| ORL A,<byte> | A = A.OR.<byte> | X | X | X | X | 1 |
| ORL <byte>,A | <byte> = <byte> .OR.A | X | | | | 1 |
| ORL <byte>,#data | <byte> = <byte> .OR.#data | X | | | | 2 |
| XRL A,<byte> | A = A.XOR. <byte> | X | X | X | X | 1 |
| XRL <byte>,A | <byte> = <byte> .XOR.A | X | | | | 1 |
| XRL <byte>,#data | <byte> = <byte> .XOR.#data | X | | | | 2 |
| CRL A | A = 00H | Accumulator only | | | | 1 |
| CPL A | A = .NOT.A | Accumulator only | | | | 1 |
| RL A | Rotate ACC Left 1 bit | Accumulator only | | | | 1 |
| RLC A | Rotate Left through Carry | Accumulator only | | | | 1 |
| RR A | Rotate ACC Right 1 bit | Accumulator only | | | | 1 |
| RRC A | Rotate Right through Carry | Accumulator only | | | | 1 |
| SWAP A | Swap Nibbles in A | Accumulator only | | | | 1 |

**Table 3. Data Transfer Instructions that Access Internal Data Memory Space**

| MNEMONIC | OPERATION | ADDRESSING MODES | | | | EXECUTION |
|---|---|---|---|---|---|---|
| | | DIR | IND | REG | IMM | TIME (μs) |
| MOV A,<src> | A = <src> | X | X | X | X | 1 |
| MOV <dest>,A | <dest> = A | X | X | X | | 1 |
| MOV <dest>,<src> | <dest> = <src> | X | X | X | X | 2 |
| MOV DPTR,#data16 | DPTR = 16-bit immediate constant | | | | X | 2 |
| PUSH <src> | INC SP:MOV"@SP",<src> | X | | | | 2 |
| POP <dest> | MOV <dest>,"@SP":DEC SP | X | | | | 2 |
| XCH A,<byte> | ACC and <byte> exchange data | X | X | X | | 1 |
| XCHD A,@Ri | ACC and @Ri exchange low nibbles | | X | | | 1 |

The XCH A, <byte> instruction causes the Accumulator and addressed byte to exchange data. The XCHD A, @Ri instruction is similar, but only the low nibbles are involved in the exchange.

To see how XCH and XCHD can be used to facilitate data manipulations, consider first the problem of shifting an 8-digit BCD number two digits to the right. Figure 11 shows how this can be done using direct MOVs, and for comparison how it can be done using XCH instructions. To aid in understanding how the code works, the contents of the registers that are holding the BCD number and the content of the Accumulator are shown alongside each instruction to indicate their status after the instruction has been executed.

After the routine has been executed, the Accumulator contains the two digits that were shifted out on the right. Doing the routine with direct MOVs uses 14 code bytes and 9μs of execution time (assuming a 12MHz clock). The same operation with XCHs uses only 9 bytes and executes almost twice as fast.

To right-shift by an odd number of digits, a one-digit shift must be executed.

Figure 12 shows a sample of code that will right-shift a BCD number one digit, using the XCHD instruction. Again, the contents of the registers holding the number and of the Accumulator are shown alongside each instruction.

First, pointers R1 and R0 are set up to point to the two bytes containing the last four BCD digits. Then a loop is executed which leaves the last byte, location 2EH, holding the last two digits of the shifted number. The pointers are decremented, and the loop is repeated for location 2DH. The CJNE instruction (Compare and Jump if Not Equal) is a loop control that will be described later. The loop executed from LOOP to CJNE for R1 = 2EH, 2DH, 2CH, and 2BH. At that point the digit that was originally shifted out on the right has propagated to location 2AH. Since that location should be left with 0s, the lost digit is moved to the Accumulator.

**External RAM**

Table 4 shows a list of the Data Transfer instructions that access external Data Memory. Only indirect addressing can be used. The choice is whether to use a one-byte address, @Ri, where Ri can be either R0 or R1 of the selected register bank, or a two-byte address, @DPTR. The disadvantage to using 16-bit addresses if only a few k bytes of external RAM are involved is that 16-bit addresses use all 8 bits of Port 2 as address bus. On the other hand, 8-bit addresses allow one to address a few bytes of RAM, as shown in Figure 5, without having to sacrifice all of Port 2. All of these instructions execute in 2 μs, with a 12MHz clock.

Note that in all external Data RAM accesses, the Accumulator is always either the destination or source of the data.

The read and write strobes to external RAM are activated only during the execution of a MOVX instruction. Normally these signals are inactive, and in fact if they're not going to be used at all, their pins are available as extra I/O lines.

| | | 2A | 2B | 2C | 2D | 2E | ACC |
|---|---|---|---|---|---|---|---|
| MOV | A,2EH | 00 | 12 | 34 | 56 | 78 | 78 |
| MOV | 2EH,2DH | 00 | 12 | 34 | 56 | 56 | 78 |
| MOV | 2DH,2CH | 00 | 12 | 34 | 34 | 56 | 78 |
| MOV | 2CH,2BH | 00 | 12 | 12 | 34 | 56 | 78 |
| MOV | 2BH,#0 | 00 | 00 | 12 | 34 | 56 | 78 |

**A. Using direct MOVs: 14 bytes, 9 μs**

| | | 2A | 2B | 2C | 2D | 2E | ACC |
|---|---|---|---|---|---|---|---|
| CLR | A | 00 | 12 | 34 | 56 | 78 | 00 |
| XCH | A,2BH | 00 | 00 | 34 | 56 | 78 | 12 |
| XCH | A,2CH | 00 | 00 | 12 | 56 | 78 | 34 |
| XCH | A,2DH | 00 | 00 | 12 | 34 | 78 | 56 |
| XCH | A2EH | 00 | 00 | 12 | 34 | 56 | 78 |

**B. Using XCHs: 9 bytes, 5 μs**

SU00468

**Figure 11. Shifting a BCD Number Two Digits to the Right**

| | | | 2A | 2B | 2C | 2D | 2E | ACC |
|---|---|---|---|---|---|---|---|---|
| | MOV | R1,#2EH | 00 | 12 | 34 | 56 | 78 | XX |
| | MOV | R0,#2DH | 00 | 12 | 34 | 56 | 78 | XX |
| | loop for R1 = 2EH: | | | | | | | |
| LOOP: | MOV | A,@R1 | 00 | 12 | 34 | 56 | 78 | 78 |
| | XCHD | A,@R0 | 00 | 12 | 34 | 58 | 78 | 76 |
| | SWAP | A | 00 | 12 | 34 | 58 | 78 | 67 |
| | MOV | @R1,A | 00 | 12 | 34 | 58 | 67 | 67 |
| | DEC | R1 | 00 | 12 | 34 | 58 | 67 | 67 |
| | DEC | R0 | 00 | 12 | 34 | 58 | 67 | 67 |
| | CJNE | R1,#2AH,LOOP | | | | | | |
| | loop for R1 = 2DH: | | 00 | 12 | 38 | 45 | 67 | 45 |
| | loop for R1 = 2CH: | | 00 | 18 | 23 | 45 | 67 | 23 |
| | loop for R1 = 2BH: | | 08 | 01 | 23 | 45 | 67 | 01 |
| | CLR | A | 08 | 01 | 23 | 45 | 67 | 00 |
| | XCH | A,2AH | 00 | 01 | 23 | 45 | 67 | 08 |

SU00469

**Figure 12. Shifting a BCD Number One Digit to the Right**

**Table 4. 80C51 Data Transfer Instructions that Access External Data Memory Space**

| ADDRESS WIDTH | MNEMONIC | OPERATION | EXECUTION TIME (μs) |
|---|---|---|---|
| 8 bits | MOVX A,@Ri | Read external RAM @Ri | 2 |
| 8 bits | MOVX @Ri,A | Write external RAM @ Ri | 2 |
| 16 bits | MOVX A,@DPTR | Read external RAM @ DPTR | 2 |
| 16 bits | MOVX @DPTR,A | Write external RAM @ DPTR | 2 |

**Table 5. 80C51 Lookup Table Read Instructions**

| MNEMONIC | OPERATION | EXECUTION TIME (μs) |
|---|---|---|
| MOVC A,@A+DPTR | Read program memory at (A + DPTR) | 2 |
| MOVC A,@A+PC | Read program memory at (A + PC) | 2 |

**Lookup Tables**

Table 5 shows the two instructions that are available for reading lookup tables in Program Memory. Since these instructions access only Program Memory, the lookup tables can only be read, not updated.

If the table access is to external Program Memory, then the read strobe is $\overline{PSEN}$.

The mnemonic is MOVC for "move constant." The first MOVC instruction in Table 5 can accommodate a table of up to 256 entries numbered 0 through 255. The number of the desired entry is loaded into the Accumulator, and the Data Pointer is set up to point to the beginning of the table. Then:

```
MOVC    A,@A+DPTR
```

copies the desired table entry into the Accumulator.

The other MOVC instruction works the same way, except the Program Counter (PC) is used as the table base, and the table is accessed through a subroutine. First the number of the desired entry is loaded into the Accumulator, and the subroutine is called:

```
        MOV     A,ENTRY NUMBER
        CALL    TABLE
```

The subroutine "TABLE" would look like this:

```
TABLE:  MOVC    A,@A+PC
        RET
```

The table itself immediately follows the RET (return) instruction in Program Memory. This type of table can have up to 255 entries, numbered 1 through 255. Number 0 cannot be used, because at the time the MOVC instruction is executed, the PC contains the address of the RET instruction. An entry numbered 0 would be the RET opcode itself.

## Boolean Instructions

80C51 devices contain a complete Boolean (single-bit) processor. The internal RAM contains 128 addressable bits, and the SFR space can support up to 128 addressable bits as well. All of the port lines are bit-addressable, and each one can be treated as a separate single-bit port. The instructions that access these bits are not just conditional branches, but a complete menu of move, set, clear, complement, OR, and AND instructions. These kinds of bit operations are not easily obtained in other architectures with any amount of byte-oriented software.

The instruction set for the Boolean processor is shown in Table 6. All bit accesses are by direct addressing.

Bit addresses 00H through 7FH are in the Lower 128, and bit addresses 80H through FFH are in SFR space.

Note how easily an internal flag can be moved to a port pin:

```
        MOV     C,FLAG
        MOV     P1.0,C
```

In this example, FLAG is the name of any addressable bit in the Lower 128 or SFR space. An I/O line (the LSB of Port 1, in this case) is set or cleared depending on whether the flag bit is 1 or 0.

The Carry bit in the PSW is used as the single-bit Accumulator of the Boolean processor. Bit instructions that refer to the Carry bit as C assemble as Carry-specific instructions (CLR C, etc.). The Carry bit also has a direct address, since it resides in the PSW register, which is bit-addressable.

Note that the Boolean instruction set includes ANL and ORL operations, but not the XRL (Exclusive OR) operation. An XRL operation is simple to implement in software. Suppose, for example, it is required to form the Exclusive OR of two bits:

```
        C = bit1 .XRL. bit2
```

The software to do that could be as follows:

```
        MOV     C,bit1
        JNB     bit2,OVER
        CPL     C
OVER:   (continue)
```

First, bit1 is moved to the Carry. If bit2 = 0, then C now contains the correct result. That is, bit1 .XRL. bit2 = bit1 if bit2 = 0. On the other hand, if bit2 = 1, C now contains the complement of the correct result. It need only be inverted (CPL C) to complete the operation.

This code uses the JNB instruction, one of a series of bit-test instructions which execute a jump if the addressed bit is set (JC, JB, JBC) or if the addressed bit is not set (JNC, JNB). In the above case, bit2 is being tested, and if bit2 = 0, the CPL C instruction is jumped over.

JBC executes the jump if the addressed bit is set, and also clears the bit. Thus a flag can be tested and cleared in one operation. All the PSW bits are directly addressable, so the Parity bit, or the general purpose flags, for example, are also available to the bit-test instructions.

**Relative Offset**

The destination address for these jumps is specified to the assembler by a label or by an actual address in Program memory. However, the destination address assembles to a relative offset byte. This is a signed (two's complement) offset byte which is added to the PC in two's complement arithmetic if the jump is executed. The range of the jump is therefore –128 to +127 Program Memory bytes relative to the first byte following the instruction.

**Table 6. 80C51 Boolean Instructions**

| MNEMONIC | | OPERATION | EXECUTION TIME (μs) |
|---|---|---|---|
| ANL | C,bit | C = C.AND.bit | 2 |
| ANL | C,/bit | C = C.AND..NOT.bit | 2 |
| ORL | C,bit | C = C.OR.bit | 2 |
| ORL | C,/bit | C = C.OR..NOT.bit | 2 |
| MOV | C,bit | C = bit | 1 |
| MOV | bit,C | bit = C | 2 |
| CLR | C | C = 0 | 1 |
| CLR | bit | bit = 0 | 1 |
| SETB | C | C = 1 | 1 |
| SETB | bit | bit = 1 | 1 |
| CPL | C | C = .NOT.C | 1 |
| CPL | bit | bit = .NOT.bit | 1 |
| JC | rel | Jump if C = 1 | 2 |
| JNC | rel | Jump if C = 0 | 2 |
| JB | bit,rel | Jump if bit = 1 | 2 |
| JNB | bit,rel | Jump if bit = 0 | 2 |
| JBC | bit,rel | Jump if bit = 1; CLR bit | 2 |

**Table 7. Unconditional Jumps in 80C51 Devices**

| MNEMONIC | | OPERATION | EXECUTION TIME (μs) |
|---|---|---|---|
| JMP | addr | Jump to addr | 2 |
| JMP | @A+DPTR | Jump to A + DPTR | 2 |
| CALL | addr | Call subroutine at addr | 2 |
| RET | | Return from subroutine | 2 |
| RETI | | Return from interrupt | 2 |
| NOP | | No operation | 1 |

## Jump Instructions

Table 7 shows the list of unconditional jumps with execution time for a 12MHz clock.

The table lists a single "JMP addr" instruction, but in fact there are three SJMP, LJMP, and AJMP, which differ in the format of the destination address. JMP is a generic mnemonic which can be used if the programmer does not care which way the jump is encoded.

The SJMP instruction encodes the destination address as a relative offset, as described above. The instruction is 2 bytes long, consisting of the opcode and the relative offset byte. The jump distance is limited to a range of –128 to +127 bytes relative to the instruction following the SJMP.

The LJMP instruction encodes the destination address as a 16-bit constant. The instruction is 3 bytes long, consisting of the opcode and two address bytes. The destination address can be anywhere in the 64k Program Memory space.

The AJMP instruction encodes the destination address as an 11-bit constant. The instruction is 2 bytes long, consisting of the opcode, which itself contains 3 of the 11 address bits, followed by another byte containing the low 8 bits of the destination address. When the instruction is executed, these 11 bits are simply substituted for the low 11 bits in the PC. The high 5 bits stay the same. Hence the destination has to be within the same 2k block as the instruction following the AJMP.

In all cases the programmer specifies the destination address to the assembler in the same way: as a label or as a 16-bit constant. The assembler will put the destination address into the correct format for the given instruction. If the format required by the instruction will not support the distance to the specified destination address, a "Destination out of range" message is written into the List file.

The JMP @A+DPTR instruction supports case jumps. The destination address is computed at execution time as the sum of the 16-bit DPTR register and the Accumulator. Typically, DPTR is set up with the address of a jump table. In a 5-way branch, for example, an integer 0 through 4 is loaded into the Accumulator. The code to be executed might be as follows:

```
MOV   DPTR,#JUMP TABLE
MOV   A,INDEX_NUMBER
RL    A
JMP   @A+DPTR
```

The RL A instruction converts the index number (0 through 4) to an even number on the range 0 through 8, because each entry in the jump table is 2 bytes long:

```
JUMP TABLE:
      AJMP    CASE 0
      AJMP    CASE 1
      AJMP    CASE 2
      AJMP    CASE 3
      AJMP    CASE 4
```

Table 7 shows a single "CALL addr" instruction, but there are two of them, LCALL and ACALL, which differ in the format in which the subroutine address is given to the CPU. CALL is a generic mnemonic which can be used if the programmer does not care which way the address is encoded.

The LCALL instruction uses the 16-bit address format, and the subroutine can be anywhere in the 64k Program Memory space. The ACALL instruction uses the 11-bit format, and the subroutine must be in the same 2k block as the instruction following the ACALL.

In any case, the programmer specifies the subroutine address to the assembler in the same way: as a label or as a 16-bit constant. The assembler will put the address into the correct format for the given instructions.

Subroutines should end with a RET instruction, which returns execution to the instruction following the CALL.

RETI is used to return from an interrupt service routine. The only difference between RET and RETI is that RETI tells the interrupt control system that the interrupt in progress is done. If there is no interrupt in progress at the time RETI is executed, then the RETI is functionally identical to RET.

Table 8 shows the list of conditional jumps available to the 80C51 user. All of these jumps specify the destination address by the relative offset method, and so are limited to a jump distance of –128 to +127 bytes from the instruction following the conditional jump instruction. Important to note, however, the user specifies to the assembler the actual destination address the same way as the other jumps: as a label or a 16-bit constant.

There is no Zero bit in the PSW. The JZ and JNZ instructions test the Accumulator data for that condition.

The DJNZ instruction (Decrement and Jump if Not Zero) is for loop control. To execute a loop N times, load a counter byte with N and terminate the loop with a DJNZ to the beginning of the loop, as shown below for N = 10.

```
        MOV         COUNTER,#10
LOOP: (begin loop)
          •
          •
          •
        (end loop)
        DJNZ        COUNTER,LOOP
        (continue)
```

The CJNE instruction (Compare and Jump if Not Equal) can also be used for loop control as in Figure 12. Two bytes are specified in the operand field of the instruction. The jump is executed only if the two bytes are not equal. In the example of Figure 12, the two bytes were data in R1 and the constant 2AH. The initial data in R1 was 2EH. Every time the loop was executed, R1 was decremented, and the looping was to continue until the R1 data reached 2AH.

Another application of this instruction is in "greater than, less than" comparisons. The two bytes in the operand field are taken as unsigned integers. If the first is less than the second, then the Carry bit is set (1). If the first is greater than or equal to the second, then the Carry bit is cleared.

## CPU Timing

All 80C51 microcontrollers have an on-chip oscillator which can be used if desired as the clock source for the CPU. To use the on-chip oscillator, connect a crystal or ceramic resonator between the XTAL1 and XTAL2 pins of the microcontroller, and capacitors to ground as shown in Figure 13.

Examples of how to drive the clock with an external oscillator are shown in Figure 14. Note that in the NMOS devices (8051, etc.) the signal at the XTAL2 pin actually drives the internal clock generator. In the CMOS devices (80C51, etc.), the signal at the XTAL1 pin drives the internal clock generator. The internal clock generator defines the sequence of states that make up the 80C51 machine cycle.

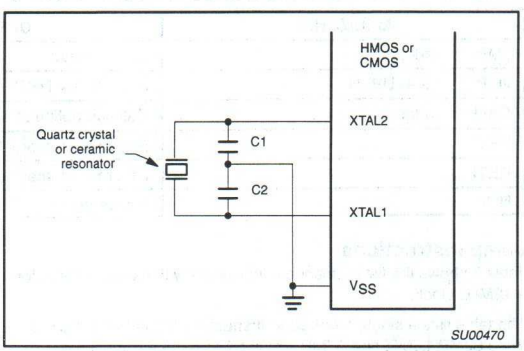



**Figure 13. Using the On-Chip Oscillator**

**Table 8. Conditional Jumps in 80C51 Devices**

| MNEMONIC | OPERATION | ADDRESSING MODES | | | | EXECUTION |
|---|---|---|---|---|---|---|
| | | DIR | IND | REG | IMM | TIME (μs) |
| JZ rel | Jump if A = 0 | Accumulator only | | | | 2 |
| JNZ rel | Jump if A ≠ 0 | Accumulator only | | | | 2 |
| DJNZ <byte>,rel | Decrement and jump if not zero | X | | X | | 2 |
| CJNE A,<byte>,rel | Jump if A ≠ <byte> | X | | | X | 2 |
| CJNE <byte>,#data,rel | Jump if <byte> ≠ #data | | X | X | | 2 |

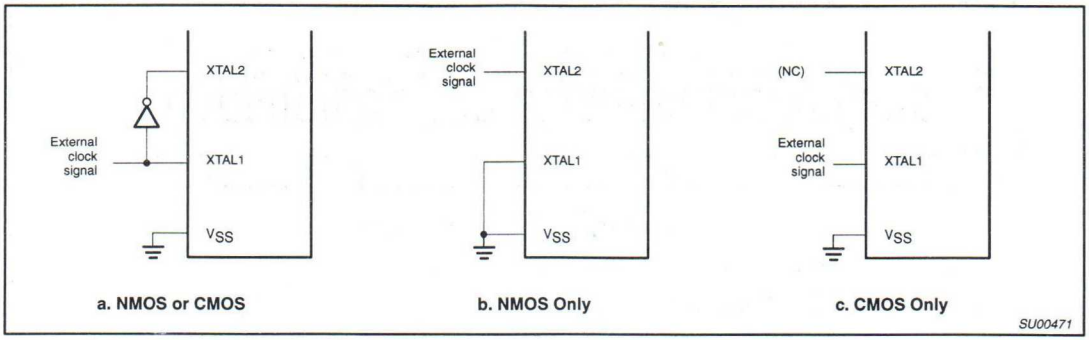



**Figure 14. Using an External Clock**

## Machine Cycles

A machine cycle consists of a sequence of 6 states, numbered S1 through S6. Each state time lasts for two oscillator periods. Thus a machine cycle takes 12 oscillator periods or 1µs if the oscillator frequency is 12MHz.

Each state is divided into a Phase 1 half and a Phase 2 half. Figure 15 shows that fetch/execute sequences in states and phases for various kinds of instructions. Normally two program fetches are generated during each machine cycle, even if the instruction being executed doesn't require it. If the instruction being executed doesn't need more code bytes, the CPU simply ignores the extra fetch, and the Program Counter is not incremented.

Execution of a one-cycle instruction (Figures 15a and 15b) begins during State 1 of the machine cycle, when the opcode is latched into the Instruction Register. A second fetch occurs during S4 of the same machine cycle. Execution is complete at the end of State 6 of this machine cycle.

The MOVX instructions take two machine cycles to execute. No program fetch is generated during the second cycle of a MOVX instruction. This is the only time program fetches are skipped. The fetch/execute sequence for MOVX instructions is shown in Figure 15d.

The fetch/execute sequences are the same whether the Program Memory is internal or external to the chip. Execution times do not depend on whether the Program Memory is internal or external.

Figure 16 shows the signals and timing involved in program fetches when the Program Memory is external. If Program Memory is external, then the Program Memory read strobe $\overline{\text{PSEN}}$ is normally activated twice per machine cycle, as shown in Figure 16a. If an access to external Data Memory occurs, as shown in Figure 16b, two $\overline{\text{PSEN}}$s are skipped, because the address and data bus are being used for the Data Memory access.

Note that a Data Memory bus cycle takes twice as much time as a Program Memory bus cycle. Figure 16 shows the relative timing of the addresses being emitted at Ports 0 and 2, and of ALE and $\overline{\text{PSEN}}$. ALE is used to latch the low address byte from P0 into the address latch.

When the CPU is executing from internal Program Memory, PSEN is not activated, and program addresses are not emitted. However, ALE continues to be activated twice per machine cycle and so it is available as a clock output signal. Note, however, that one ALE is skipped during the execution of the MOVX instruction.

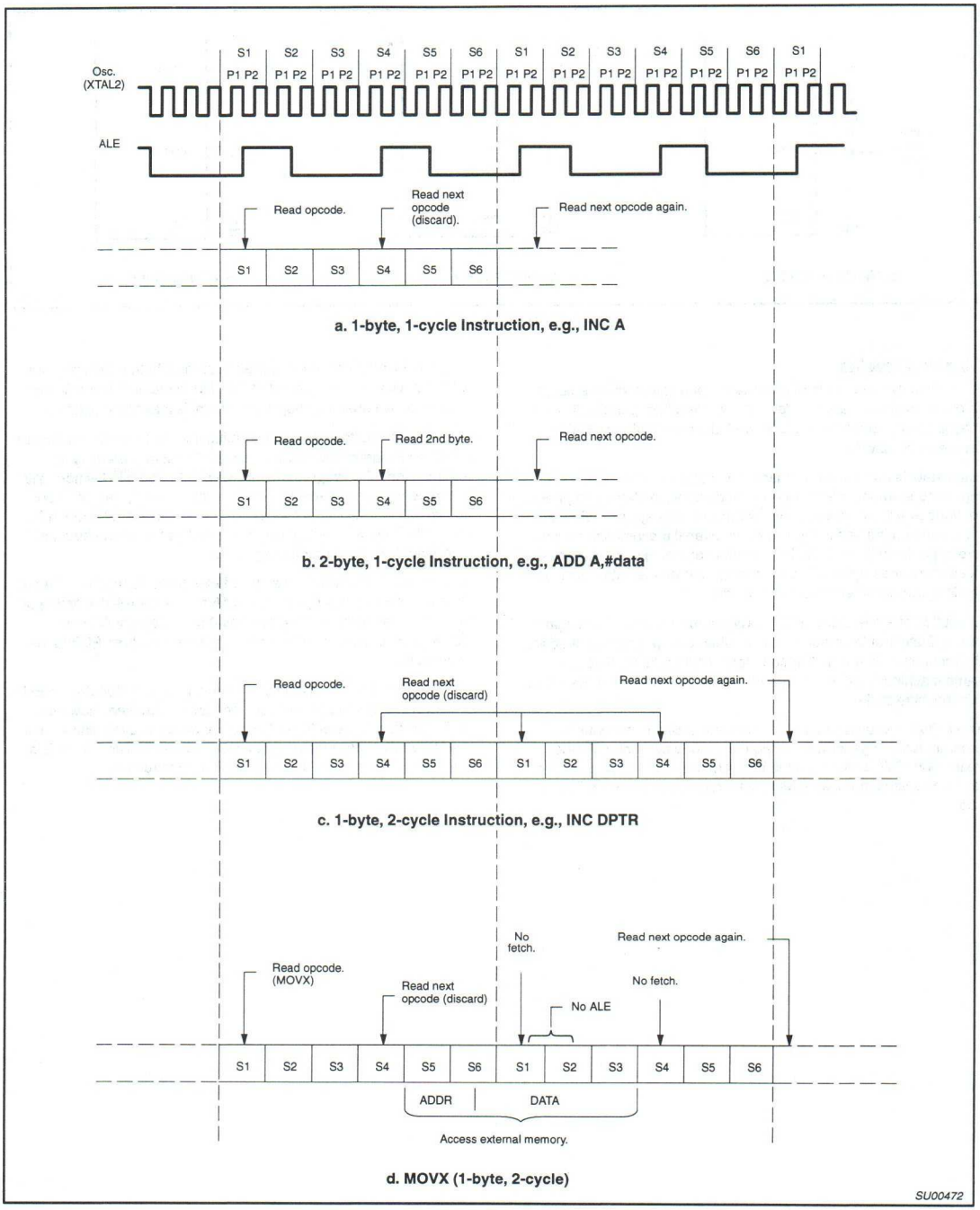



**Figure 15. State Sequence in 80C51 Family Devices**

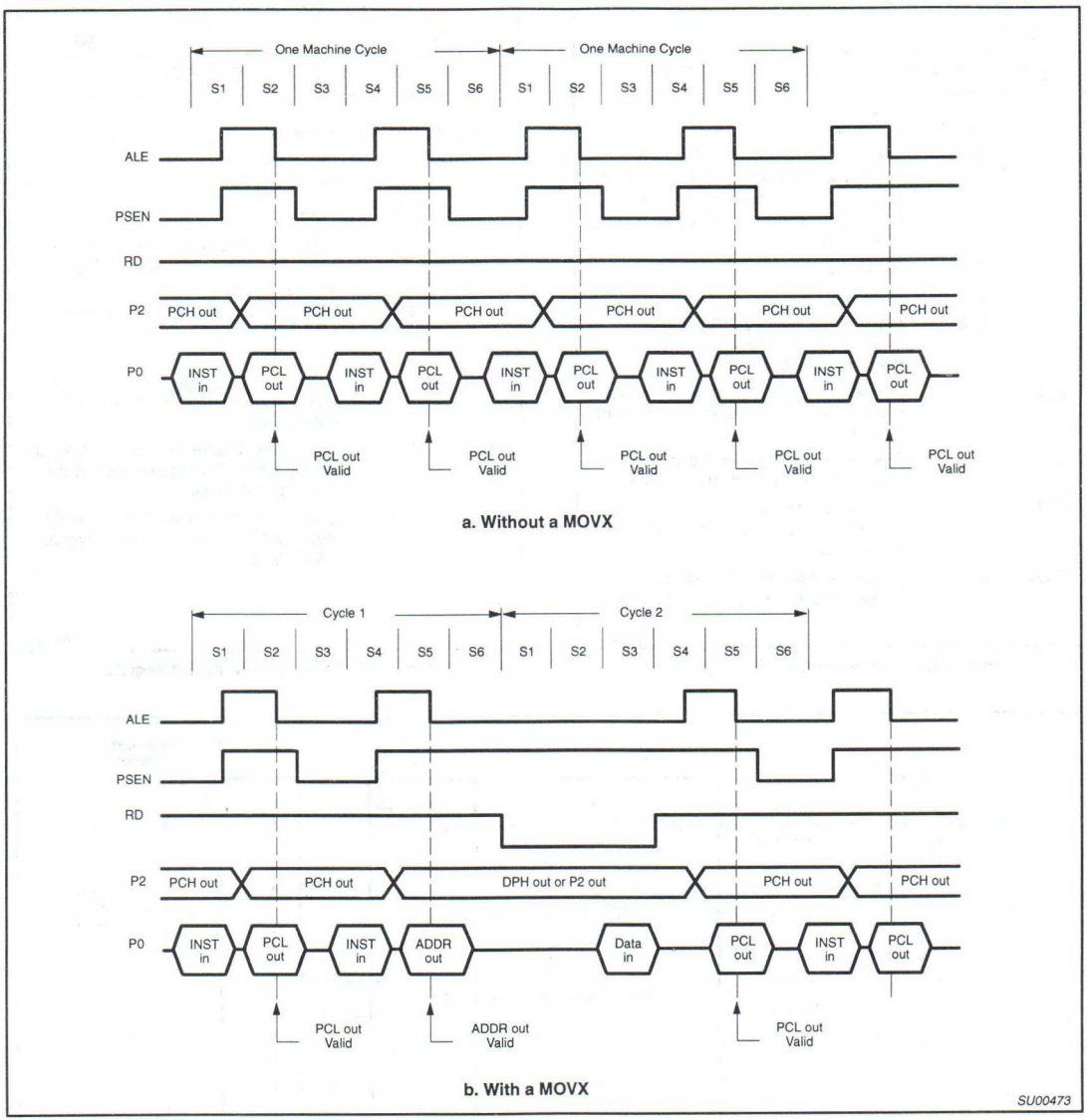



**Figure 16. Bus Cycles in 80C51 Family Devices Executing from External Program Memory**

(MSB) (LSB)

| $\overline{EA}$ | X | X | ES | ET1 | EX1 | ET0 | EX0 |
|---|---|---|---|---|---|---|---|

| Symbol | Position | Function |
|---|---|---|
| $\overline{EA}$ | IE.7 | Disables all interrupts. If $\overline{EA}$ = 0, no interrupt will be acknowledged. If $\overline{EA}$ = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit. |
| | IE.6 | Reserved. |
| | IE.5 | Reserved. |
| ES | IE.4 | Enables or disables the Serial Port interrupt. If ES = 0, the Serial Port interrupt is disabled. |
| ET1 | IE.3 | Enables or disables the Timer 1 Overflow interrupt. If ET1 = 0, the Timer 1 interrupt is disabled. |
| EX1 | IE.2 | Enables or disables External Interrupt 1. If EX1 = 0, External Interrupt 1 is disabled. |
| ET0 | IE.1 | Enables or disables the Timer 0 Overflow interrupt. If ET0 = 0, the Timer 0 interrupt is disabled. |
| EX0 | IE.0 | Enables or disables Exeternal Interrupt 0. If EX0 = 0, External Interrupt 0 is disabled. |

SU00474

**Figure 17. Interrupt Enable (IE) Register**

(MSB) (LSB)

| X | X | X | PS | PT1 | PX1 | PT0 | PX0 |
|---|---|---|---|---|---|---|---|

| Symbol | Position | Function |
|---|---|---|
| | IP.7 | Reserved. |
| | IP.6 | Reserved. |
| | IP.5 | Reserved. |
| PS | IP.4 | Defines the Serial Port interrupt priority level. PS = 1 programs it to the higher priority level. |
| PT1 | IP.3 | Defines the Timer 1 interrupt priority level. PT1 = 1 programs it to the higher priority level. |
| PX1 | IP.2 | Defines the External Interrupt 1 priority level. PX1 = 1 programs it to the higher priority level. |
| PT0 | IP.1 | Enables or disables the Timer 0 Interrupt priority level. PT) = 1 programs it to the higher priority level. |
| PX0 | IP.0 | Defines the External Interrupt 0 priority level. PX0 = 1 programs it to the higher priority level. |

SU00475

**Figure 18. Interrupt Priority (IP) Register**

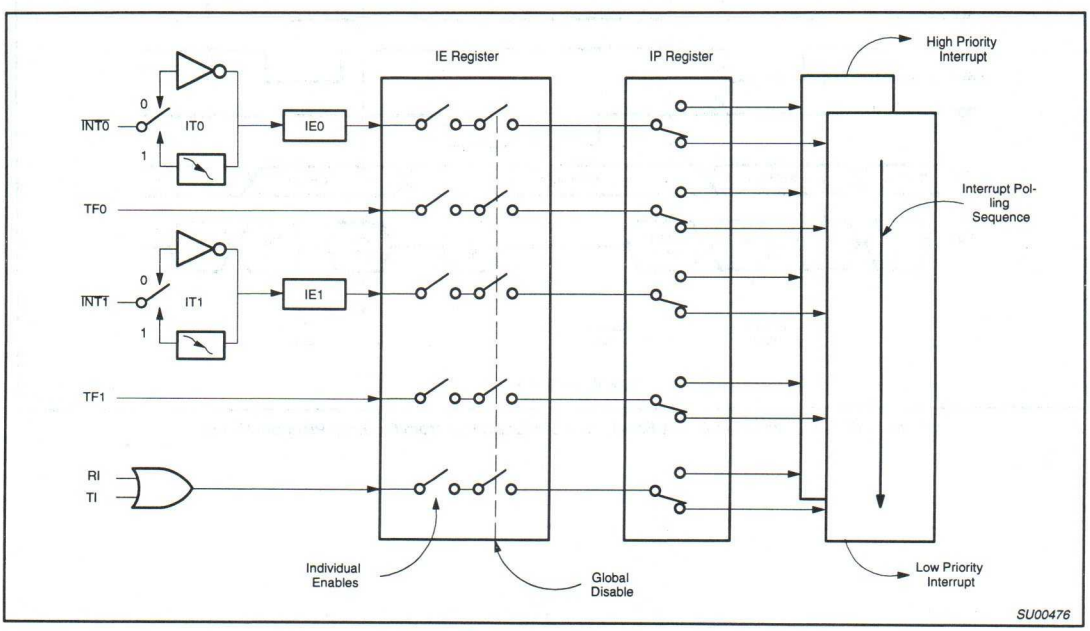



**Figure 19. Interrupt Control System**

## Interrupt Structure

The 80C51 and its ROMless and EPROM versions have 5 interrupt sources: 2 external interrupts, 2 timer interrupts, and the serial port interrupt.

What follows is an overview of the interrupt structure for the device. More detailed information for specific members of the 80C51 derivative family is provided in later chapters of this user's guide.

### Interrupt Enables

Each interrupt source can be individually enabled or disabled by setting or clearing a bit in the SFR named IE (Interrupt Enable). This register also contains a global disable bit, which can be cleared to disable all interrupts at once. Figure 17 shows the IE register.

### Interrupt Priorities

Each interrupt source can also be individually programmed to one of two priority levels by setting or clearing a bit in the SFR named IP (Interrupt Priority). Figure 18 shows the IP register. A low-priority interrupt can be interrupted by a high-priority interrupt, but not by another low-priority interrupt. A high-priority interrupt can't be interrupted by any other interrupt source.

If two interrupt requests of different priority levels are received simultaneously, the request of higher priority is serviced. If interrupt requests of the same priority level are received simultaneously, an internal polling sequence determines which request is serviced. Thus within each priority level there is a second priority structure determined by the polling sequence. Figure 19 shows how the IE and IP registers and the polling sequence work to determine which if any interrupt will be serviced.

In operation, all the interrupt flags are latched into the interrupt control system during State 5 of every machine cycle. The samples are polled during the following machine cycle. If the flag for an enabled interrupt is found to be set (1), the interrupt system generates an LCALL to the appropriate location in Program Memory, unless some other condition blocks the interrupt. Several conditions can block an interrupt, among them that an interrupt of equal or higher priority level is already in progress.

The hardware-generated LCALL causes the contents of the Program Counter to be pushed into the stack, and reloads the PC with the beginning address of the service routine. As previously noted (Figure 3), the service routine for each interrupt begins at a fixed location.

Only the Program Counter is automatically pushed onto the stack, not the PSW or any other register. Having only the PC automatically saved allows the programmer to decide how much time should be spent saving other registers. This enhances the interrupt response time, albeit at the expense of increasing the programmer's burden of responsibility. As a result, many interrupt functions that are typical in control applications toggling a port pin for example, or reloading a timer, or unloading a serial buffer can often be completed in less time than it takes other architectures to complete.

### Simulating a Third Priority Level in Software

Some applications require more than two priority levels that are provided by on-chip hardware in 80C51 devices. In these cases, relatively simple software can be written to produce the same effect as a third priority level. First, interrupts that are to have higher priority than 1 are assigned to priority 1 in the Interrupt Priority (IP) register. The service routines for priority 1 interrupts that are supposed to be interruptable by priority 2 interrupts are written to include the following code:

```
        PUSH   IE
        MOV    IE,#MASK
        CALL   LABEL
        ************************
        (execute service routine)
        ************************
        POP    IE
        RET
LABEL:  RETI
```

As soon as any priority interrupt is acknowledged, the Interrupt Enable (IE) register is redefined so as to disable all but priority 2 interrupts. Then a CALL to LABEL executes the RETI instruction, which clears the priority 1 interrupt-in-progress flip-flop. At this point any priority 1 interrupt that is enabled can be serviced, but only priority 2 interrupts are enabled.

POPing IE restores the original enable byte. Then a normal RET (rather than another RETI) is used to terminate the service routine. The additional software adds 10µs (at 12MHz) to priority 1 interrupts.

## HARDWARE DESCRIPTION

This chapter provides a detailed description of the 80C51 microcontroller (see Figure 1). Included in this description are:

- The port drivers and how they function both as ports and, for Ports 0 and 2, in bus operations
- The Timers/Counters
- The Serial Interface
- The Interrupt System
- Reset
- The Reduced Power Modes in CMOS devices
- The EPROM version of the 80C51

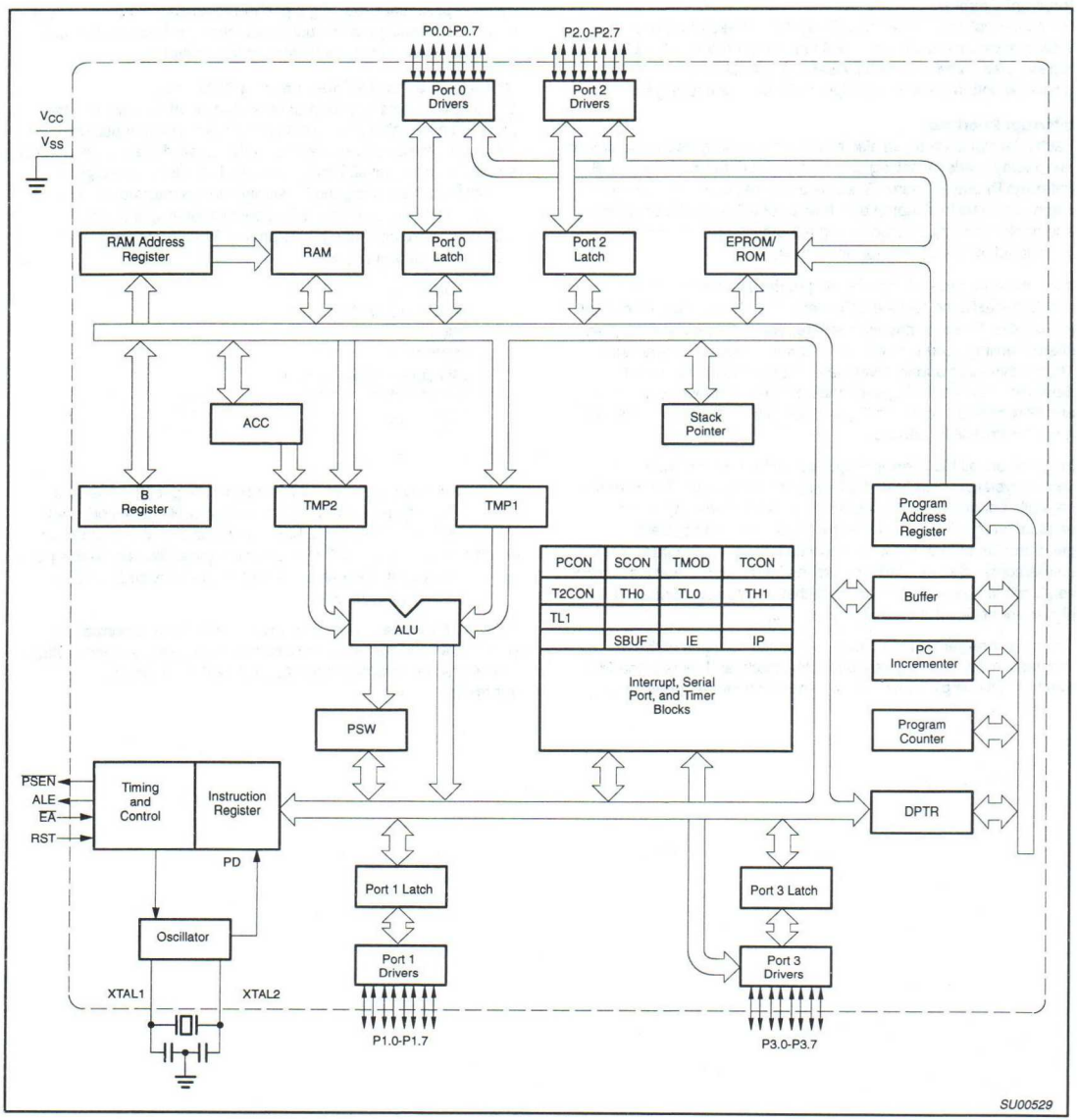



**Figure 1. 80C51 Architecture**

## Special Function Registers

A Map of the on-chip memory area called the Special Function Register (SFR) space is shown in Figure 2.

Note that in the SFRs not all of the addresses are occupied. Unoccupied addresses are not implemented on the chip. Read accesses to these addresses will in general return random data, and write accesses will have no effect.

User software should not write 1s to these unimplemented locations, since they may be used in other 80C51 Family derivative products to invoke new features. The functions of the SFRs are described in the text that follows.

**Accumulator**
ACC is the Accumulator register. The mnemonics for Accumulator-Specific instructions, however, refer to the Accumulator simply as A.

**B Register**
The B register is used during multiply and divide operations. For other instructions it can be treated as another scratch pad register.

**Program Status Word**
The PSW register contains program status information as detailed in Figure 3.

**Stack Pointer**
The Stack Pointer register is 8 bits wide. It is incremented before data is stored during PUSH and CALL executions. While the stack may reside anywhere in on-chip RAM, the Stack Pointer is initialized to 07H after a reset. This causes the stack to begin at locations 08H.

**Data Pointer**
The Data Pointer (DPTR) consists of a high byte (DPH) and a low byte (DPL). Its intended function is to hold a 16-bit address. It may be manipulated as a 16-bit register or as two independent 8-bit registers.

**Ports 0 to 3**
P0, P1, P2, and P3 are the SFR latches of Ports 0, 1, 2, and 3, respectively. Writing a one to a bit of a port SFR (P0, P1, P2, or P3) causes the corresponding port output pin to switch high. Writing a zero causes the port output pin to switch low. When used as an input, the external state of a port pin will be held in the port SFR (i.e., if the external state of a pin is low, the corresponding port SFR bit will contain a 0; if it is high, the bit will contain a 1).

**Serial Data Buffer**
The Serial Buffer is actually two separate registers, a transmit buffer and a receive buffer. When data is moved to SBUF, it goes to the transmit buffer and is held for serial transmission. (Moving a byte to SBUF is what initiates the transmission.) When data is moved from SBUF, it comes from the receive buffer.

**Timer Registers Basic to 80C51**
Register pairs (TH0, TL0), and (TH1, TL1) are the 16-bit Counting registers for Timer/Counters 0 and 1, respectively.

**Control Register for the 80C51**
Special Function Registers IP, IE, TMOD, TCON, SCON, and PCON contain control and status bits for the interrupt system, the Timer/Counters, and the serial port. They are described in later sections.

## Port Structures and Operation

All four ports in the 80C51 are bidirectional. Each consists of a latch (Special Function Registers P0 through P3), an output driver, and an input buffer.

The output drivers of Ports 0 and 2, and the input buffers of Port 0, are used in accesses to external memory. In this application, Port 0 outputs the low byte of the external memory address, time-multiplexed with the byte being written or read.

Port 2 outputs the high byte of the external memory address when the address is 16 bits wide. Otherwise, the Port 2 pins continue to emit the P2 SFR content.

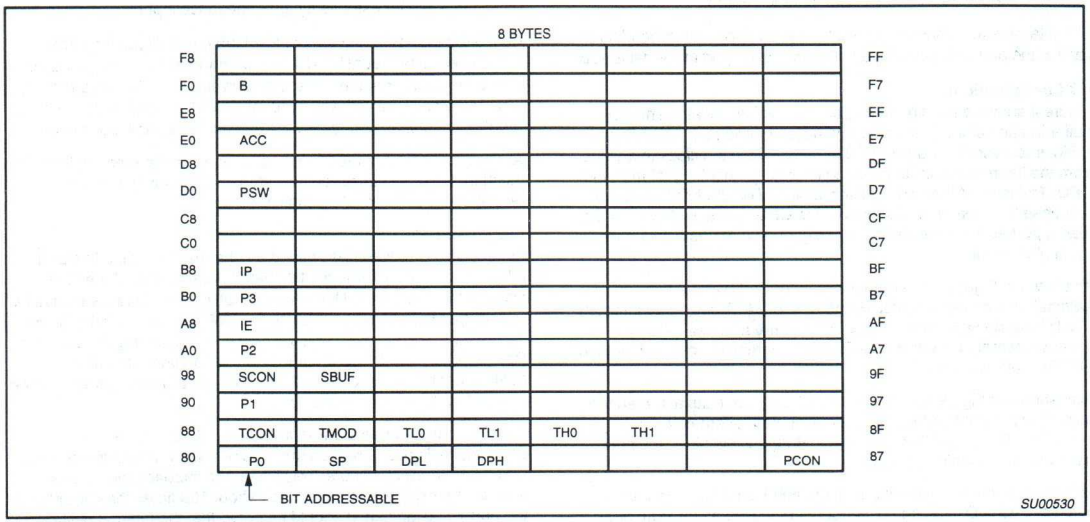



**Figure 2. 80C51 SFR Memory Map**

MSB LSB

| CY | AC | F0 | RS1 | RS0 | OV | — | P |
|---|---|---|---|---|---|---|---|

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| PSW.7 | CY | Carry flag. |
| PSW.6 | AC | Auxilliary Carry flag. (For BCD operations.) |
| PSW.5 | F0 | Flag 0. (Available to the user for general purposes.) |
| PSW.4 | RS1 | Register bank select control bit 1.<br>Set/cleared by software to determine working register bank. (See Note.) |
| PSW.3 | RS0 | Register bank select control bit 0.<br>Set/cleared by software todetermine working register bank. (See Note.) |
| PSW.2 | OV | Overflow flag. |
| PSW.1 | — | User-definable flag. |
| PSW.0 | P | Parity flag.<br>Set/cleared by hardware each instruction cycle to indicate an odd/even number of "one" bits in the Accumulator, i.e., even parity. |

NOTE: The contents of (RS1, RS0) enable the working register banks as follows:
(0,0)— Bank 0 (00H–07H)
(0,1)— Bank 1 (08H–0fH)
(1,0)— Bank 2 (10H–17H)
(1,1)— Bank 3 (18H–17H)

SU00531A

**Figure 3. Program Status Word (PSW) Register**

All the Port 3 pins are multifunctional. They are not only port pins, but also serve the functions of various special features as listed below:

| Port Pin | Alternate Function |
|---|---|
| P3.0 | RxD (serial input port) |
| P3.1 | TxD (serial output port) |
| P3.2 | $\overline{INT0}$ (external interrupt) |
| P3.3 | $\overline{INT1}$ (external interrupt) |
| P3.4 | T0 (Timer/Counter 0 external input) |
| P3.5 | T1 (Timer/Counter 1 external input) |
| P3.6 | $\overline{WR}$ (external Data Memory write strobe) |
| P3.7 | $\overline{RD}$ (external Data Memory read strobe) |

The alternate functions can only be activated if the corresponding bit latch in the port SFR contains a 1. Otherwise the port pin remains at 0.

**I/O Configurations**

Figure 4 shows a functional diagram of a typical bit latch and I/O buffer in each of the four ports. The bit latch (one bit in the port's SFR) is represented as a Type D flip-flop, which will clock in a value from the internal bus in response to a "write to latch" signal from the CPU. The level of the port pin itself is placed on the internal bus in response to a "read pin" signal from the CPU. Some instructions that read a port activate the "read latch" signal, and others activate the "read pin" signal.

As shown in Figure 4, the output drivers of Port 0 and 2 are switchable to an internal ADDR and ADDR/DATA bus by an internal CONTROL signal for use in external memory accesses. During external memory accesses, the P2 SFR remains unchanged, but the P0 SFR gets 1s written to it.

Also shown in Figure 4 is that if a P3 bit latch contains a 1, then the output level is controlled by the signal labeled "alternate output function." The actual P3.X pin level is always available to the pin's alternate input function, if any.

Ports 1, 2, and 3 have internal pullups, and Port 0 has open drain outputs. Each I/O line can be independently used as an input or an output. (Port 0 and 2 may not be used as general purpose I/O when being used as the ADDR/DATA BUS for external memory during normal operation.) To be used as an input, the port bit latch must contain a 1, which turns off the output driver FET. Then, for Ports 1, 2, and 3, the pin is pulled high by a weak internal pullup, and can be pulled low by an external source.

Port 0 differs in that its internal pullups are not active during normal port operation. The pullup FET in the P0 output driver (see Figure 4) is used only when the port is emitting 1s during external memory accesses. Otherwise the pullup FET is off. Consequently P0 lines that are being used as output port lines are open drain. Writing a 1 to the bit latch leaves both output FETs off, so the pin floats. In that condition it can be used as a high-impedance input.

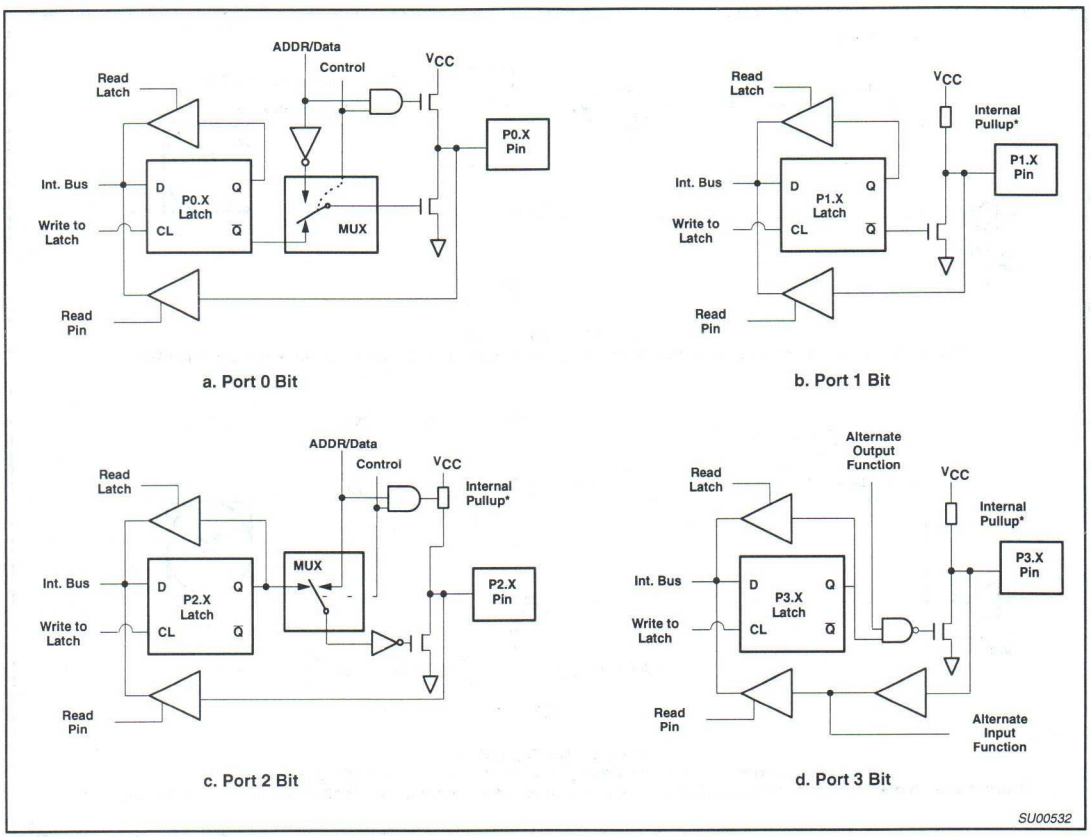
Because Ports 1, 2, and 3 have fixed internal pullups, they are sometimes called "quasi- bidirectional" ports. When configured as inputs they pull high and will source current ($I_{IL}$, in the data sheets) when externally pulled low. Port 0, on the other hand, is considered "true" bidirectional, because when configured as an input it floats.

All the port latches in the 80C51 have 1s written to them by the reset function. If a 0 is subsequently written to a port latch, it can be reconfigured as an input by writing a 1 to it.

**Writing to a Port**

In the execution of an instruction that changes the value in a port latch, the new value arrives at the latch during S6P2 of the final cycle of the instruction. However, port latches are in fact sampled by their output buffers only during Phase 1 of an clock period. (During Phase 2 the output buffer holds the value it saw during the previous Phase 1). Consequently, the new value in the port latch won't actually appear at the output pin until the next Phase 1, which will be at S1P1 of the next machine cycle.

If the change requires a 0-to-1 transition in Port 1, 2, or 3, an additional pullup is turned on during S1P1 and S1P2 of the cycle in which the transition occurs. This is done to increase the transition speed. The extra pullup can source about 100 times the current that the normal pullup can. It should be noted that the internal pullups are field-effect transistors, not linear resistors. The pullup arrangements are shown in Figure 5.




*See Figure 5 for details of the internal pullup.

**Figure 4. 80C51 Port Bit Latches and I/O Buffers**

In the NMOS 8051 part, the fixed part of the pullup is a depletion mode transistor with the gate wired to the source. This transistor will allow the pin to source about 0.25mA when shorted to ground. In parallel with the fixed pullup is an enhancement mode transistor, which is activated during S1 whenever the port bit does a 0-to-1 transition. During this interval, if the port pin is shorted to ground, this extra transistor will allow the pin to source an additional 30mA.

In the CMOS 80C51, the pullup consists of three pFETs. It should be noted that an n-channel FET (nFET) is turned on when a logical 1 is applied to its gate, and is turned off when a logical 0 is applied to its gate. A p-channel FET (pFET) is the opposite: it is on when its gate sees a 0, and off when its gate sees a 1.

pFET1 in Figure 5 is the transistor that is turned on for 2 oscillator periods after a 0-to-1 transition in the port latch. While it's on, it turns on pFET3 (a weak pullup), through the inverter. This inverter and pFET3 form a latch which holds the 1.

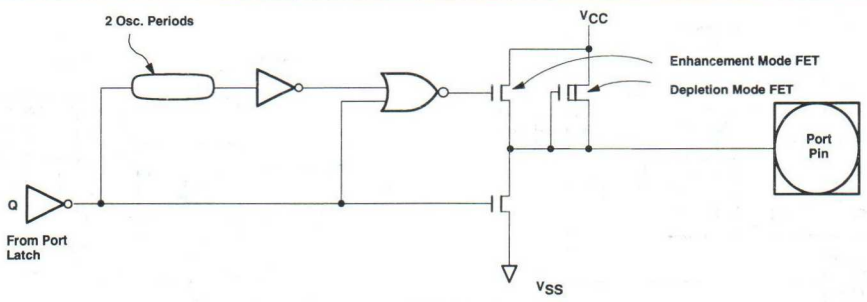
Note that if the pin is emitting a 1, a negative glitch on the pin from some external source can turn off pFET3, causing the pin to go into a float state. pFET2 is a very weak pullup which is on whenever the nFET is off, in traditional CMOS style. It's only about 1/10 the strength of pFET1. Its function is to restore a 1 to the pin in the event the pin had a 1 and lost it to a glitch.

**Port Loading and Interfacing**

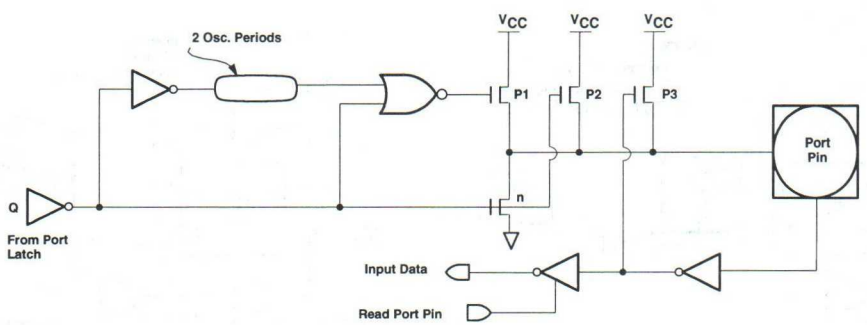
The output buffers of Ports 1, 2, and 3 can each drive 4 LS TTL inputs. These ports on NMOS versions can be driven in a normal manner by a TTL or NMOS circuit. Both NMOS and CMOS pins can be driven by open-collector and open-drain outputs, but note that 0-to-1 transitions will not be fast.

In the NMOS device, if the pin is driven by an open-collector output, a 0-to-1 transition will have to be driven by the relatively weak depletion mode FET in Figure 5a. In the CMOS device, an input 0 turns off pullup pFET3, leaving only the very weak pullup pFET2 to drive the transition.

Port 0 output buffers can each drive 8 LS TTL inputs. They do, however, require external pullups to drive NMOS inputs, except when being used as the ADDRESS/DATA bus for external memory.




**a. NMOS Configuration.**
**The enhancement mode transistor is turned on for 2 oscillator periods after Q makes a 0-to-1 transition.**




**b. CMOS Configuration.**
**pFET1 is turned on for 2 oscillator periods after Q makes a 0-to-1 transition.**
**During this time, pFET1 also turns on pFET3 through the inverter to form a latch which holds the 1. pFET2 is also on.**

SU00533

**Figure 5. Ports 1 and 3 NMOS and CMOS Internal Pullup Configurations**
(Port 2 is similar except that it holds the strong pullup on while emitting 1s that are address bits. See *Accessing External Memory.*)

**Read-Modify-Write Feature**

Some instructions that read a port read the latch and others read the pin. Which ones do which? The instructions that read the latch rather than the pin are the ones that read a value, possibly change it, and then rewrite it to the latch. These are called "read-modify-write" instructions. The instructions listed below are read-modify-write instructions. When the destination operand is a port, or a port bit, these instructions read the latch rather than the pin:

| | |
|---|---|
| ANL | (logical AND, e.g., ANL P1,A) |
| ORL | (logical OR, e.g., ORL P2,A) |
| XRL | (logical EX-OR, e.g., XRL P3,A) |
| JBC | (jump if bit = 1 and clear bit, e.g., JBC P1.1,LABEL) |
| CPL | (complement bit, e.g., CPL P3.0) |
| INC | (increment, e.g., INC P2) |
| DEC | (decrement, e.g., DEC P2) |
| DJNZ | (decrement and jump if not zero, e.g., DJNZ P3,LABEL) |
| MOV PX.Y,C | (move carry bit to bit Y of Port X) |
| CLR PX.Y | (clear bit Y of Port X) |
| SET PX.Y | (set bit Y of Port X) |

It is not obvious that the last three instructions in this list are read-modify-write instructions, but they are. They read the port byte, all 8 bits, modify the addressed bit, then write the new byte back to the latch.

The reason that read-modify-write instructions are directed to the latch rather than the pin is to avoid a possible misinterpretation of the voltage level at the pin. For example, a port bit might be used to drive the base of a transistor. When a 1 is written to the bit, the transistor is turned on. If the CPU then reads the same port bit at the pin rather than the latch, it will read the base voltage of the transistor and interpret it as a 0. Reading the latch rather than the pin will return the correct value of 1.

## Accessing External Memory

Accesses to external memory are of two types: accesses to external Program Memory and accesses to external Data Memory. Accesses to external Program Memory use signal $\overline{PSEN}$ (program store enable) as the read strobe. Accesses to external Data Memory use $\overline{RD}$ or $\overline{WR}$ (alternate functions of P3.7 and P3.6) to strobe the memory. Fetches from external Program Memory always use a 16-bit address. Accesses to external Data Memory can use either a 16-bit address (MOVX @ DPTR) or an 8-bit address (MOVX @Ri).

Whenever a 16-bit address is used, the high byte of the address comes out on Port 2, where it is held for the duration of the read or write cycle. Note that the Port 2 drivers use the strong pullups during the entire time that they are emitting address bits that are 1s. This is during the execution of a MOVX @DPTR instruction. During this time the Port 2 latch (the Special Function Register) does not have to contain 1s, and the contents of the Port 2 SFR are not modified. If the external memory cycle is not immediately followed by another external memory cycle, the undisturbed contents of the Port 2 SFR will reappear in the next cycle.

If an 8-bit address is being used (MOVX @Ri), the contents of the Port 2 SFR remain at the Port 2 pins throughout the external memory cycle. This will facilitate paging.

In any case, the low byte of the address is time-multiplexed with the data byte on Port 0. The ADDR/DATA signals drive both FETs in the Port 0 output buffers. Thus, in this application the Port 0 pins are not open-drain outputs, and do not require external pullups. ALE (Address Latch Enable) should be used to capture the address byte into an external latch. The address byte is valid at the negative transition of ALE. Then, in a write cycle, the data byte to be written appears on Port 0 just before $\overline{WR}$ is activated, and remains there until after $\overline{WR}$ is deactivated. In a read cycle, the incoming byte is accepted at Port 0 just before the read strobe is deactivated.

During any access to external memory, the CPU writes 0FFH to the Port 0 latch (the Special Function Register), thus obliterating whatever information the Port 0 SFR may have been holding.

External Program Memory is accessed under two conditions: Whenever signal $\overline{EA}$ is active; or whenever the program counter (PC) contains a number that is larger than 0FFFH (in the 80C51).

This require that the ROMless versions have $\overline{EA}$ wired low to enable the lower 4k program bytes to be fetched from external memory.

When the CPU is executing out of external Program Memory, all 8 bits of Port 2 are dedicated to an output function and may not be used for general purpose I/O. During external program fetches they output the high byte of the PC. During this time the Port 2 drivers use the strong pullups to emit PC bits that are 1s.

## Timer/Counters

The 80C51 has two 16-bit Timer/Counter registers: Timer 0 and Timer 1. Both can be configured to operate either as timers or event counters (see Figure 6).

In the "Timer" function, the register is incremented every machine cycle. Thus, one can think of it as counting machine cycles. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

In the "Counter" function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T0 or T1. In this function, the external input is sampled during S5P2 of every machine cycle.

When the samples show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register during S3P1 of the cycle following the one in which the transition was detected. Since it takes 2 machine cycles (24 oscillator periods) to recognize a 1-to-0 transition, the maximum count rate is 1/24 of the oscillator frequency. There are no restrictions on the duty cycle of the external input signal, but to ensure that a given level is sampled at least once before it changes, it should be held for at least one full cycle. In addition to the "Timer" or "Counter" selection, Timer 0 and Timer 1 have four operating modes from which to select.

### Timer 0 and Timer 1

The "Timer" or "Counter" function is selected by control bits C/T in the Special Function Register TMOD. These two Timer/Counters have four operating modes, which are selected by bit-pairs (M1, M0) in TMOD. Modes 0, 1, and 2 are the same for both Timers/Counters. Mode 3 is different. The four operating modes are described in the following text.

### Mode 0

Putting either Timer into Mode 0 makes it look like an 8048 Timer, which is an 8-bit Counter with a divide-by-32 prescaler. Figure 7 shows the Mode 0 operation as it applies to Timer 1.

In this mode, the Timer register is configured as a 13-bit register. As the count rolls over from all 1s to all 0s, it sets the Timer interrupt flag TF1. The counted input is enabled to the Timer when TR1 = 1 and either GATE = 0 or $\overline{INT1}$ = 1. (Setting GATE = 1 allows the Timer to be controlled by external input $\overline{INT1}$, to facilitate pulse width measurements). TR1 is a control bit in the Special Function Register TCON (Figure 8). GATE is in TMOD.

The 13-bit register consists of all 8 bits of TH1 and the lower 5 bits of TL1. The upper 3 bits of TL1 are indeterminate and should be ignored. Setting the run flag (TR1) does not clear the registers.

Mode 0 operation is the same for the Timer 0 as for Timer 1. Substitute TR0, TF0, and $\overline{INT0}$ for the corresponding Timer 1 signals in Figure 7. There are two different GATE bits, one for Timer 1 (TMOD.7) and one for Timer 0 (TMOD.3).

### Mode 1

Mode 1 is the same as Mode 0, except that the Timer register is being run with all 16 bits.

### Mode 2

Mode 2 configures the Timer register as an 8-bit Counter (TL1) with automatic reload, as shown in Figure 9. Overflow from TL1 not only sets TF1, but also reloads TL1 with the contents of TH1, which is preset by software. The reload leaves TH1 unchanged.

Mode 2 operation is the same for Timer/Counter 0.

### Mode 3

Timer 1 in Mode 3 simply holds its count. The effect is the same as setting TR1 = 0.

Timer 0 in Mode 3 establishes TL0 and TH0 as two separate counters. The logic for Mode 3 on Timer 0 is shown in Figure 10. TL0 uses the Timer 0 control bits: C/T, GATE, TR0, INT0, and TF0. TH0 is locked into a timer function (counting machine cycles) and takes over the use of TR1 and TF1 from Timer 1. Thus, TH0 now controls the "Timer 1" interrupt.

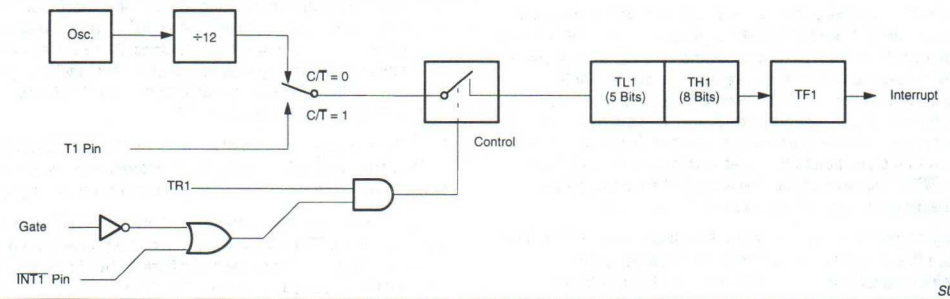
Mode 3 is provided for applications requiring an extra 8-bit timer on the counter. With Timer 0 in Mode 3, an 80C51 can look like it has three Timer/Counters. When Timer 0 is in Mode 3, Timer 1 can be turned on and off by switching it out of and into its own Mode 3, or can still be used by the serial port as a baud rate generator, or in fact, in any application not requiring an interrupt.




| | | |
|---|---|---|
| | GATE | Gating control when set. Timer/Counter "x" is enabled only while "INTx" pin is high and "TRx" control pin is set. when cleared Timer "x" is enabled whenever "TRx" control bit is set. |
| | C/T | Timer or Counter Selector cleared for Timer operation (input from in=ternal system clock.) Set for Counter operation (input from "Tx" input pin). |
| **M1** | **M0** | **OPERATING** |
| 0 | 0 | 8048 Timer "TLx" serves as 5-bit prescaler. |
| 0 | 1 | 16-bit Timer/Counter "THx" and "TLx" are cascaded; there is no prescaler. |
| 1 | 0 | 8-bit auto-reload Timer/Counter "THx" holds a value which is to be reloaded into "TLx" each time it overflows. |
| 1 | 1 | (Timer 0) TL0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits. TH0 is an 8-bit timer only controlled by Timer 1 control bits. |
| 1 | 1 | (Timer 1) Timer/Counter 1 stopped. |

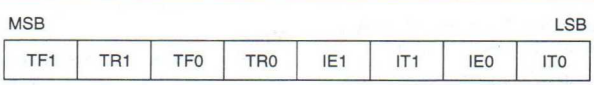
SU00534

**Figure 6. Timer/Counter Mode Control (TMOD) Register**




SU00535

**Figure 7. Timer/Counter Mode 0: 13-Bit Counter**




| **BIT** | **SYMBOL** | **FUNCTION** |
|---|---|---|
| TCON.7 | TF1 | Timer 1 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine, or clearing the bit in software. |
| TCON.6 | TR1 | Timer 1 Run control bit. Set/cleared by software to turn Timer/Counter on/off. |
| TCON.5 | TF0 | Timer 0 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine, or by clearing the bit in software. |
| TCON.4 | TR0 | Timer 0 Run control bit. Set/cleared by software to turn Timer/Counter on/off. |
| TCON.3 | IE1 | Interrupt 1 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed. |
| TCON.2 | IT1 | Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts. |
| TCON.1 | IE0 | Interrupt 0 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed. |
| TCON.0 | IT0 | Interrupt 0 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts. |

SU00536

**Figure 8. Timer/Counter Control (TCON) Register**

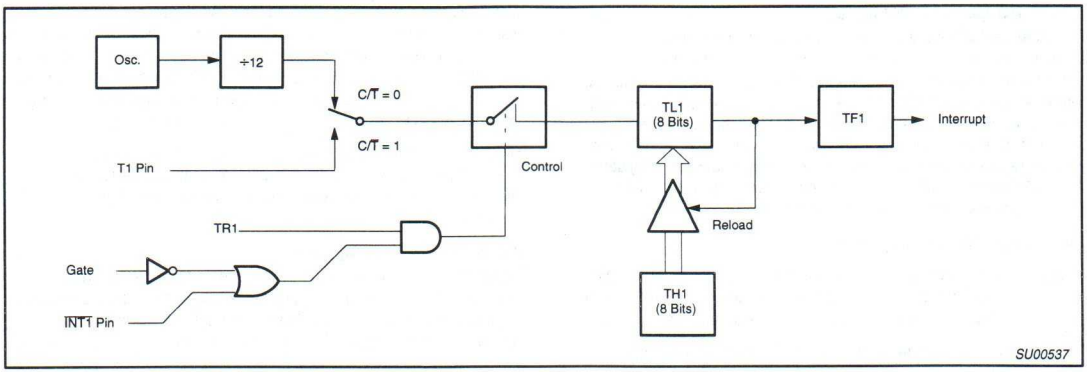



**Figure 9. Timer/Counter Mode 2: 8-Bit Auto-Load**

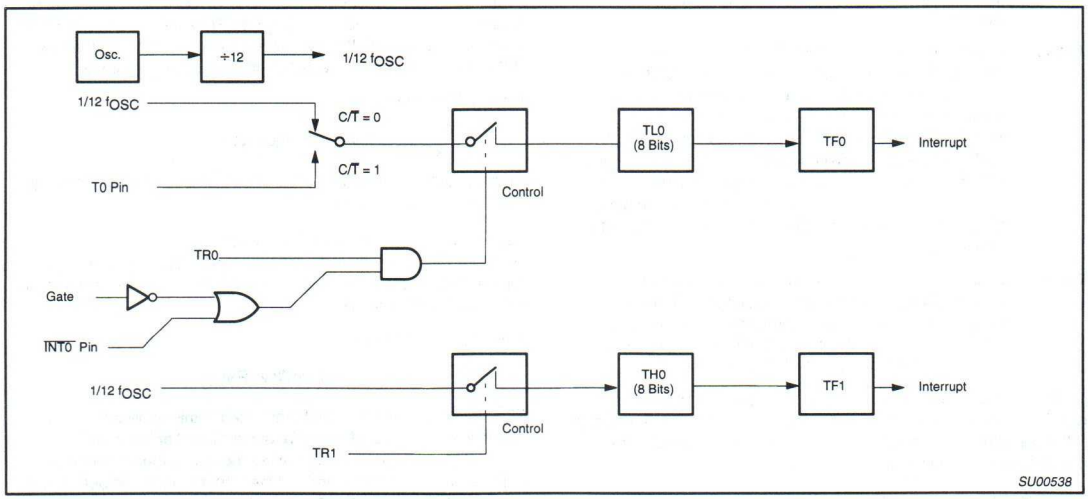



**Figure 10. Timer/Counter 0 Mode 3: Two 8-Bit Counters**

## Standard Serial Interface

The serial port is full duplex, meaning it can transmit and receive simultaneously. It is also receive-buffered, meaning it can commence reception of a second byte before a previously received byte has been read from the register. (However, if the first byte still hasn't been read by the time reception of the second byte is complete, one of the bytes will be lost.) The serial port receive and transmit registers are both accessed at Special Function Register SBUF. Writing to SBUF loads the transmit register, and reading SBUF accesses a physically separate receive register.

The serial port can operate in 4 modes:

**Mode 0:** Serial data enters and exits through RxD. TxD outputs the shift clock. 8 bits are transmitted/received (LSB first). The baud rate is fixed at 1/12 the oscillator frequency.

**Mode 1:** 10 bits are transmitted (through TxD) or received (through RxD): a start bit (0), 8 data bits (LSB first), and a stop bit (1). On receive, the stop bit goes into RB8 in Special Function Register SCON. The baud rate is variable.

**Mode 2:** 11 bits are transmitted (through TxD) or received (through RxD): start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). On Transmit, the 9th data bit (TB8 in SCON) can be assigned the value of 0 or 1. Or, for example, the parity bit (P, in the PSW) could be moved into TB8. On receive, the 9th data bit goes into RB8 in Special Function Register SCON, while the stop bit is ignored. The baud rate is programmable to either 1/32 or 1/64 the oscillator frequency.

**Mode 3:** 11 bits are transmitted (through TxD) or received (through RxD): a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). In fact, Mode 3 is the same as Mode 2 in all respects except baud rate. The baud rate in Mode 3 is variable.

In all four modes, transmission is initiated by any instruction that uses SBUF as a destination register. Reception is initiated in Mode 0 by the condition RI = 0 and REN = 1. Reception is initiated in the other modes by the incoming start bit if REN = 1.

### Multiprocessor Communications

Modes 2 and 3 have a special provision for multiprocessor communications. In these modes, 9 data bits are received. The 9th one goes into RB8. Then comes a stop bit. The port can be programmed such that when the stop bit is received, the serial port interrupt will be activated only if RB8 = 1. This feature is enabled by setting bit SM2 in SCON. A way to use this feature in multiprocessor systems is as follows:

When the master processor wants to transmit a block of data to one of several slaves, it first sends out an address byte which identifies the target slave. An address byte differs from a data byte in that the 9th bit is 1 in an address byte and 0 in a data byte. With SM2 = 1, no slave will be interrupted by a data byte. An address byte, however, will interrupt all slaves, so that each slave can examine the received byte and see if it is being addressed. The addressed slave will clear its SM2 bit and prepare to receive the data bytes that will be coming. The slaves that weren't being addressed leave their SM2s set and go on about their business, ignoring the coming data bytes.

SM2 has no effect in Mode 0, and in Mode 1 can be used to check the validity of the stop bit. In a Mode 1 reception, if SM2 = 1, the receive interrupt will not be activated unless a valid stop bit is received.

### Serial Port Control Register

The serial port control and status register is the Special Function Register SCON, shown in Figure 11. This register contains not only the mode selection bits, but also the 9th data bit for transmit and receive (TB8 and RB8), and the serial port interrupt bits (TI and RI).

### Baud Rates

The baud rate in Mode 0 is fixed: Mode 0 Baud Rate = Oscillator Frequency / 12. The baud rate in Mode 2 depends on the value of bit SMOD in Special Function Register PCON. If SMOD = 0 (which is the value on reset), the baud rate is 1/64 the oscillator frequency. If SMOD = 1, the baud rate is 1/32 the oscillator frequency.

Mode 2 Baud Rate =

$$\frac{2^{SMOD}}{64} \times (\text{Oscillator Frequency})$$

In the 80C51, the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate.

### Using Timer 1 to Generate Baud Rates

When Timer 1 is used as the baud rate generator, the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate and the value of SMOD as follows:

Mode 1, 3 Baud Rate =

$$\frac{2^{SMOD}}{32} \times (\text{Timer 1 Overflow Rate})$$

The Timer 1 interrupt should be disabled in this application. The Timer itself can be configured for either "timer" or "counter" operation, and in any of its 3 running modes. In the most typical applications, it is configured for "timer" operation, in the auto-reload mode (high nibble of TMOD = 0010B). In that case the baud rate is given by the formula:

Mode 1, 3 Baud Rate =

$$\frac{2^{SMOD}}{32} \times \frac{\text{Oscillator Frequency}}{12 \times [256 - (TH1)]}$$

One can achieve very low baud rates with Timer 1 by leaving the Timer 1 interrupt enabled, and configuring the Timer to run as a 16-bit timer (high nibble of TMOD = 0001B), and using the Timer 1 interrupt to do a 16-bit software reload. Figure 12 lists various commonly used baud rates and how they can be obtained from Timer 1.

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |

Where SM0, SM1 specify the serial port mode, as follows:

| SM0 | SM1 | Mode | Description | Baud Rate |
|---|---|---|---|---|
| 0 | 0 | 0 | shift register | $f_{osc}$/ 12 |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{osc}$/64 or $f_{osc}$/32 |
| 1 | 1 | 3 | 9-bit UART | variable |

**SM2** Enables the multiprocessor communication feature in Modes 2 and 3. In Mode 2 or 3, if SM2 is set to 1, then RI will not be activated if the received 9th data bit (RB8) is 0. In Mode 1, if SM2=1 then RI will not be activated if a valid stop bit was not received. In Mode 0, SM2 should be 0.

**REN** Enables serial reception. Set by software to enable reception. Clear by software to disable reception.

**TB8** The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired.

**RB8** In Modes 2 and 3, is the 9th data bit that was received. In Mode 1, it SM2=0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used.

**TI** Transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software.

**RI** Receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software.

SU00120

**Figure 11. Serial Port Control (SCON) Register**

| Baud Rate | $f_{osc}$ | SMOD | Timer 1 | | |
|---|---|---|---|---|---|
| | | | C/T | Mode | Reload Value |
| Mode 0 Max: 1.67MHz | 20MHz | X | X | X | X |
| Mode 2 Max: 625k | 20MHz | 1 | X | X | X |
| Mode 1, 3 Max: 104.2k | 20MHz | 1 | 0 | 2 | FFH |
| 19.2k | 11.059MHz | 1 | 0 | 2 | FDH |
| 9.6k | 11.059MHz | 0 | 0 | 2 | FDH |
| 4.8k | 11.059MHz | 0 | 0 | 2 | FAH |
| 2.4k | 11.059MHz | 0 | 0 | 2 | F4H |
| 1.2k | 11.059MHz | 0 | 0 | 2 | E8H |
| 137.5 | 11.986MHz | 0 | 0 | 2 | 1DH |
| 110 | 6MHz | 0 | 0 | 2 | 72H |
| 110 | 12MHz | 0 | 0 | 1 | FEEBH |

**Figure 12. Timer 1 Generated Commonly Used Baud Rates**

**More About Mode 0**

Serial data enters and exits through RxD. TxD outputs the shift clock. 8 bits are transmitted/received: 8 data bits (LSB first). The baud rate is fixed a 1/12 the oscillator frequency.

Figure 13 shows a simplified functional diagram of the serial port in Mode 0, and associated timing.

Transmission is initiated by any instruction that uses SBUF as a destination register. The "write to SBUF" signal at S6P2 also loads a 1 into the 9th position of the transmit shift register and tells the TX Control block to commence a transmission. The internal timing is such that one full machine cycle will elapse between "write to SBUF" and activation of SEND.

SEND enables the output of the shift register to the alternate output function line of P3.0 and also enable SHIFT CLOCK to the alternate output function line of P3.1. SHIFT CLOCK is low during S3, S4, and S5 of every machine cycle, and high during S6, S1, and S2. At S6P2 of every machine cycle in which SEND is active, the contents of the transmit shift are shifted to the right one position.

As data bits shift out to the right, zeros come in from the left. When the MSB of the data byte is at the output position of the shift register, then the 1 that was initially loaded into the 9th position, is just to the left of the MSB, and all positions to the left of that contain zeros. This condition flags the TX Control block to do one last shift and then deactivate SEND and set T1. Both of these actions occur at S1P1 of the 10th machine cycle after "write to SBUF."

Reception is initiated by the condition REN = 1 and R1 = 0. At S6P2 of the next machine cycle, the RX Control unit writes the bits 11111110 to the receive shift register, and in the next clock phase activates RECEIVE.

RECEIVE enable SHIFT CLOCK to the alternate output function line of P3.1. SHIFT CLOCK makes transitions at S3P1 and S6P1 of every machine cycle. At S6P2 of every machine cycle in which RECEIVE is active, the contents of the receive shift register are shifted to the left one position. The value that comes in from the right is the value that was sampled at the P3.0 pin at S5P2 of the same machine cycle.

As data bits come in from the right, 1s shift out to the left. When the 0 that was initially loaded into the rightmost position arrives at the leftmost position in the shift register, it flags the RX Control block to do one last shift and load SBUF. At S1P1 of the 10th machine cycle after the write to SCON that cleared RI, RECEIVE is cleared as RI is set.

**More About Mode 1**

Ten bits are transmitted (through TxD), or received (through RxD): a start bit (0), 8 data bits (LSB first), and a stop bit (1). On receive, the stop bit goes into RB8 in SCON. In the 80C51 the baud rate is determined by the Timer 1 overflow rate.

Figure 14 shows a simplified functional diagram of the serial port in Mode 1, and associated timings for transmit receive.

Transmission is initiated by any instruction that uses SBUF as a destination register. The "write to SBUF" signal also loads a 1 into the 9th bit position of the transmit shift register and flags the TX Control unit that a transmission is requested. Transmission actually commences at S1P1 of the machine cycle following the next rollover in the divide-by-16 counter. (Thus, the bit times are synchronized to the divide-by-16 counter, not to the "write to SBUF" signal.)

The transmission begins with activation of SEND which puts the start bit at TxD. One bit time later, DATA is activated, which enables the output bit of the transmit shift register to TxD. The first shift pulse occurs one bit time after that.

As data bits shift out to the right, zeros are clocked in from the left. When the MSB of the data byte is at the output position of the shift register, then the 1 that was initially loaded into the 9th position is just to the left of the MSB, and all positions to the left of that contain zeros. This condition flags the TX Control unit to do one last shift and then deactivfate SEND and set TI. This occurs at the 10th divide-by-16 rollover after "write to SBUF."

Reception is initiated by a detected 1-to-0 transition at RxD. For this purpose RxD is sampled at a rate of 16 times whatever baud rate has been established. When a transition is detected, the divide-by-16 counter is immediately reset, and 1FFH is written into the input shift register. Resetting the divide-by-16 counter aligns its rollovers with the boundaries of the incoming bit times.

The 16 states of the counter divide each bit time into 16ths. At the 7th, 8th, and 9th counter states of each bit time, the bit detector samples the value of RxD. The value accepted is the value that was seen in at least 2 of the 3 samples. This is done for noise rejection. If the value accepted during the first bit time is not 0, the receive circuits are reset and the unit goes back to looking for another 1-to-0 transition. This is to provide rejection of false start bits. If the start bit proves valid, it is shifted into the input shift register, and reception of the rest of the frame will proceed.

As data bits come in from the right, 1s shift out to the left. When the start bit arrives at the leftmost position in the shift register (which in mode 1 is a 9-bit register), it flags the RX Control block to do one last shift, load SBUF and RB8, and set RI. The signal to load SBUF and RB8, and to set RI, will be generated if, and only if, the following conditions are met at the time the final shift pulse is generated.:

1. R1 = 0, and
2. Either SM2 = 0, or the received stop bit = 1.

If either of these two conditions is not met, the received frame is irretrievably lost. If both conditions are met, the stop bit goes into RB8, the 8 data bits go into SBUF, and RI is activated. At this time, whether the above conditions are met or not, the unit goes back to looking for a 1-to-0 transition in RxD.

**More About Modes 2 and 3**

Eleven bits are transmitted (through TxD), or received (through RxD): a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). On transmit, the 9th data bit (TB8) can be assigned the value of 0 or 1. On receive, the 9the data bit goes into RB8 in SCON. The baud rate is programmable to either 1/32 or 1/64 the oscillator frequency in Mode 2. Mode 3 may have a variable baud rate generated from Timer 1.

Figures 15 and 16 show a functional diagram of the serial port in Modes 2 and 3. The receive portion is exactly the same as in Mode 1. The transmit portion differs from Mode 1 only in the 9th bit of the transmit shift register.

Transmission is initiated by any instruction that uses SBUF as a destination register. The "write to SBUF" signal also loads TB8 into the 9th bit position of the transmit shift register and flags the TX Control unit that a transmission is requested. Transmission commences at S1P1 of the machine cycle following the next rollover in the divide-by-16 counter. (Thus, the bit times are synchronized to the divide-by-16 counter, not to the "write to SBUF" signal.)

The transmission begins with activation of SEND, which puts the start bit at TxD. One bit time later, DATA is activated, which enables the output bit of the transmit shift register to TxD. The first shift pulse occurs one bit time after that. The first shift clocks a 1 (the stop bit) into the 9th bit position of the shift register. Thereafter, only zeros are clocked in. Thus, as data bits shift out to the right, zeros are clocked in from the left. When TB8 is at the output position of the shift register, then the stop bit is just to the left of TB8, and all positions to the left of that contain zeros. This condition flags the TX Control unit to do one last shift and then deactivate SEND and set TI. This occurs at the 11th divide-by-16 rollover after "write to SUBF."

Reception is initiated by a detected 1-to-0 transition at RxD. For this purpose RxD is sampled at a rate of 16 times whatever baud rate has been established. When a transition is detected, the divide-by-16 counter is immediately reset, and 1FFH is written to the input shift register.

At the 7th, 8th, and 9th counter states of each bit time, the bit detector samples the value of R-D. The value accepted is the value that was seen in at least 2 of the 3 samples. If the value accepted during the first bit time is not 0, the receive circuits are reset and the unit goes back to looking for another 1-to-0 transition. If the start bit proves valid, it is shifted into the input shift register, and reception of the rest of the frame will proceed.

As data bits come in from the right, 1s shift out to the left. When the start bit arrives at the leftmost position in the shift register (which in Modes 2 and 3 is a 9-bit register), it flags the RX Control block to do one last shift, load SBUF and RB8, and set RI.

The signal to load SBUF and RB8, and to set RI, will be generated if, and only if, the following conditions are met at the time the final shift pulse is generated.

1. RI = 0, and
2. Either SM2 = 0, or the received 9th data bit = 1.

If either of these conditions is not met, the received frame is irretrievably lost, and RI is not set. If both conditions are met, the received 9th data bit goes into RB8, and the first 8 data bits go into SBUF. One bit time later, whether the above conditions were met or not, the unit goes back to looking for a 1-to-0 transition at the RxD input.

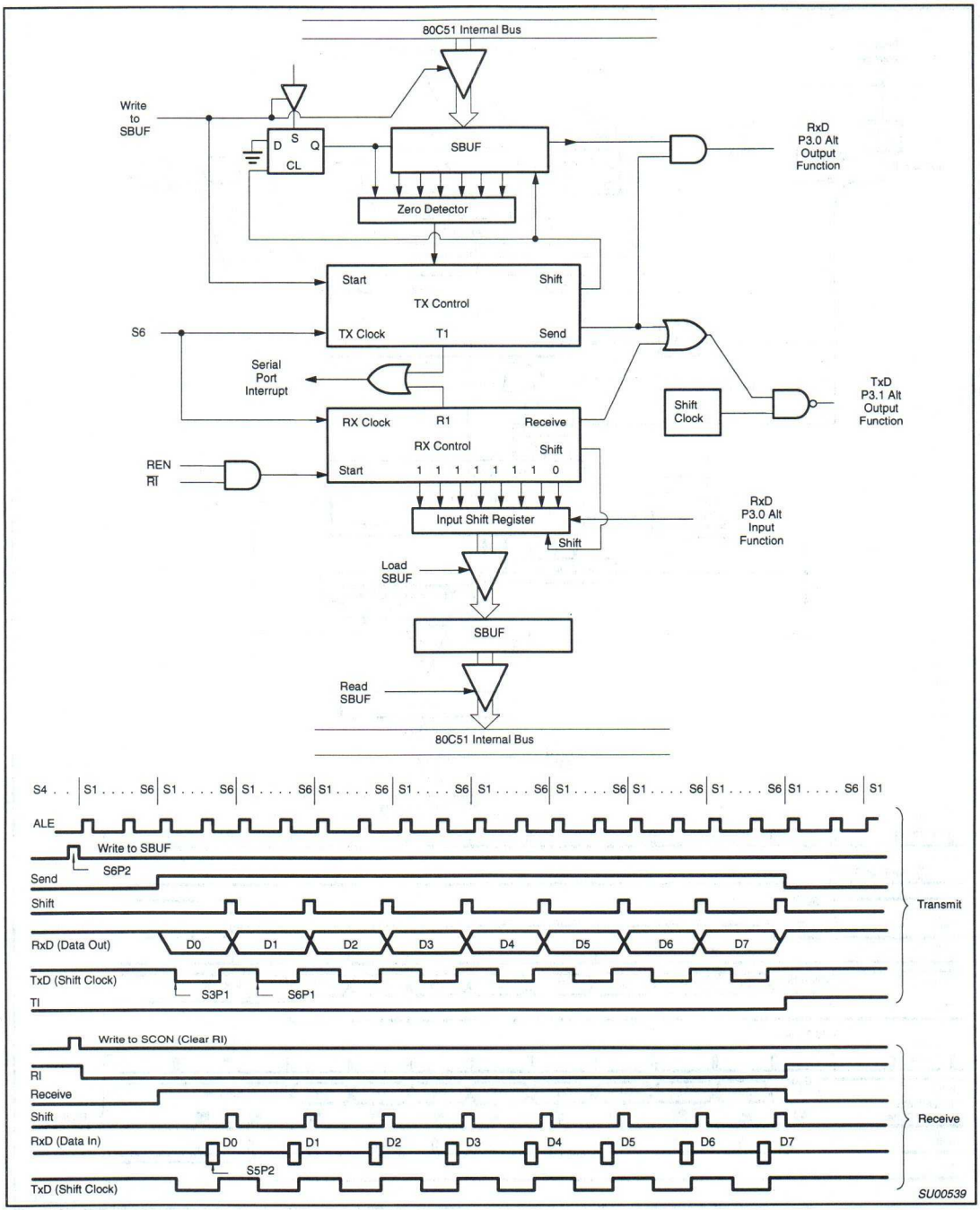



**Figure 13. Serial Port Mode 0**

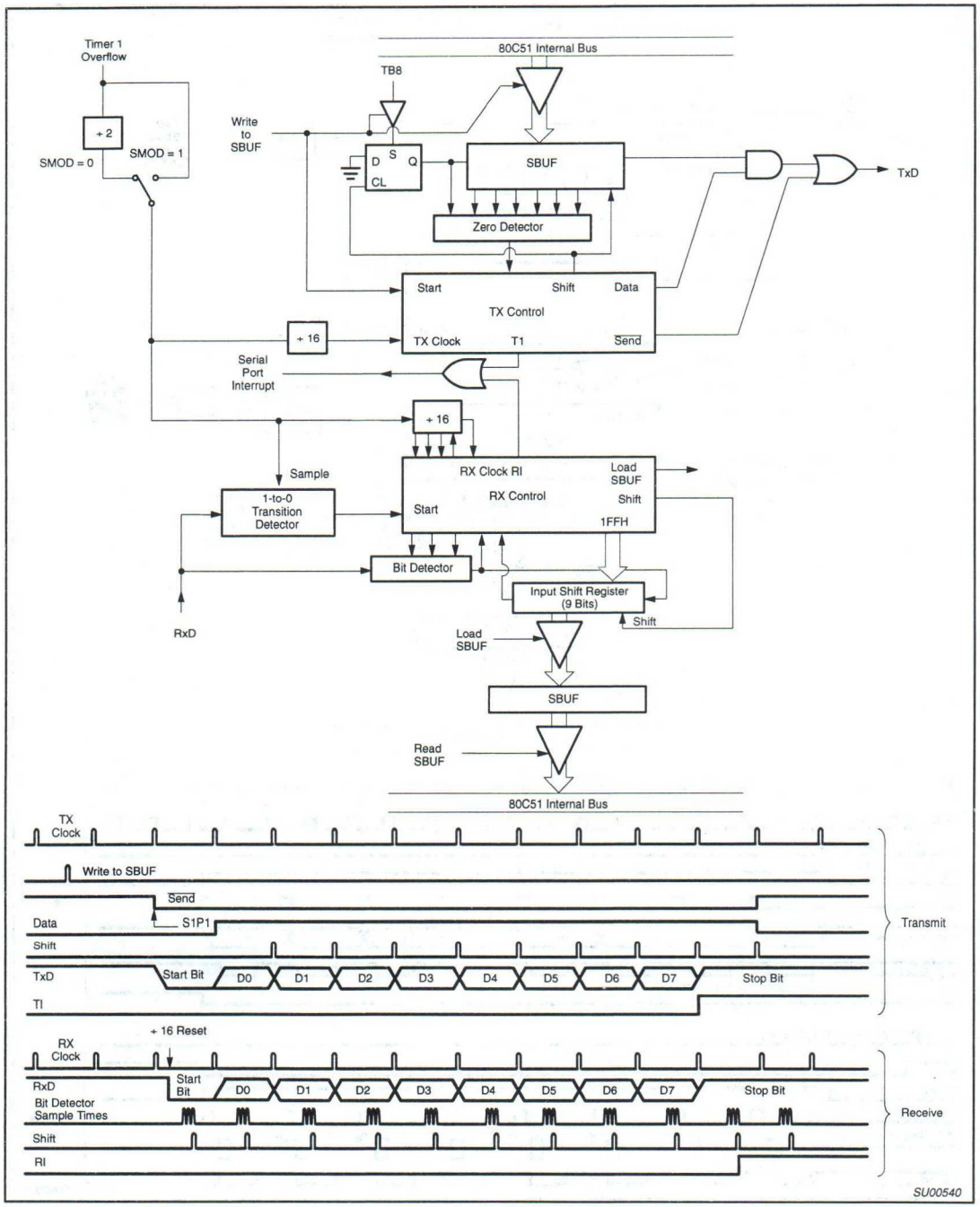



**Figure 14. Serial Port Mode 1**

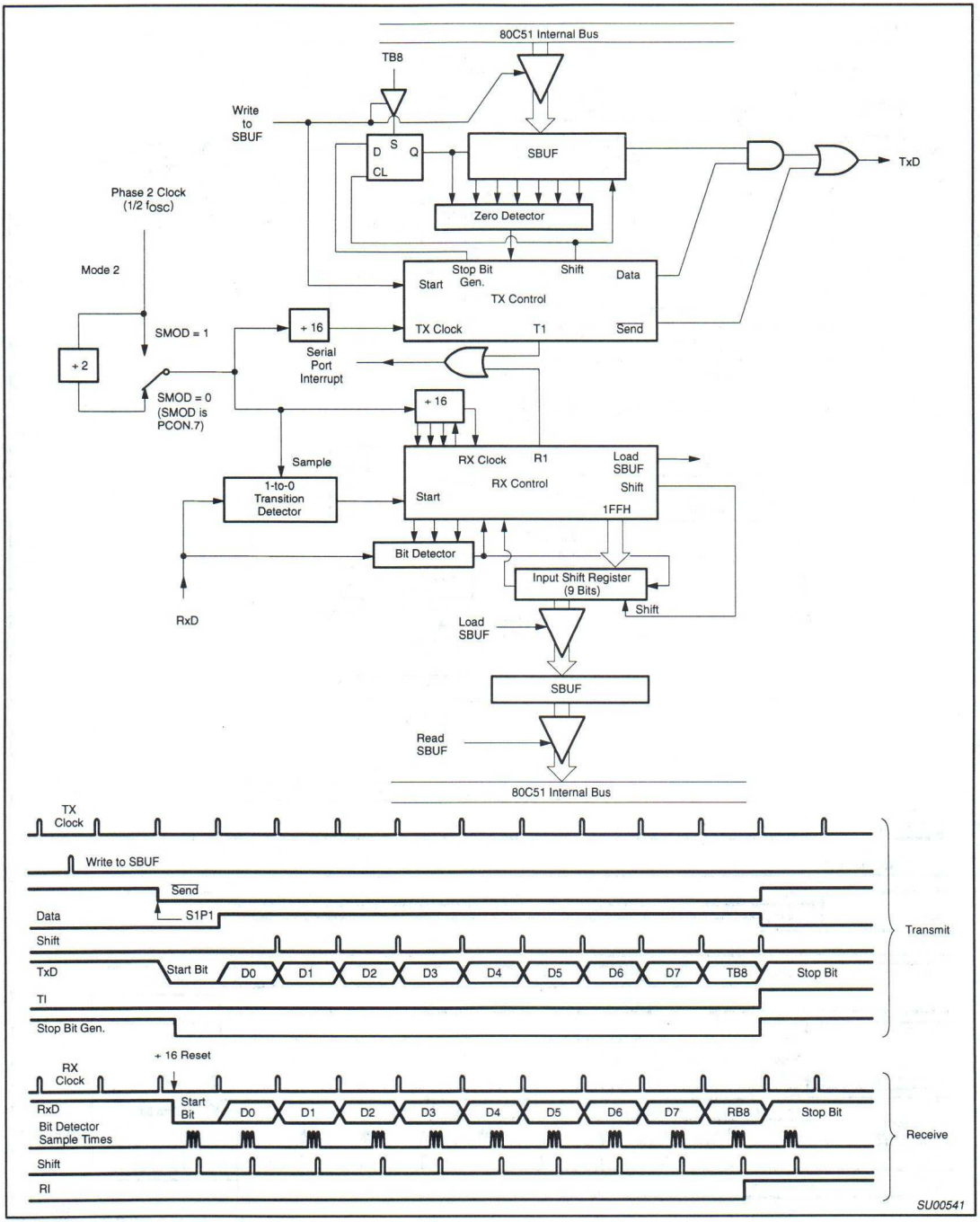



**Figure 15. Serial Port Mode 2**

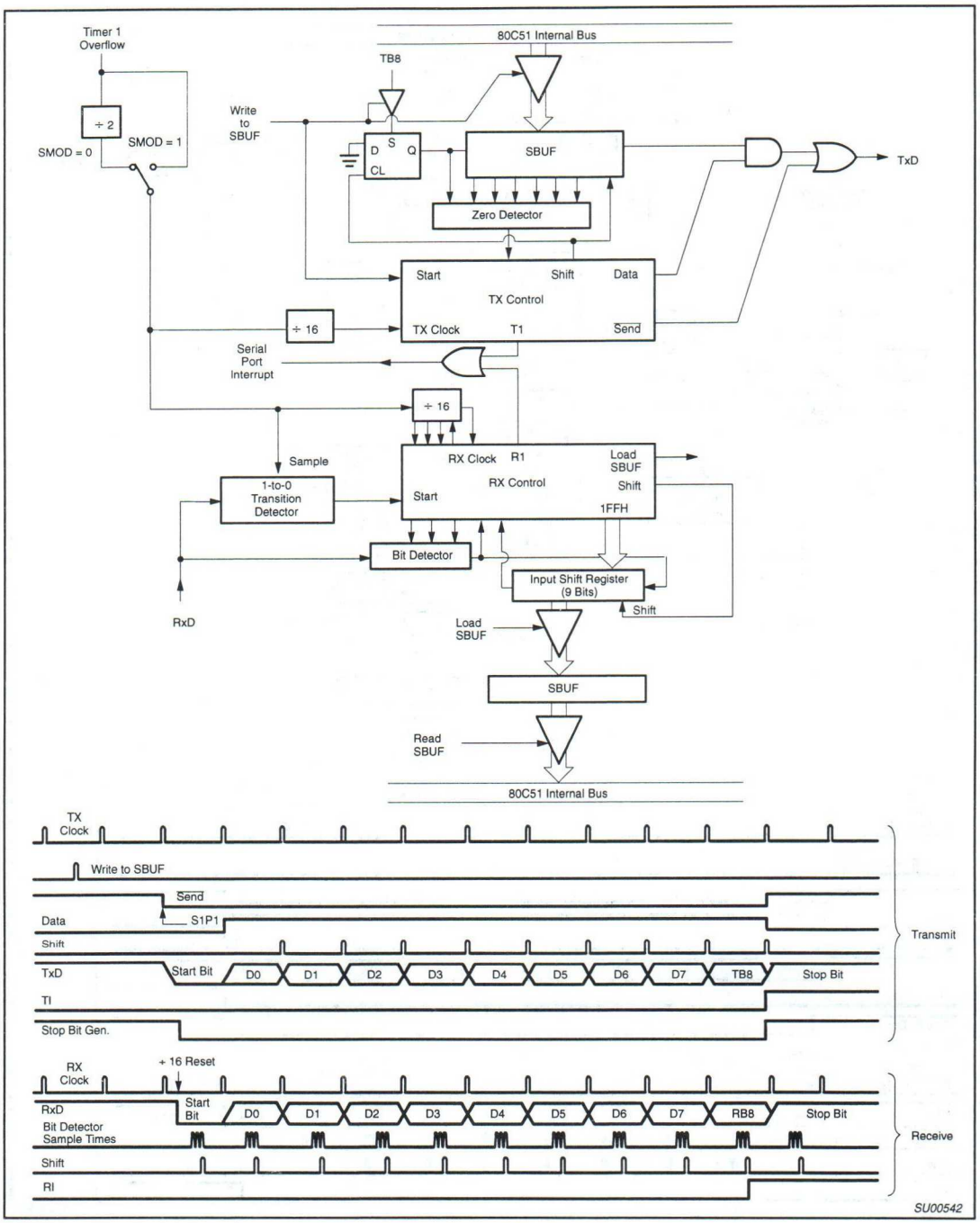



**Figure 16. Serial Port Mode 3**

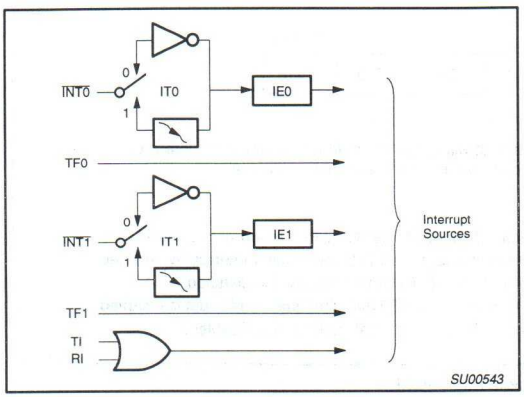



**Figure 17. 80C51 Interrupt Sources**

## Interrupts

The 80C51 provides 5 interrupt sources. These are shown in Figure 17. The External Interrupts $\overline{INT0}$ and $\overline{INT1}$ can each be either level-activated or transition-activated, depending on bits IT0 and IT1 in Register TCON. The flags that actually generate these interrupts are bits IE0 and IE1 in TCON. When an external interrupt is generated, the flag that generated it is cleared by the hardware when the service routine is vectored to only if the interrupt was transition-activated. If the interrupt was level-activated, then the external requesting source is what controls the request flag, rather than the on-chip hardware.

The Timer 0 and Timer 1 Interrupts are generated by TF0 and TF1, which are set by a rollover in their respective Timer/Counter registers (except see Timer 0 in Mode 3). When a timer interrupt is generated, the flag that generated it is cleared by the on-chip hardware when the service routine is vectored to.

The Serial Port Interrupt is generated by the logical OR of RI and TI. Neither of these flags is cleared by hardware when the service routine is vectored to. In fact, the service routine will normally have to determine whether it was RI or TI that generated the interrupt, and the bit will have to be cleared in software.

All of the bits that generate interrupts can be set or cleared by software, with the same result as though it had been set or cleared by hardware. That is, interrupts can be generated or pending interrupts can be canceled in software.

Each of these interrupt sources can be individually enabled or disabled by setting or clearing a bit in Special Function Register IE (Figure 18). IE also contains a global disable bit, EA, which disables all interrupts at once.

### Priority Level Structure

Each interrupt source can also be individually programmed to one of two priority levels by setting or clearing a bit in Special Function Register IP (Figure 19). A low-priority interrupt can itself be interrupted by a high-priority interrupt, but not by another low-priority interrupt. A high-priority interrupt can't be interrupted by any other interrupt source.

If two request of different priority levels are received simultaneously, the request of higher priority level is serviced. If requests of the same priority level are received simultaneously, an internal polling sequence determines which request is serviced. Thus within each priority level there is a second priority structure determined by the polling sequence as follows:

| Source | Priority Within Level |
|---|---|
| 1. IE0 | (highest) |
| 2. TF0 | |
| 3. IE1 | |
| 4. TF1 | |
| 5. RI+TI | (lowest) |

Note that the "priority within level" structure is only used to resolve simultaneous requests of the same priority level.

The IP register contains a number of unimplemented bits. IP.7, IP.6, and IP.5 are reserved in the 80C51. User software should not write 1s to these positions, since they may be used in other 8051 Family products.

### How Interrupts Are Handled

The interrupt flags are sampled at S5P2 of every machine cycle. The samples are polled during the following machine cycle. If one of the flags was in a set condition at S5P2 of the preceding cycle, the polling cycle will find it and the interrupt system will generate an LCALL to the appropriate service routine, provided this hardware-generated LCALL is not blocked by any of the following conditions:

1. An interrupt of equal or higher priority level is already in progress.
2. The current (polling) cycle is not the final cycle in the execution of the instruction in progress.
3. The instruction in progress is RETI or any write to the IE or IP registers.

Any of these three conditions will block the generation of the LCALL to the interrupt service routine. Condition 2 ensures that the instruction in progress will be completed before vectoring to any service routine. Condition 3 ensures that if the instruction in progress is RETI or any access to IE or IP, then at least one more instruction will be executed before any interrupt is vectored to.

The polling cycle is repeated with each machine cycle, and the values polled are the values that were present at S5P2 of the previous machine cycle. Note that if an interrupt flag is active but not being responded to for one of the above conditions, if the flag is not still active when the blocking condition is removed, the denied interrupt will not be serviced. In other words, the fact that the interrupt flag was once active but not serviced is not remembered. Every polling cycle is new.

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| EA | X | X | ES | ET1 | EX1 | ET0 | EX0 |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IE.7 | EA | Disables all interrupts. If EA=0, no interrupt will be acknowledged. If EA=1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit. |
| IE.6 | — | Reserved. |
| IE.5 | — | Reserved. |
| IE.4 | ES | Enables or disables the Serial Port interrupt. If ES=0, the Serial Port interrupt is disabled. |
| IE.3 | ET1 | Enables or disables the Timer 1 Overflow interrupt. If ET1=0, the Timer 1 interrupt is disabled. |
| IE.2 | EX1 | Enables or disables External Interrupt 1. If EX1=0, External interrupt 1 is disabled. |
| IE.1 | ET0 | Enables or disables the Timer 0 Overflow interrupt. If ET0=0, the Timer 0 interrupt is disabled. |
| IE.0 | EX0 | Enables or disables External interrupt 0. If EX0=0, External interrupt 0 is disabled. |

SU00544

**Figure 18. Interrupt Enable Register (IE)**

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| X | X | X | PS | PT1 | PX1 | PT0 | PX0 |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IP.7 | — | Reserved. |
| IP.6 | — | Reserved. |
| IP.5 | — | Reserved. |
| IP.4 | PS | Defines the Serial Port interrupt priority level. PS=1 programs it to the higher priority level. |
| IP.3 | PT1 | Defines the Timer 1 interrupt priority level. PT1=1 programs it to the higher priority level. |
| IP.2 | PX1 | Defines the External Interrupt 1 priority level. PX1=1 programs it to the higher priority level. |
| IP.1 | PT0 | Enables or disables the Timer 0 interrupt priority level. PT0=1 programs it to the higher priority level. |
| IP.0 | PX0 | Defines the External Interrupt 0 priority level. PX0=1 programs it to the higher priority level. |

SU00545

**Figure 19. Interrupt Priority Register (IP)**

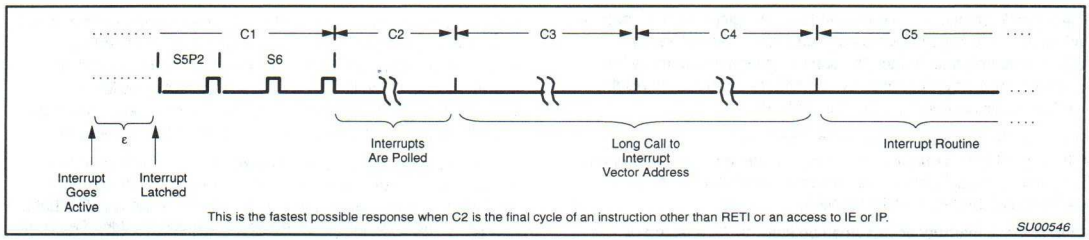



**Figure 20. Interrupt Response Timing Diagram**

The polling cycle/LCALL sequence is illustrated in Figure 20.

Note that if an interrupt of higher priority level goes active prior to S5P2 of the machine cycle labeled C3 in Figure 20, then in accordance with the above rules it will be vectored to during C5 and C6, without any instruction of the lower priority routine having been executed.

Thus the processor acknowledges an interrupt request by executing a hardware-generated LCALL to the appropriate servicing routine. In some cases it also clears the flag that generated the interrupt, and in other cases it doesn't. It never clears the Serial Port flag. This has to be done in the user's software. It clears an external interrupt flag (IE0 or IE1) only if it was transition-activated. The hardware-generated LCALL pushes the contents of the Program Counter on to the stack (but it does not save the PSW) and reloads the PC with an address that depends on the source of the interrupt being vectored to, as shown below:

| Source | Vector Address |
|---|---|
| IE0 | 0003H |
| TF0 | 000BH |
| IE1 | 0013H |
| TF1 | 001BH |
| RI+TI | 0023H |

Execution proceeds from that location until the RETI instruction is encountered. The RETI instruction informs the processor that this

interrupt routine is no longer in progress, then pops the top two bytes from the stack and reloads the Program Counter. Execution of the interrupted program continues from where it left off.

Note that a simple RET instruction would also have returned execution to the interrupted program, but it would have left the interrupt control system thinking an interrupt was still in progress, making future interrupts impossible.

**External Interrupts**

The external sources can be programmed to be level-activated or transition-activated by setting or clearing bit IT1 or IT0 in Register TCON. If ITx = 0, external interrupt x is triggered by a detected low at the INTx pin. If ITx = 1, external interrupt x is edge triggered. In this mode if successive samples of the INTx pin show a high in one cycle and a low in the next cycle, interrupt request flag IEx in TCON is set. Flag bit IEx then requests the interrupt.

Since the external interrupt pins are sampled once each machine cycle, an input high or low should hold for at least 12 oscillator periods to ensure sampling. If the external interrupt is transition-activated, the external source has to hold the request pin high for at least one cycle, and then hold it low for at least one cycle. This is done to ensure that the transition is seen so that interrupt request flag IEx will be set. IEx will be automatically cleared by the CPU when the service routine is called.

If the external interrupt is level-activated, the external source has to hold the request active until the requested interrupt is actually generated. Then it has to deactivate the request before the interrupt service routine is completed, or else another interrupt will be generated.

**Response Time**

The $\overline{INT0}$ and $\overline{INT1}$ levels are inverted and latched into IE0 and IE1 at S5P2 of every machine cycle. The values are not actually polled by the circuitry until the next machine cycle. If a request is active and conditions are right for it to be acknowledged, a hardware subroutine call to the requested service routine will be the next instruction to be executed. The call itself takes two cycles. Thus, a minimum of three complete machine cycles elapse between activation of an external interrupt request and the beginning of execution of the first instruction of the service routine. Figure 20 shows interrupt response timings.

A longer response time would result if the request is blocked by one of the 3 previously listed conditions. If an interrupt of equal or higher priority level is already in progress, the additional wait time obviously depends on the nature of the other interrupt's service routine. If the instruction in progress is not in its final cycle, the additional wait time cannot be more the 3 cycles, since the longest instructions (MUL and DIV) are only 4 cycles long, and if the instruction in progress is RETI or an access to IE or IP, the additional wait time cannot be more than 5 cycles (a maximum of one more cycle to complete the instruction in progress, plus 4 cycles to complete the next instruction if the instruction is MUL or DIV).

Thus, in a single-interrupt system, the response time is always more than 3 cycles and less than 9 cycles.

## Single-Step Operation

The 80C51 interrupt structure allows single-step execution with very little software overhead. As previously noted, an interrupt request will not be responded to while an interrupt of equal priority level is still in progress, nor will it be responded to after RETI until at least one other instruction has been executed. Thus, once an interrupt routine has been entered, it cannot be re-entered until at least one instruction of the interrupted program is executed. One way to use this feature for single-step operation is to program one of the external interrupts (e.g., $\overline{INT0}$) to be level-activated. The service routine for the interrupt will terminate with the following code:

```
JNB   P3.2,$  ;Wait Till INT0 Goes High
JB    P3.2,$  ;Wait Till INT0 Goes Low
RETI          ;Go Back and Execute One Instruction
```

Now if the $\overline{INT0}$ pin, which is also the P3.2 pin, is held normally low, the CPU will go right into the External Interrupt 0 routine and stay there until $\overline{INT0}$ is pulsed (from low to high to low). Then it will execute RETI, go back to the task program, execute one instruction, and immediately re-enter the External Interrupt 0 routine to await the next pulsing of P3.2. One step of the task program is executed each time P3.2 is pulsed.

## Reset

The reset input is the RST pin, which is the input to a Schmitt Trigger. A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. The CPU responds by generating an internal reset, with the timing shown in Figure 21.

The external reset signal is asynchronous to the internal clock. The RST pin is sampled during State 5 Phase 2 of every machine cycle. The port pins will maintain their current activities for 19 oscillator periods after a logic 1 has been sampled at the RST pin; that is, for 19 to 31 oscillator periods after the external reset signal has been applied to the RST pin.

The internal reset algorithm writes 0s to all the SFRs except the port latches, the Stack Pointer, and SBUF. The port latches are initialized to FFH, the Stack Pointer to 07H, and SBUF is indeterminate. Table 1 lists the SFR reset values. The internal RAM is not affected by reset. On power up the RAM content is indeterminate.

**Table 1. 80C51 SFR Reset Values**

| REGISTER | RESET VALUE |
|---|---|
| PC | 000H |
| ACC | 00H |
| B | 00H |
| PSW | 00H |
| SP | 07H |
| DPTR | 0000H |
| P0–P3 | FFH |
| IP | XXX00000B |
| IE | 0XX00000B |
| TMOD | 00H |
| TCON | 00H |
| TH0 | 00H |
| TL0 | 00H |
| TH1 | 00H |
| TL1 | 00H |
| SCON | 00H |
| SBUF | Indeterminate |
| PCON (NMOS) | 0XXXXXXXB |
| PCON (CMOS) | 0XXX0000B |

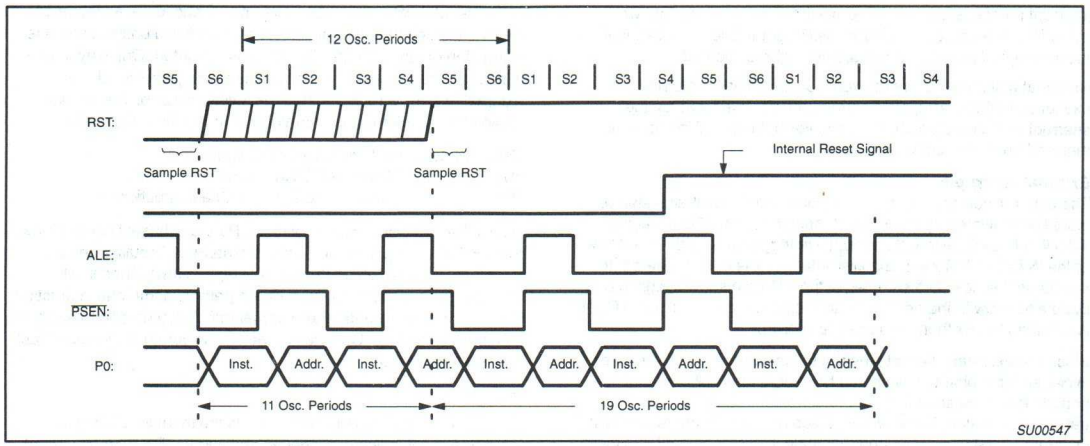



**Figure 21. Reset Timing**

## Power-on Reset

An automatic reset can be obtained when $V_{CC}$ is turned on by connecting the RST pin to $V_{CC}$ through a 10µf capacitor and to $V_{SS}$ through an 8.2k resistor, providing the $V_{CC}$ rise time does not exceed 1 millisecond and the oscillator start-up time does not exceed 10 milliseconds. This power-on reset circuit is shown in Figure 22. The CMOS devices do not require the 8.2k pulldown resistor, although its presence does no harm.

When power is turned on, the circuit holds the RST pin high for an amount of time that depends on the value of the capacitor and the rate at which it charges. To ensure a good reset, the RST pin must be high long enough to allow the oscillator time to start-up (normally a few ms) plus two machine cycles.

*Note that the port pins will be in a random state until the oscillator has started and the internal reset algorithm has written 1s to them.*

With this circuit, reducing $V_{CC}$ quickly to 0 causes the RST pin voltage to momentarily fall below 0V. However, this voltage is internally limited, and will not harm the device.

## Power-Saving Modes of Operation

For applications where power consumption is critical the CMOS version provides power reduced modes of operation as a standard feature. The power down mode in NMOS devices is no longer a standard feature.

### CMOS Power Reduction Mode

CMOS versions have two power reducing modes, Idle and Power Down. The input through which backup power is supplied during these operations is $V_{CC}$. Figure 23 shows the internal circuitry which implements these features. In the Idle modes (IDL = 1), the oscillator continues to run and the Interrupt, Serial Port, and Timer blocks continue to be clocked, but the clock signal is gated off to the CPU. In Power Down (PD = 1), the oscillator is frozen. The Idle and Power Down Modes are activated by setting bits in Special Function Register PCON. The address of this register is 87H. Figure 24 details its contents.

In the NMOS devices the PCON register only contains SMOD. The other four bits are implemented only in the CMOS devices. User software should never write 1s to unimplemented bits, since they may be used in other 80C51 Family products.

## Idle Mode

An instruction that sets PCON.0 causes that to be the last instruction executed before going into the Idle mode, the internal clock signal is gated off to the CPU but not to the Interrupt, Timer, and Serial Port functions. The CPU status is preserved in its entirety; the Stack Pointer, Program Counter, Program Status Word, Accumulator, and all other registers maintain their data during Idle. The port pins hold the logical states they had at the time Idle was activated. ALE and $\overline{PSEN}$ hold at logic high levels.

There are two ways to terminate the Idle. Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware, terminating the Idle mode. The interrupt will be serviced, and following RETI, the next instruction to be executed will be the one following the instruction that put the device into Idle.

The flag bits GF0 and GF1 can be used to give an indication if an interrupt occurred during normal operation or during an Idle. For example, an instruction that activates Idle can also set one or both flag bits. When Idle is terminated by an interrupt, the interrupt service routine can examine the flag bits. The other way of terminating the Idle mode is with a hardware reset. Since the clock oscillator is still running, the hardware reset needs to be held active for only two machine cycles (24 oscillator periods) to complete the reset.

The signal at the RST pin clears the IDL bit directly and asynchronously. At this time the CPU resumes program execution from where it left off; that is, at the instruction following the one that invoked the Idle Mode. As shown in Figure 21, two or three machine cycles of program execution may take place before the internal reset algorithm takes control. On-chip hardware inhibits access to the internal RAM during this time, but access to the port pins is not inhibited, so, the insertion of 3 NOP instructions is recommended following the instruction that invokes idle mode. To eliminate the possibility of unexpected outputs at the port pins, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external Data RAM.

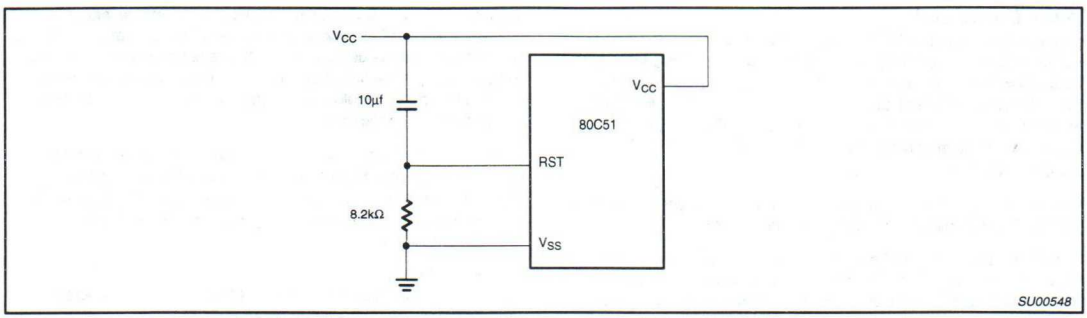



**Figure 22. Power-On Reset Circuit**

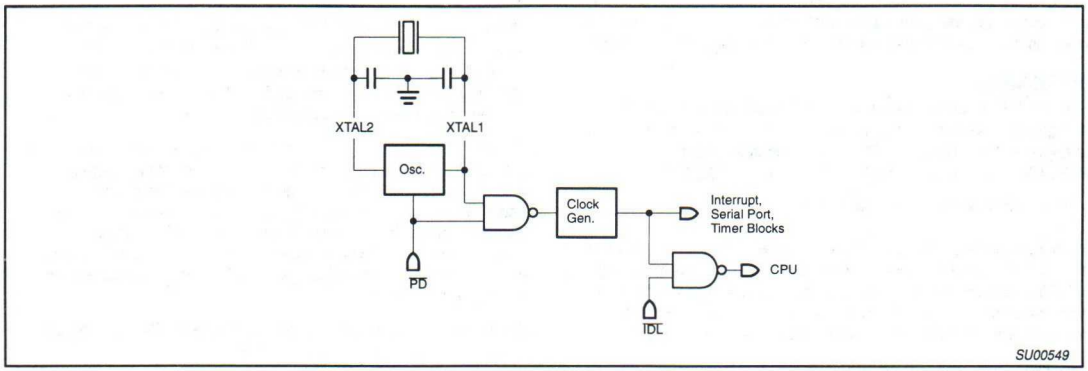



**Figure 23. Idle and Power Down Hardware**

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| SMOD | — | — | — | GF1 | GF0 | PD | IDL |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| PCON.7 | SMOD | Double Baud rate bit. When set to a 1 and Timer 1 is used to generate baud rate, and the Serial Port is used in modes 1, 2, or 3. |
| PCON.6 | — | Reserved. |
| PCON.5 | — | Reserved. |
| PCON.4 | — | Reserved. |
| PCON.3 | GF1 | General-purpose flag bit. |
| PCON.2 | GF0 | General-purpose flag bit. |
| PCON.1 | PD | Power-Down bit. Setting this bit activates power-down operation. |
| PCON.0 | IDL | Idle mode bit. Setting this bit activate idle mode operation. |

If 1s are written to PD and IDL at the same time, PD takes precedence. The reset value of PCON is (0XXX0000).
In the NMOS devices, the PCON register only contains SMOD. The other four bits are implemented only in the CMOS devices. User software should never write 1s to unimplemented bits, since they may be used in future products.

SU00550

**Figure 24. Power Control (PCON) Register**

## Power-Down Mode

An instruction that sets PCON.1 causes that to be the last instruction executed before going into the Power Down mode. In the Power Down mode, the on-chip oscillator is stopped. With the clock frozen, all functions are stopped, the contents of the on-chip RAM and Special Function Registers are maintained. The port pins output the values held by their respective SFRs. The ALE and $\overline{PSEN}$ output are held low.

The only exit from Power Down is a hardware reset. Reset redefines all the SFRs, but does not change the on-chip RAM.

In the Power Down mode of operation, $V_{CC}$ can be reduced to as low as 2V. Care must be taken, however, to ensure that $V_{CC}$ is not reduced before the Power Down mode is invoked, and that $V_{CC}$ is restored to its normal operating level, before the Power Down mode is terminated. The reset that terminates Power Down also frees the oscillator. The reset should not be activated before $V_{CC}$ is restored to its normal operating level, and must be held active long enough to allow the oscillator to restart and stabilize (normally less than 10ms).

## ONCE Mode

The ONCE ("on-circuit emulation") mode facilitates testing and debugging of systems using the device without the device having to be removed from the circuit. The ONCE mode is invoked by:

1. Pull ALE low while the device in in reset and $\overline{PSEN}$ is high;
2. Hold ALE low as RST is deactivated.

While the device is in the ONCE mode, the Port 0 pins go into a float state, and the other port pins and ALE and $\overline{PSEN}$ are weakly pulled high. The oscillator circuit remains active. While the device is in this mode, an emulator or test CPU can be used to drive the circuit. Normal operation is restored after a normal reset is applied.

## The On-Chip Oscillators

### NMOS Version

The on-chip oscillator circuitry for the NMOS members of the 80C51 family is a single stage linear inverter (Figure 25), intended for use as a crystal-controlled, positive reactance oscillator (Figure 26). In this application the crystal is operated in its fundamental response mode as an inductive reactance in parallel resonance with capacitance external to the crystal.

The crystal specifications and capacitance values (C1 and C2 in Figure 26) are not critical. 30pF can be used in these positions at any frequency with good quality crystals. A ceramic resonator can be used in place of the crystal in cost-sensitive applications. When a ceramic resonator is used, C1 and C2 are normally selected to be of somewhat higher values, typically, 47pF. The manufacturer of the ceramic resonator should be consulted for recommendation on the values of these capacitors.

To drive the NMOS parts with an external clock source, apply the external clock signal to XTAL2, and ground XTAL1, as shown in Figure 27. A pullup resistor may be used (to increase noise margin), but is optional if $V_{OH}$ of the driving gate exceeds the $V_{IH}$ minimum specification of XTAL.

### CMOS Versions

The on-chip oscillator circuitry for the 80C51, shown in Figure 28, consists of a single stage linear inverter intended for use as a crystal-controlled, positive reactance oscillator in the same manner as the NMOS parts. However, there are some important differences.

One difference is that the 80C51 is able to turn off its oscillator under software control (by writing a 1 to the PD bit in PCON). Another difference is that, in the 80C51, the internal clocking circuitry is driven by the signal at XTAL1, whereas in the NMOS versions it is by the signal at XTAL2.

The feedback resistor $R_f$ in Figure 28 consists of paralleled n- and p-channel FETs controlled by the PD bit, such that $R_f$ is opened when PD = 1. The diodes D1 and D2, which act as clamps to $V_{CC}$ and $V_{SS}$, are parasitic to the $R_f$ FETs. The oscillator can be used with the same external components as the NMOS versions, as shown in Figure 29. Typically, C1 = C2 = 30pF when the feedback element is a quartz crystal, and C1 = C2 = 47pF when a ceramic resonator is used.

When a crystal is used at frequencies above 25MHz, C1 and C2 should be in the range of 20pF to 25pF.

To drive the CMOS parts with an external clock source, apply the external clock signal to XTAL1, and leave XTAL2 float, as shown in Figure 30.

The reason for this change from the way the NMOS part is driven can be seen by comparing Figures 26 and 28. In the NMOS devices the internal timing circuits are driven by the signal at XTAL2. In the CMOS devices the internal timing circuits are driven by the signal at XTAL1.

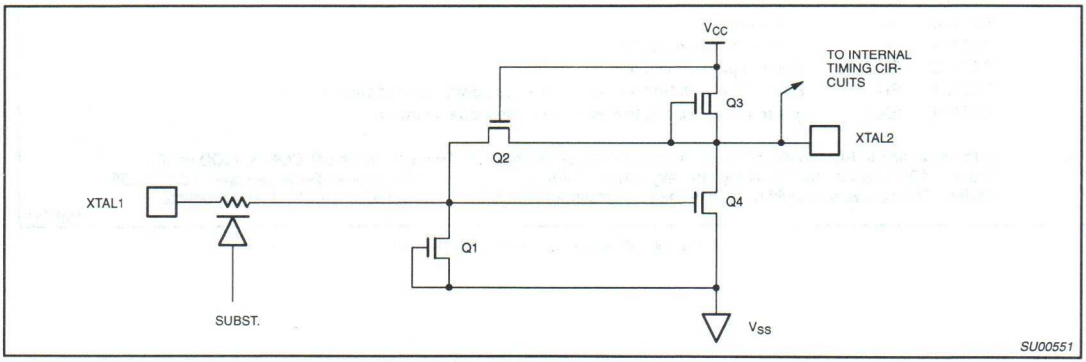



**Figure 25. On-Chip Oscillator in the NMOS Version of the 8051 Family**

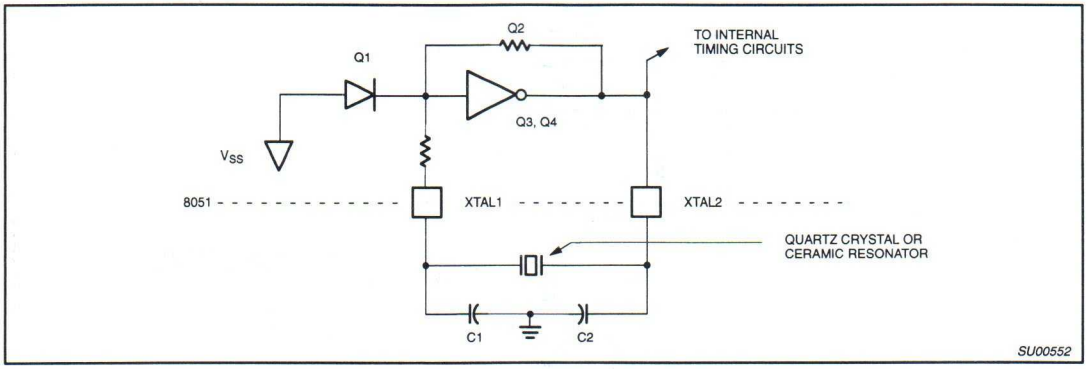



**Figure 26. Using the NMOS On-Chip Oscillator**

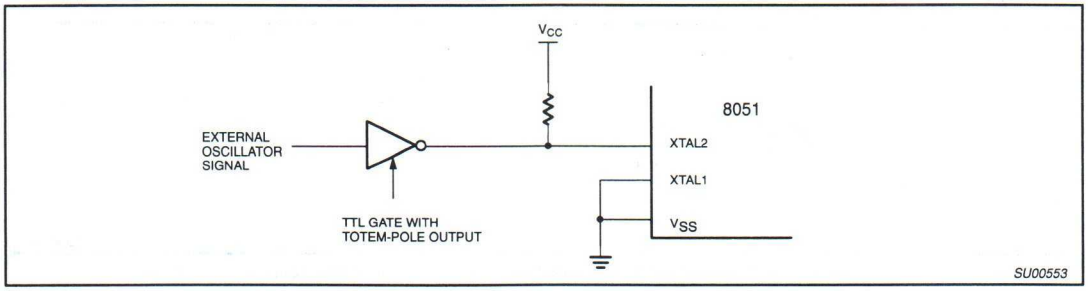



**Figure 27. Driving the NMOS 8051 Family Parts with an External Clock**

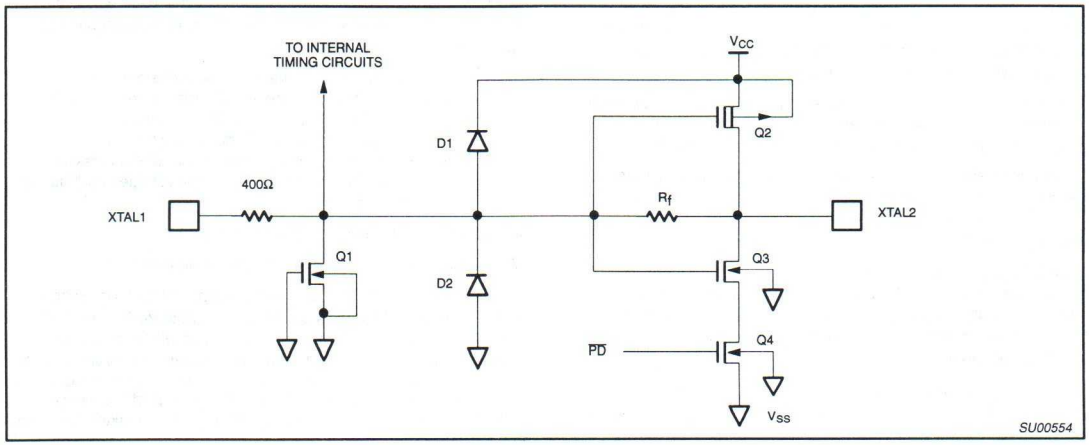



**Figure 28. On-Chip Oscillator Circuitry in the CMOS Version of the 80C51 Family**

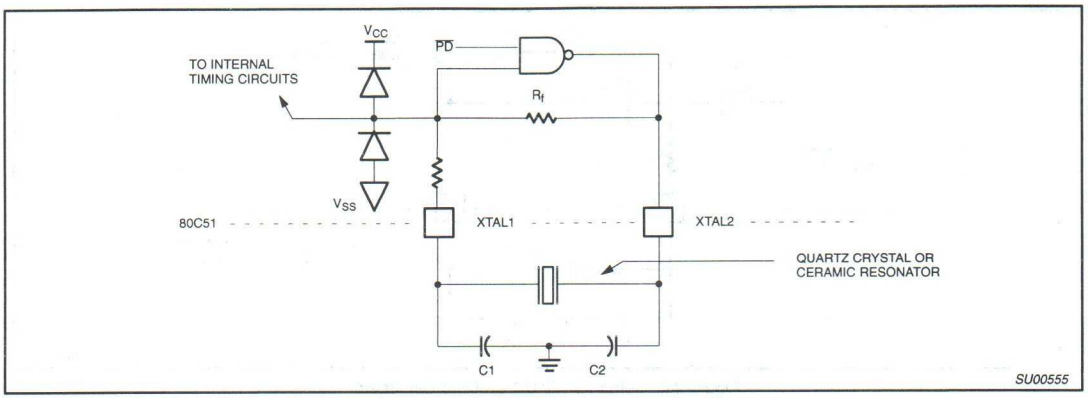



**Figure 29. Using the CMOS On-Chip Oscillator**

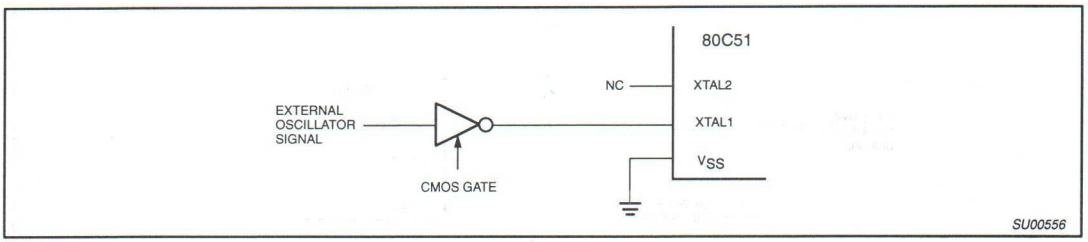



**Figure 30. Driving the CMOS Family Parts with an External Clock Source**

## Internal Timing

Figures 31 through 34 show when the various strobe and port signals are clocked internally. The figures do not show rise and fall times of the signals, nor do they show propagation delays between the XTAL2 signal and events at other pins.

Rise and fall times are dependent on the external loading that each pin must drive. They are often taken to be something in the neighborhood of 10ns, measured between 0.8V and 2.0V.

Propagation delays are different for different pins. For a given pin they vary with pin loading, temperature, $V_{CC}$, and manufacturing lot. If the XTAL2 waveform is taken as the timing reference, prop delays may vary up to ±200%.

The AC Timings section of the data sheets do not reference any timing to the XTAL2 waveform. Rather, they relate the critical edges of control and input signals to each other. The timings published in the data sheets include the effects of propagation delays under the specified test conditions.

## 80C51 Pin Descriptions

**ALE/PROG:** Address Latch Enable output pulse for latching the low byte of the address during accesses to external memory. ALE is emitted at a constant rate of 1/6 of the oscillator frequency, for external timing or clocking purposes, even when there are no accesses to external memory. (However, one ALE pulse is skipped during each access to external Data Memory.) This pin is also the program pulse input (PROG) during EPROM programming.

**$\overline{\text{PSEN}}$:** Program Store Enable is the read strobe to external Program Memory. When the device is executing out of external Program Memory, $\overline{\text{PSEN}}$ is activated twice each machine cycle (except that two $\overline{\text{PSEN}}$ activations are skipped during accesses to external Data Memory). $\overline{\text{PSEN}}$ is not activated when the device is executing out of internal Program Memory.

**$\overline{\text{EA}}$/$V_{PP}$:** When $\overline{\text{EA}}$ is held high the CPU executes out of internal Program Memory (unless the Program Counter exceeds 0FFFH in the 80C51). Holding $\overline{\text{EA}}$ low forces the CPU to execute out of external memory regardless of the Program Counter value. In the 80C31, $\overline{\text{EA}}$ must be externally wired low. In the EPROM devices, this pin also receives the programming supply voltage ($V_{PP}$) during EPROM programming.

**XTAL1:** Input to the inverting oscillator amplifier.

**XTAL2:** Output from the inverting oscillator amplifier.

**Port 0:** Port 0 is an 8-bit open drain bidirectional port. As an open drain output port, it can sink eight LS TTL loads. Port 0 pins that have 1s written to them float, and in that state will function as high impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external memory. In this application it uses strong internal pullups when emitting 1s. Port 0 emits code bytes during program verification. In this application, external pullups are required.

**Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pullups. Port 1 pins that have 1s written to them are pulled high by the internal pullups, and in that state can be used as inputs. As inputs, port 1 pins that are externally being pulled low will source current because of the internal pullups.

**Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pullups. Port 2 emits the high-order address byte during accesses to external memory that use 16-bit addresses. In this application, it uses the strong internal pullups when emitting 1s.

**Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pullups. It also serves the functions of various special features of the 80C51 Family as follows:

| Port Pin | Alternate Function |
|---|---|
| P3.0 | RxD (serial input port) |
| P3.1 | TxD (serial output port) |
| P3.2 | $\overline{INT0}$ (external interrupt 0) |
| P3.3 | $\overline{INT1}$ (external interrupt 1) |
| P3.4 | T0 (timer 0 external input) |
| P3.5 | T1 (timer 1 external input) |
| P3.6 | $\overline{WR}$ (external data memory write strobe) |
| P3.7 | $\overline{RD}$ (external data memory read strobe) |

$V_{CC}$: Supply voltage

$V_{SS}$: Circuit ground potential

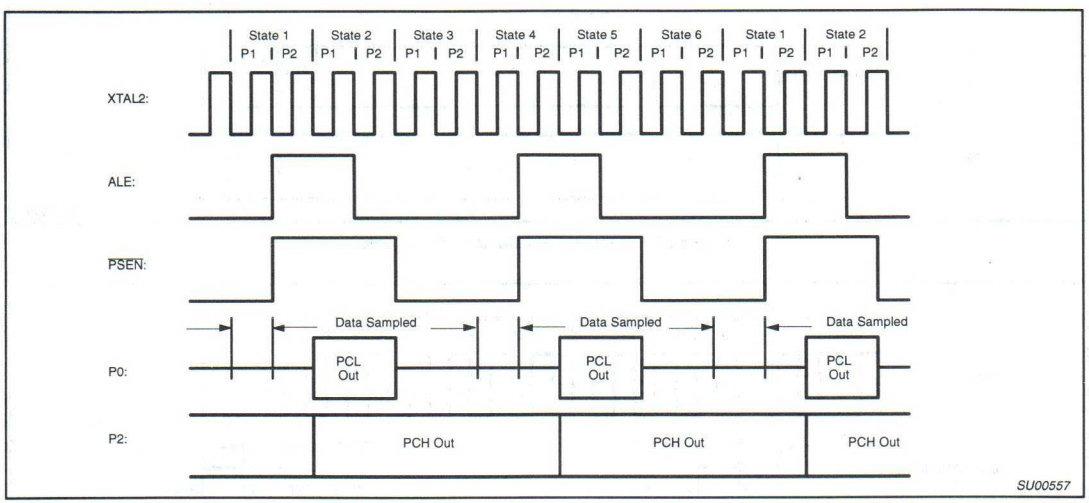



**Figure 31. External Program Memory Fetches**

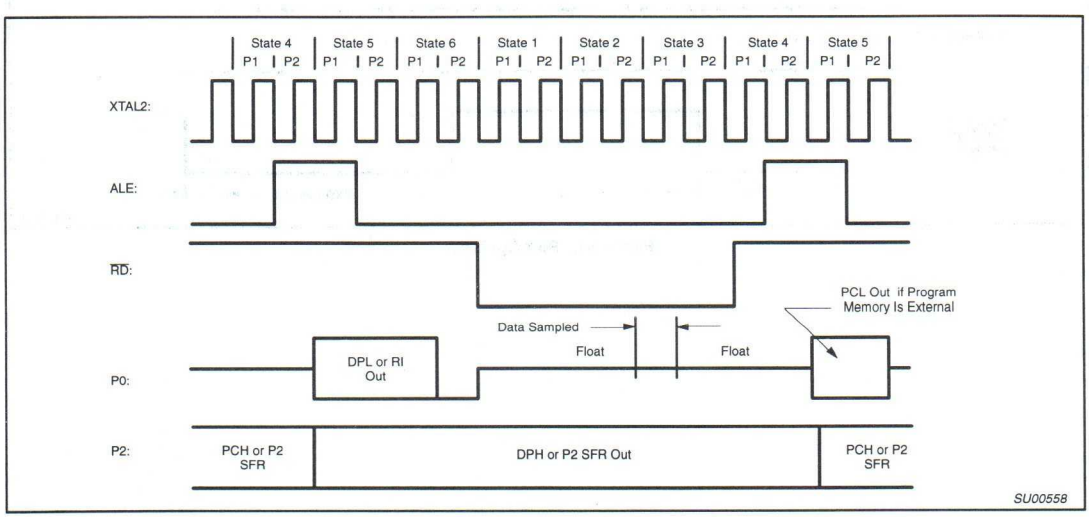



**Figure 32. External Data Memory Read Cycle**

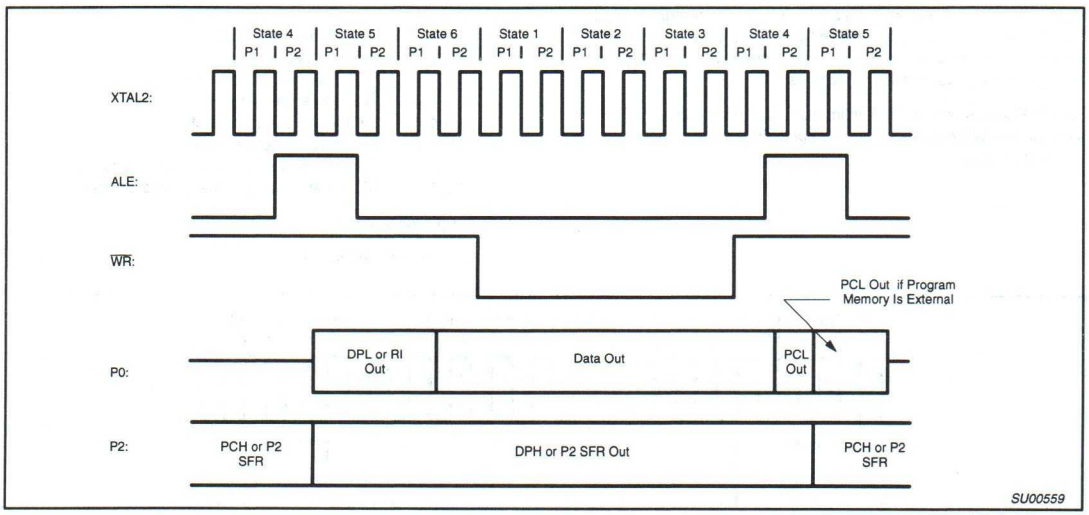



**Figure 33. External Data Memory Write Cycle**

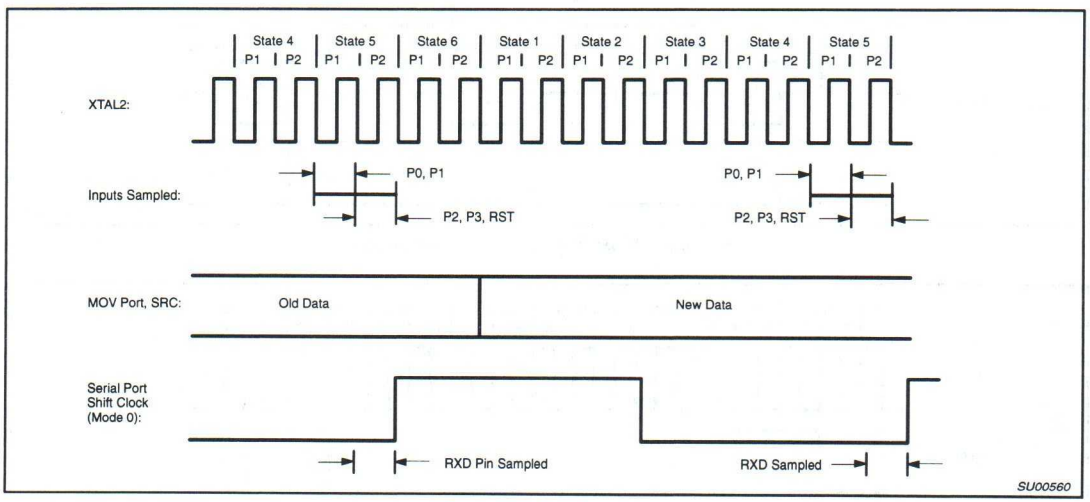



**Figure 34. Port Operation**

## PROGRAMMER'S GUIDE AND INSTRUCTION SET

### Memory Organization

**Program Memory**

The 80C51 has separate address spaces for program and data memory. The Program memory can be up to 64k bytes long. The lower 4k can reside on-chip. Figure 1 shows a map of the 80C51 program memory.

The 80C51 can address up to 64k bytes of data memory to the chip. The MOVX instruction is used to access the external data memory.

The 80C51 has 128 bytes of on-chip RAM, plus a number of Special Function Registers (SFRs). The lower 128 bytes of RAM can be accessed either by direct addressing (MOV data addr) or by indirect addressing (MOV @Ri). Figure 2 shows the Data Memory organization.

**Direct and Indirect Address Area**

The 128 bytes of RAM which can be accessed by both direct and indirect addressing can be divided into three segments as listed below and shown in Figure 3.

1. Register Banks 0-3: Locations 0 through 1FH (32 bytes). The device after reset defaults to register bank 0. To use the other register banks, the user must select them in software. Each register bank contains eight 1-byte registers 0 through 7. Reset initializes the stack pointer to location 07H, and it is incremented once to start from location 08H, which is the first register (R0) of the second register bank. Thus, in order to use more than one register bank, the SP should be initialized to a different location of the RAM where it is not used for data storage (i.e., the higher part of the RAM).
2. Bit Addressable Area: 16 bytes have been assigned for this segment, 20H-2FH. Each one of the 128 bits of this segment can be directly addressed (0-7FH). The bits can be referred to in two ways, both of which are acceptable by most assemblers. One way is to refer to their address (i.e., 0-7FH). The other way is with reference to bytes 20H to 2FH. Thus, bits 0-7 can also be referred to as bits 20.0-20.7, and bits 8-FH are the same as 21.0-21.7, and so on. Each of the 16 bytes in this segment can also be addressed as a byte.
3. Scratch Pad Area: 30H through 7FH are available to the user as data RAM. However, if the stack pointer has been initialized to this area, enough bytes should be left aside to prevent SP data destruction.

Figure 2 shows the different segments of the on-chip RAM.

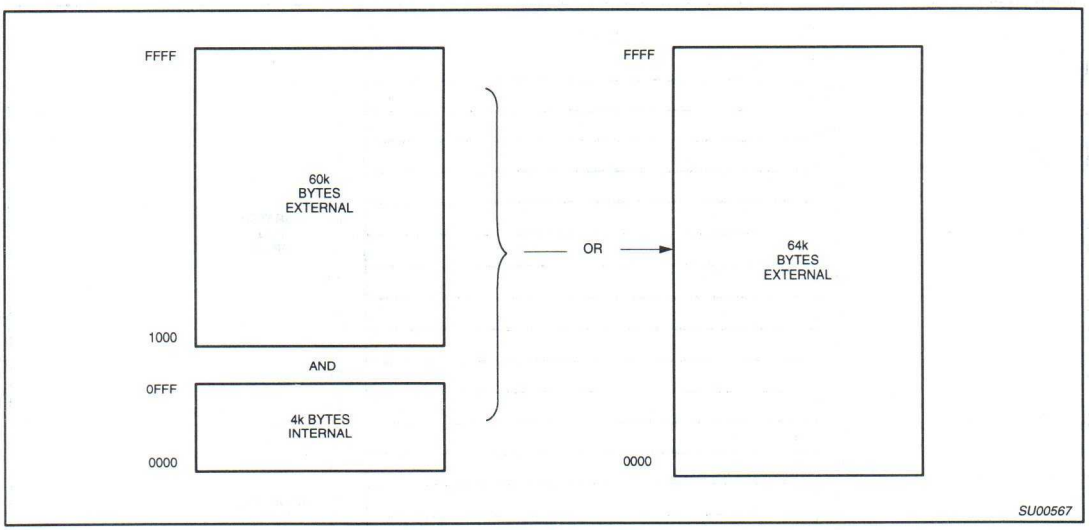



**Figure 1. 80C51 Program Memory**

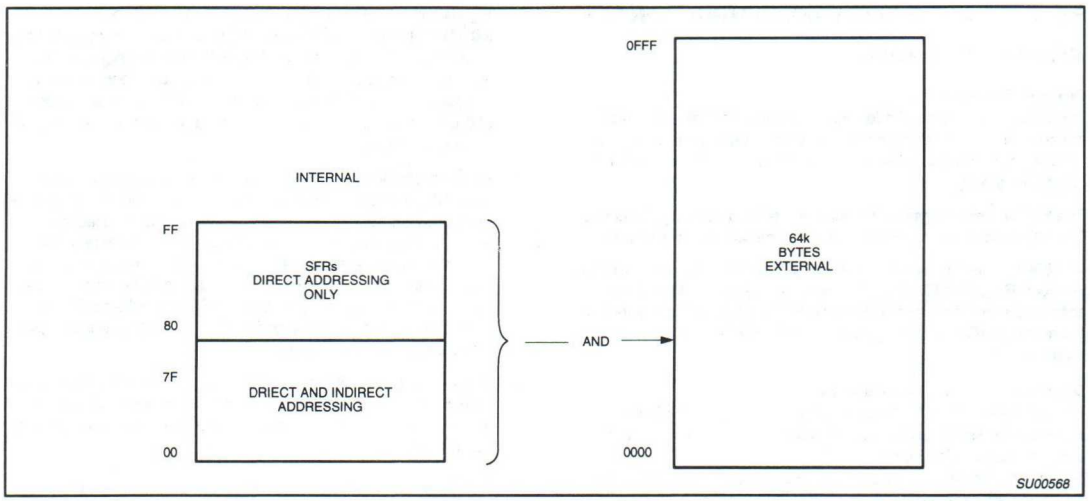



**Figure 2. 80C51 Data Memory**

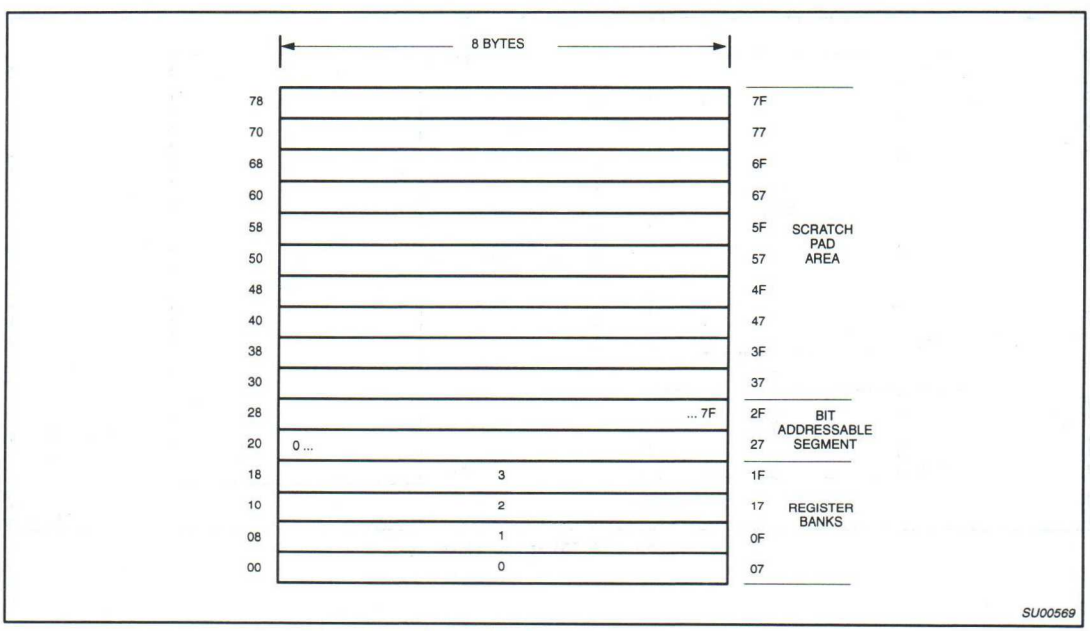



**Figure 3. 128 Bytes of RAM Direct and Indirect Addressable**

**Table 1. 80C51 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR | Data pointer (2 bytes) | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE* | Interrupt enable | A8H | EA | – | – | ES | ET1 | EX1 | ET0 | EX0 | 0x000000B |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP* | Interrupt priority | B8H | – | – | – | PS | PT1 | PX1 | PT0 | PX0 | xx000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | – | – | – | – | – | – | T2EX | T2 | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TxD | Rxd | FFH |
| PCON[1] | Power control | 87H | SMOD | – | – | – | GF1 | GF0 | PD | IDL | 0xxxxxxxB |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| SBUF | Serial data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial controller | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | |
| TH0 | Timer high 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer high 1 | 8DH | | | | | | | | | 00H |
| TL0 | Timer low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer low 1 | 8BH | | | | | | | | | 00H |
| TMOD | Timer mode | 89H | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 | 00H |

**NOTES:**
* Bit addressable
1. Bits GF1, GF0, PD, and IDL of the PCON register are not implemented on the NMOS 8051/8031.

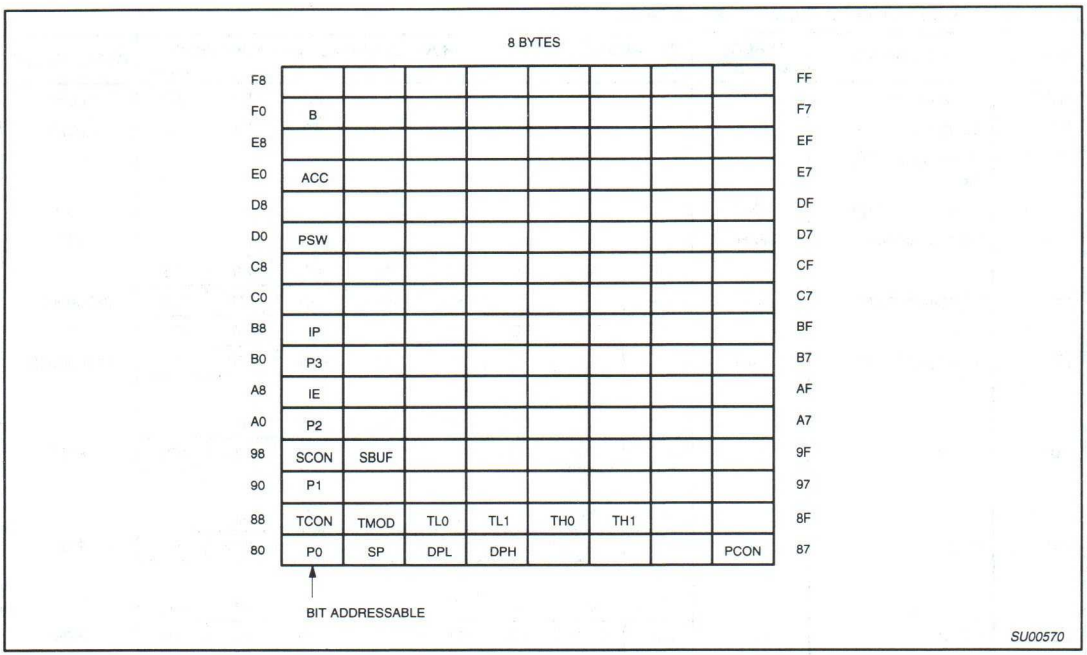



**Figure 4. SFR Memory Map**

Those SFRs that have their bits assigned for various functions are listed in this section. A brief description of each bit is provided for quick reference. For more detailed information refer to the Architecture Chapter of this book.

### PSW: PROGRAM STATUS WORD. BIT ADDRESSABLE.

| CY | AC | F0 | RS1 | RS0 | OV | – | P |
|---|---|---|---|---|---|---|---|

| | | |
|---|---|---|
| CY | PSW.7 | Carry Flag. |
| AC | PSW.6 | Auxiliary Carry Flag. |
| F0 | PSW.5 | Flag 0 available to the user for general purpose. |
| RS1 | PSW.4 | Register Bank selector bit 1 (SEE NOTE 1). |
| RS0 | PSW.3 | Register Bank selector bit 0 (SEE NOTE 1). |
| OV | PSW.2 | Overflow Flag. |
| – | PSW.1 | Usable as a general purpose flag. |
| P | PSW.0 | Parity flag. Set/cleared by hardware each instruction cycle to indicate an odd/even number of '1' bus in the accumulator. |

**NOTE:**
1. The value presented by RS0 and RS1 selects the corresponding register bank.

| RS1 | RS0 | REGISTER BANK | ADDRESS |
|---|---|---|---|
| 0 | 0 | 0 | 00H-07H |
| 0 | 1 | 1 | 08H-0FH |
| 1 | 0 | 2 | 10H-17H |
| 1 | 1 | 3 | 18H-1FH |

### PCON: POWER CONTROL REGISTER. NOT BIT ADDRESSABLE.

| SMOD | – | – | – | GF1 | GF0 | PD | IDL |
|---|---|---|---|---|---|---|---|

SMOD Double baud rate bit. If Timer 1 is used to generate baud rate and SMOD = 1, the baud rate is doubled when the Serial Port is used in modes 1, 2, or 3.

– Not implemented, reserved for future use.*

– Not implemented reserved for future use.*

– Not implemented reserved for future use.*

GF1 General purpose flag bit.

GF0 General purpose flag bit.

PD Power Down Bit. Setting this bit activates Power Down operation in the 80C51. (Available only in CMOS.)

IDL Idle mode bit. Setting this bit activates Idle Mode operation in the 80C51. (Available only in CMOS.)

If 1s are written to PD and IDL at the same time, PD takes precedence.

* User software should not write 1s to reserved bits. These bits may be used in future 8051 products to invoke new features.

## INTERRUPTS:

To use any of the interrupts in the 80C51 Family, the following three steps must be taken.

1. **Set the EA (enable all) bit in the IE register to 1.**
2. **Set the corresponding individual interrupt enable bit in the IE register to 1.**
3. **Begin the interrupt service routine at the corresponding Vector Address of that interrupt. See Table below.**

| INTERRUPT SOURCE | VECTOR ADDRESS |
|---|---|
| IE0 | 0003H |
| TF0 | 000BH |
| IE1 | 0013H |
| TF1 | 001BH |
| RI & TI | 0023H |

In addition, for external interrupts, pins INT0 and INT1 (P3.2 and P3.3) must be set to 1, and depending on whether the interrupt is to be level or transition activated, bits IT0 or IT1 in the TCON register may need to be set to 1.

ITx = 0 level activated

ITx = 1 transition activated

## IE: INTERRUPT ENABLE REGISTER. BIT ADDRESSABLE.

If the bit is 0, the corresponding interrupt is disabled. If the bit is 1, the corresponding interrupt is enabled.

| EA | – | – | ES | ET1 | EX1 | ET0 | EX0 |
|---|---|---|---|---|---|---|---|

| | | |
|---|---|---|
| EA | IE.7 | Disables all interrupts. If EA = 0, no interrupt will be acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit. |
| — | IE.6 | Not implemented, reserved for future use.* |
| — | IE.5 | Not implemented, reserved for future use.* |
| ES | IE.4 | Enable or disable the serial port interrupt. |
| ET1 | IE.3 | Enable or disable the Timer 1 overflow interrupt. |
| EX1 | IE.2 | Enable or disable External Interrupt 1. |
| ET0 | IE.1 | Enable or disable the Timer 0 overflow interrupt. |
| EX0 | IE.0 | Enable or disable External Interrupt 0. |

* User software should not write 1s to reserved bits. These bits may be used in future 80C51 products to invoke new features.

### ASSIGNING HIGHER PRIORITY TO ONE OR MORE INTERRUPTS:

In order to assign higher priority to an interrupt the corresponding bit in the IP register must be set to 1.

Remember that while an interrupt service is in progress, it cannot be interrupted by a lower or same level interrupt.

### PRIORITY WITHIN LEVEL:

Priority within level is only to resolve simultaneous requests of the same priority level.

From high to low, interrupt sources are listed below:

IE0
TF0
IE1
TF1
RI or TI

### IP: INTERRUPT PRIORITY REGISTER. BIT ADDRESSABLE.

If the bit is 0, the corresponding interrupt has a lower priority and if the bit is 1 the corresponding interrupt has a higher priority.

| – | – | – | PS | PT1 | PX1 | PT0 | PX0 |
|---|---|---|---|---|---|---|---|

| | | |
|---|---|---|
| – | IP.7 | Not implemented, reserved for future use.* |
| – | IP.6 | Not implemented, reserved for future use.* |
| – | IP.5 | Not implemented, reserved for future use.* |
| PS | IP.4 | Defines the Serial Port interrupt priority level. |
| PT1 | IP.3 | Defines the Timer 1 interrupt priority level. |
| PX1 | IP.2 | Defines External Interrupt 1 priority level. |
| PT0 | IP.1 | Defines the Timer 0 interrupt priority level. |
| PX0 | IP.0 | Defines the External Interrupt 0 priority level. |

* User software should not write 1s to reserved bits. These bits may be used in future 80C51 products to invoke new features.

### TCON: TIMER/COUNTER CONTROL REGISTER. BIT ADDRESSABLE.

| TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |
|---|---|---|---|---|---|---|---|

| | | |
|---|---|---|
| TF1 | TCON.7 | Timer 1 overflow flag. Set by hardware when the Timer/Counter 1 overflows. Cleared by hardware as processor vectors to the interrupt service routine. |
| TR1 | TCON.6 | Timer 1 run control bit. Set/cleared by software to turn Timer/Counter 1 ON/OFF. |
| TF0 | TCON.5 | Timer 0 overflow flag. Set by hardware when the Timer/Counter 0 overflows. Cleared by hardware as processor vectors to the service routine. |
| TR0 | TCON.4 | Timer 0 run control bit. Set/cleared by software to turn Timer/Counter 0 ON/OFF. |
| IE1 | TCON.3 | External Interrupt 1 edge flag. Set by hardware when External Interrupt edge is detected. Cleared by hardware when interrupt is processed. |
| IT1 | TCON.2 | Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered External Interrupt. |
| IE0 | TCON.1 | External Interrupt 0 edge flag. Set by hardware when External Interrupt edge detected. Cleared by hardware when interrupt is processed. |
| IT0 | TCON.0 | Interrupt 0 type control bit. Set/cleared by software to specify falling edge/low level triggered External Interrupt. |

### TMOD: TIMER/COUNTER MODE CONTROL REGISTER. NOT BIT ADDRESSABLE.

| GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 |
|---|---|---|---|---|---|---|---|
| Timer 1 | | | | Timer 0 | | | |

| | |
|---|---|
| GATE | When TRx (in TCON) is set and GATE = 1, TIMER/COUNTERx will run only while INTx pin is high (hardware control). When GATE = 0, TIMER/COUNTERx will run only while TRx = 1 (software control). |
| C/$\overline{T}$ | Timer or Counter selector. Cleared for Timer operation (input from internal system clock). Set for Counter operation (input from Tx input pin). |
| M1 | Mode selector bit. (NOTE 1) |
| M0 | Mode selector bit. (NOTE 1) |

**NOTE 1:**

| M1 | M0 | | Operating Mode |
|---|---|---|---|
| 0 | 0 | 0 | 13-bit Timer (8048 compatible) |
| 0 | 1 | 1 | 16-bit Timer/Counter |
| 1 | 0 | 2 | 8-bit Auto-Reload Timer/Counter |
| 1 | 1 | 3 | (Timer 0) TL0 is an 8-bit Timer/Counter controlled by the standart Timer 0 control bits. TH0 is an8-bit Timer and is controlled by Timer 1 control bits. |
| 1 | 1 | 3 | (Timer 1) Timer/Counter 1 stopped. |

## TIMER SET-UP

Tables 2 through 5 give some values for TMOD which can be used to set up Timer 0 in different modes.

It is assumed that only one timer is being used at a time. If it is desired to run Timers 0 and 1 simultaneously, in any mode, the value in TMOD for Timer 0 must be ORed with the value shown for Timer 1 (Tables 5 and 6).

For example, if it is desired to run Timer 0 in mode 1 GATE (external control), and Timer 1 in mode 2 COUNTER, then the value that must be loaded into TMOD is 69H (09H from Table 2 ORed with 60H from Table 5).

Moreover, it is assumed that the user, at this point, is not ready to turn the timers on and will do that at a different point in the program by setting bit TRx (in TCON) to 1.

## TIMER/COUNTER 0

**Table 2. As a Timer:**

| MODE | TIMER 0 FUNCTION | TMOD | |
|---|---|---|---|
| | | INTERNAL CONTROL (NOTE 1) | EXTERNAL CONTROL (NOTE 2) |
| 0 | 13-bit Timer | 00H | 08H |
| 1 | 16-bit Timer | 01H | 09H |
| 2 | 8-bit Auto-Reload | 02H | 0AH |
| 3 | Two 8-bit Timers | 03H | 0BH |

**Table 3. As a Counter:**

| MODE | COUNTER 0 FUNCTION | TMOD | |
|---|---|---|---|
| | | INTERNAL CONTROL (NOTE 1) | EXTERNAL CONTROL (NOTE 2) |
| 0 | 13-bit Timer | 04H | 0CH |
| 1 | 16-bit Timer | 05H | 0DH |
| 2 | 8-bit Auto-Reload | 06H | 0EH |
| 3 | One 8-bit Counter | 07H | 0FH |

**NOTES:**
1. The timer is turned ON/OFF by setting/clearing bit TR0 in the software.
2. The Timer is turned ON/OFF by the 1-to-0 transition on $\overline{INT0}$ (P3.2) when TR0 = 1 (hardware control).

## TIMER/COUNTER 1

**Table 4. As a Timer:**

| MODE | TIMER 1 FUNCTION | TMOD | |
|---|---|---|---|
| | | INTERNAL CONTROL (NOTE 1) | EXTERNAL CONTROL (NOTE 2) |
| 0 | 13-bit Timer | 00H | 80H |
| 1 | 16-bit Timer | 10H | 90H |
| 2 | 8-bit Auto-Reload | 20H | A0H |
| 3 | Does not run | 30H | B0H |

**Table 5. As a Counter:**

| MODE | COUNTER 1 FUNCTION | TMOD | |
|---|---|---|---|
| | | INTERNAL CONTROL (NOTE 1) | EXTERNAL CONTROL (NOTE 2) |
| 0 | 13-bit Timer | 40H | C0H |
| 1 | 16-bit Timer | 50H | D0H |
| 2 | 8-bit Auto-Reload | 60H | E0H |
| 3 | Not available | – | – |

**NOTES:**
1. The timer is turned ON/OFF by setting/clearing bit TR1 in the software.
2. The Timer is turned ON/OFF by the 1-to-0 transition on $\overline{INT1}$ (P3.2) when TR1 = 1 (hardware control).

## SCON: SERIAL PORT CONTROL REGISTER. BIT ADDRESSABLE.

| SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |
|---|---|---|---|---|---|---|---|

| | | |
|---|---|---|
| SM0 | SCON.7 | Serial Port mode specifier. (NOTE 1) |
| SM1 | SCON.6 | Serial Port mode specifier. (NOTE 1) |
| SM2 | SCON.5 | Enables the multiprocessor communication feature in modes 2 & 3. In mode 2 or 3, if SM2 is set to 1 then RI will not be activated if the received 9th data bit (RB8) is 0. In mode 1, if SM2 = 1 then RI will not be activated if a valid stop bit was not received. In mode 0, SM2 should be 0. (See Table 6.) |
| REN | SCON.4 | Set/Cleared by software to Enable/Disable reception. |
| TB8 | SCON.3 | The 9th bit that will be transmitted in modes 2 & 3. Set/Cleared by software. |
| RB8 | SCON.2 | In modes 2 & 3, is the 9th data bit that was received. In mode 1, if SM2 = 0, RB8 is the stop bit that was received. In mode 0, RB8 is not used. |
| TI | SCON.1 | Transmit interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or at the beginning of the stop bit in the other modes. Must be cleared by software. |
| RI | SCON.0 | Receive interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or halfway through the stop bit time in the other modes (except see SM2). Must be cleared by software. |

**NOTE 1:**

| SM0 | SM1 | Mode | Description | Baud Rate |
|---|---|---|---|---|
| 0 | 0 | 0 | Shift Register | $F_{OSC.}/12$ |
| 0 | 1 | 1 | 8-bit UART | Variable |
| 1 | 0 | 2 | 9-bit UART | $F_{OSC.}/64$ or $F_{OSC.}/32$ |
| 1 | 1 | 3 | 9-bit UART | Variable |

## SERIAL PORT SET-UP:

**Table 6.**

| MODE | SCON | SM2 VARIATION |
|---|---|---|
| 0<br>1<br>2<br>3 | 10H<br>50H<br>90H<br>D0H | Single Processor Environment (SM2 = 0) |
| 0<br>1<br>2<br>3 | NA<br>70H<br>B0H<br>F0H | Multiprocessor Environment (SM2 = 1) |

## GENERATING BAUD RATES

### Serial Port in Mode 0:

Mode 0 has a fixed baud rate which is 1/12 of the oscillator frequency. To run the serial port in this mode none of the Timer/Counters need to be set up. Only the SCON register needs to be defined.

$$\text{Baud Rate} = \frac{\text{Osc Freq}}{12}$$

### Serial Port in Mode 1:

Mode 1 has a variable baud rate. The baud rate is generated by Timer 1.

### USING TIMER/COUNTER 1 TO GENERATE BAUD RATES:

For this purpose, Timer 1 is used in mode 2 (Auto-Reload). Refer to Timer Setup section of this chapter.

$$\text{Baud Rate} = \frac{K \times \text{Osc Freq}}{32 \times 12 \times [256 - (TH1)]}$$

If SMOD = 0, then K = 1.
If SMOD = 1, then K = 2 (SMOD is in the PCON register).

Most of the time the user knows the baud rate and needs to know the reload value for TH1.

$$TH1 = 256 - \frac{K \times \text{Osc Freq}}{384 \times \text{baud rate}}$$

TH1 must be an integer value. Rounding off TH1 to the nearest integer may not produce the desired baud rate. In this case, the user may have to choose another crystal frequency.

Since the PCON register is not bit addressable, one way to set the bit is logical ORing the PCON register (i.e., ORL PCON,#80H). The address of PCON is 87H.

### SERIAL PORT IN MODE 2:

The baud rate is fixed in this mode and is 1/32 or 1/64 of the oscillator frequency, depending on the value of the SMOD bit in the PCON register.

In this mode none of the Timers are used and the clock comes from the internal phase 2 clock.

SMOD = 1, Baud Rate = 1/32 Osc Freq.

SMOD = 0, Baud Rate = 1/64 Osc Freq.

To set the SMOD bit: ORL PCON,#80H. The address of PCON is 87H.

### SERIAL PORT IN MODE 3:

The baud rate in mode 3 is variable and sets up exactly the same as in mode 1.

## 80C51 FAMILY INSTRUCTION SET

### Table 7. 80C51 Instruction Set Summary

Interrupt Response Time: Refer to Hardware Description Chapter.

**Instructions that Affect Flag Settings[1]**

| Instruction | Flag | | | Instruction | Flag | | |
|---|---|---|---|---|---|---|---|
| | C | OV | AC | | C | OV | AC |
| ADD | X | X | X | CLR C | 0 | | |
| ADDC | X | X | X | CPL C | X | | |
| SUBB | X | X | X | ANL C,bit | X | | |
| MUL | 0 | X | | ANL C,/bit | X | | |
| DIV | 0 | X | | ORL C,bit | X | | |
| DA | X | | | ORL C,/bit | X | | |
| RRC | X | | | MOV C,bit | X | | |
| RLC | X | | | CJNE | X | | |
| SETB C | 1 | | | | | | |

[1]Note that operations on SFR byte address 208 or bit addresses 209-215 (i.e., the PSW or bits in the PSW) will also affect flag settings.

**Notes on instruction set and addressing modes:**

| | |
|---|---|
| Rn | Register R7-R0 of the currently selected Register Bank. |
| direct | 8-bit internal data location's address. This could be an Internal Data RAM location (0-127) or a SFR [i.e., I/O port, control register, status register, etc. (128-255)]. |
| @Ri | 8-bit internal data RAM location (0-255) addressed indirectly through register R1 or R0. |
| #data | 8-bit constant included in the instruction. |
| #data 16 | 16-bit constant included in the instruction |
| addr 16 | 16-bit destination address. Used by LCALL and LJMP. A branch can be anywhere within the 64k-byte Program Memory address space. |
| addr 11 | 11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2k-byte page of program memory as the first byte of the following instruction. |
| rel | Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is –128 to +127 bytes relative to first byte of the following instruction. |
| bit | Direct Addressed bit in Internal Data RAM or Special Function Register. |

| MNEMONIC | | DESCRIPTION | BYTE | OSCILLATOR PERIOD |
|---|---|---|---|---|
| **ARITHMETIC OPERATIONS** | | | | |
| ADD | A,Rn | Add register to Accumulator | 1 | 12 |
| ADD | A,direct | Add direct byte to Accumulator | 2 | 12 |
| ADD | A,@Ri | Add indirect RAM to Accumulator | 1 | 12 |
| ADD | A,#data | Add immediate data to Accumulator | 2 | 12 |
| ADDC | A,Rn | Add register to Accumulator with carry | 1 | 12 |
| ADDC | A,direct | Add direct byte to Accumulator with carry | 2 | 12 |
| ADDC | A,@Ri | Add indirect RAM to Accumulator with carry | 1 | 12 |
| ADDC | A,#data | Add immediate data to $A_{CC}$ with carry | 2 | 12 |
| SUBB | A,Rn | Subtract Register from $A_{CC}$ with borrow | 1 | 12 |
| SUBB | A,direct | Subtract direct byte from $A_{CC}$ with borrow | 2 | 12 |
| SUBB | A,@Ri | Subtract indirect RAM from $A_{CC}$ with borrow | 1 | 12 |
| SUBB | A,#data | Subtract immediate data from $A_{CC}$ with borrow | 2 | 12 |
| INC | A | Increment Accumulator | 1 | 12 |
| INC | Rn | Increment register | 1 | 12 |

All mnemonics copyrighted © Intel Corporation 1980

**Table 7. 80C51 Instruction Set Summary** (Continued)

| MNEMONIC | | DESCRIPTION | BYTE | OSCILLATOR PERIOD |
|---|---|---|---|---|
| **ARITHMETIC OPERATIONS** (Continued) | | | | |
| INC | direct | Increment direct byte | 2 | 12 |
| INC | @Ri | Increment indirect RAM | 1 | 12 |
| DEC | A | Decrement Accumulator | 1 | 12 |
| DEC | Rn | Decrement Register | 1 | 12 |
| DEC | direct | Decrement direct byte | 2 | 12 |
| DEC | @Ri | Decrement indirect RAM | 1 | 12 |
| INC | DPTR | Increment Data Pointer | 1 | 24 |
| MUL | AB | Multiply A and B | 1 | 48 |
| DIV | AB | Divide A by B | 1 | 48 |
| DA | A | Decimal Adjust Accumulator | 1 | 12 |
| **LOGICAL OPERATIONS** | | | | |
| ANL | A,Rn | AND Register to Accumulator | 1 | 12 |
| ANL | A,direct | AND direct byte to Accumulator | 2 | 12 |
| ANL | A,@Ri | AND indirect RAM to Accumulator | 1 | 12 |
| ANL | A,#data | AND immediate data to Accumulator | 2 | 12 |
| ANL | direct,A | AND Accumulator to direct byte | 2 | 12 |
| ANL | direct,#data | AND immediate data to direct byte | 3 | 24 |
| ORL | A,Rn | OR register to Accumulator | 1 | 12 |
| ORL | A,direct | OR direct byte to Accumulator | 2 | 12 |
| ORL | A,@Ri | OR indirect RAM to Accumulator | 1 | 12 |
| ORL | A,#data | OR immediate data to Accumulator | 2 | 12 |
| ORL | direct,A | OR Accumulator to direct byte | 2 | 12 |
| ORL | direct,#data | OR immediate data to direct byte | 3 | 24 |
| XRL | A,Rn | Exclusive-OR register to Accumulator | 1 | 12 |
| XRL | A,direct | Exclusive-OR direct byte to Accumulator | 2 | 12 |
| XRL | A,@Ri | Exclusive-OR indirect RAM to Accumulator | 1 | 12 |
| XRL | A,#data | Exclusive-OR immediate data to Accumulator | 2 | 12 |
| XRL | direct,A | Exclusive-OR Accumulator to direct byte | 2 | 12 |
| XRL | direct,#data | Exclusive-OR immediate data to direct byte | 3 | 24 |
| CLR | A | Clear Accumulator | 1 | 12 |
| CPL | A | Complement Accumulator | 1 | 12 |
| RL | A | Rotate Accumulator left | 1 | 12 |
| RLC | A | Rotate Accumulator left through the carry | 1 | 12 |
| RR | A | Rotate Accumulator right | 1 | 12 |
| RRC | A | Rotate Accumulator right through the carry | 1 | 12 |
| SWAP | A | Swap nibbles within the Accumulator | 1 | 12 |
| **DATA TRANSFER** | | | | |
| MOV | A,Rn | Move register to Accumulator | 1 | 12 |
| MOV | A,direct | Move direct byte to Accumulator | 2 | 12 |
| MOV | A,@Ri | Move indirect RAM to Accumulator | 1 | 12 |

All mnemonics copyrighted © Intel Corporation 1980

**Table 7. 80C51 Instruction Set Summary** (Continued)

| MNEMONIC | | DESCRIPTION | BYTE | OSCILLATOR PERIOD |
|---|---|---|---|---|
| **DATA TRANSFER** (Continued) | | | | |
| MOV | A,#data | Move immediate data to Accumulator | 2 | 12 |
| MOV | Rn,A | Move Accumulator to register | 1 | 12 |
| MOV | Rn,direct | Move direct byte to register | 2 | 24 |
| MOV | RN,#data | Move immediate data to register | 2 | 12 |
| MOV | direct,A | Move Accumulator to direct byte | 2 | 12 |
| MOV | direct,Rn | Move register to direct byte | 2 | 24 |
| MOV | direct,direct | Move direct byte to direct | 3 | 24 |
| MOV | direct,@Ri | Move indirect RAM to direct byte | 2 | 24 |
| MOV | direct,#data | Move immediate data to direct byte | 3 | 24 |
| MOV | @Ri,A | Move Accumulator to indirect RAM | 1 | 12 |
| MOV | @Ri,direct | Move direct byte to indirect RAM | 2 | 24 |
| MOV | @Ri,#data | Move immediate data to indirect RAM | 2 | 12 |
| MOV | DPTR,#data16 | Load Data Pointer with a 16-bit constant | 3 | 24 |
| MOVC | A,@A+DPTR | Move Code byte relative to DPTR to $A_{CC}$ | 1 | 24 |
| MOVC | A,@A+PC | Move Code byte relative to PC to $A_{CC}$ | 1 | 24 |
| MOVX | A,@Ri | Move external RAM (8-bit addr) to $A_{CC}$ | 1 | 24 |
| MOVX | A,@DPTR | Move external RAM (16-bit addr) to $A_{CC}$ | 1 | 24 |
| MOVX | A,@Ri,A | Move $A_{CC}$ to external RAM (8-bit addr) | 1 | 24 |
| MOVX | @DPTR,A | Move $A_{CC}$ to external RAM (16-bit addr) | 1 | 24 |
| PUSH | direct | Push direct byte onto stack | 2 | 24 |
| POP | direct | Pop direct byte from stack | 2 | 24 |
| XCH | A,Rn | Exchange register with Accumulator | 1 | 12 |
| XCH | A,direct | Exchange direct byte with Accumulator | 2 | 12 |
| XCH | A,@Ri | Exchange indirect RAM with Accumulator | 1 | 12 |
| XCHD | A,@Ri | Exchange low-order digit indirect RAM with $A_{CC}$ | 1 | 12 |
| **BOOLEAN VARIABLE MANIPULATION** | | | | |
| CLR | C | Clear carry | 1 | 12 |
| CLR | bit | Clear direct bit | 2 | 12 |
| SETB | C | Set carry | 1 | 12 |
| SETB | bit | Set direct bit | 2 | 12 |
| CPL | C | Complement carry | 1 | 12 |
| CPL | bit | Complement direct bit | 2 | 12 |
| ANL | C,bit | AND direct bit to carry | 2 | 24 |
| ANL | C,/bit | AND complement of direct bit to carry | 2 | 24 |
| ORL | C,bit | OR direct bit to carry | 2 | 24 |
| ORL | C,/bit | OR complement of direct bit to carry | 2 | 24 |
| MOV | C,bit | Move direct bit to carry | 2 | 12 |
| MOV | bit,C | Move carry to direct bit | 2 | 24 |
| JC | rel | Jump if carry is set | 2 | 24 |
| JNC | rel | Jump if carry not set | 2 | 24 |

All mnemonics copyrighted © Intel Corporation 1980

**Table 7. 80C51 Instruction Set Summary** (Continued)

| MNEMONIC | | DESCRIPTION | BYTE | OSCILLATOR PERIOD |
|---|---|---|---|---|
| **BOOLEAN VARIABLE MANIPULATION** (Continued) | | | | |
| JB | rel | Jump if direct bit is set | 3 | 24 |
| JNB | rel | Jump if direct bit is not set | 3 | 24 |
| JBC | bit,rel | Jump if direct bit is set and clear bit | 3 | 24 |
| **PROGRAM BRANCHING** | | | | |
| ACALL | addr11 | Absolute subroutine call | 2 | 24 |
| LCALL | addr16 | Long subroutine call | 3 | 24 |
| RET | | Return from subroutine | 1 | 24 |
| RETI | | Return from interrupt | 1 | 24 |
| AJMP | addr11 | Absolute jump | 2 | 24 |
| LJMP | addr16 | Long jump | 3 | 24 |
| SJMP | rel | Short jump (relative addr) | 2 | 24 |
| JMP | @A+DPTR | Jump indirect relative to the DPTR | 1 | 24 |
| JZ | rel | Jump if Accumulator is zero | 2 | 24 |
| JNZ | rel | Jump if Accumulator is not zero | 2 | 24 |
| CJNE | A,direct,rel | Compare direct byte to $A_{CC}$ and jump if not equal | 3 | 24 |
| CJNE | A,#data,rel | Compare immediate to $A_{CC}$ and jump if not equal | 3 | 24 |
| CJNE | RN,#data,rel | Compare immediate to register and jump if not equal | 3 | 24 |
| CJNE | @Ri,#data,rel | Compare immediate to indirect and jump if not equal | 3 | 24 |
| DJNZ | Rn,rel | Decrement register and jump if not zero | 2 | 24 |
| DJNZ | direct,rel | Decrement direct byte and jump if not zero | 3 | 24 |
| NOP | | No operation | 1 | 12 |

All mnemonics copyrighted © Intel Corporation 1980

## INSTRUCTION DEFINITIONS

**ACALL addr11**

**Function:** Absolute Call

**Description:** ACALL unconditionally calls a subroutine located at the indicated address. The instruction increments the PC twice to obtain the address of the following instruction, then pushes the 16-bit result onto the stack (low-order byte first) and increments the Stack Pointer twice. The destination address is obtained by successively concatenating the five high-order bits of the incremented PC, opcode bits 7-5, and the second byte of the instruction. The subroutine called must therefore start within the same 2k block of the program memory as the first byte of the instruction following ACALL. No flags are affected.

**Example:** Initially SP equals 07H. The label "SUBRTN" is at program memory location 0345 H. After executing the instruction,

ACALL SUBRTN

at location 0123H, SP will contain 09H, internal RAM locations 08H and 09H will contain 25H and 01H, respectively, and the PC will contain 0345H.

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| a10 a9 a8 1 | 0 0 0 1 | a7 a6 a5 a4 | a3 a2 a1 a0 |
|---|---|---|---|

**Operation:** ACALL

(PC) ← (PC) + 2
(SP) ← (SP) + 1
(SP) ← ($PC_{7-0}$)
(SP) ← (SP) + 1
(SP) ← ($PC_{15-8}$)
($PC_{10-0}$) ← page address

**ADD A,<src-byte>**

**Function:** Add

**Description:** ADD adds the byte variable indicated to the Accumulator, leaving the result in the Accumulator. The carry and auxiliary-carry flags are set, respectively, if there is a carry-out from bit 7 or bit 3, and cleared otherwise. When adding unsigned integers, the carry flag indicates an overflow occurred.

OV is set if there is a carry-out of bit 6 but not out of bit 7, or a carry-out of bit 7 but not bit 6; otherwise OV is cleared. When adding signed integers, OV indicates a negative number produced as the sum of two positive operands, or a positive sum from two negative operands.

Four source operand addressing modes are allowed: register, direct, register-indirect, or immediate.

**Example:** The Accumulator holds 0C3H (11000011B) and register 0 holds 0AAH (10101010B). The instruction,

ADD A,R0

will leave 6DH (01101101B) in the Accumulator with the AC flag cleared and both the Carry flag and OV set to 1.

**ADD A,Rn**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 0 1 0 | 1 r r r |
|---|---|

**Operation:** ADD

$(A) \leftarrow (A) + (R_n)$

**ADD A,direct**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 0 1 0 | 0 1 0 1 | direct address |
|---|---|---|

**Operation:** ADD

(A) ← (A) + (direct)

**ADD A,@Ri**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 0 1 0 | 0 1 1 i |
|---|---|

**Operation:** ADD

$(A) \leftarrow (A) + ((R_i))$

**ADD A,#data**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 0 1 0 | 0 1 0 0 | immediate data |
|---|---|---|

**Operation:** ADD

(A) ← (A) + #data

**ADDC A,<src-byte>**

**Function:** Add with Carry

**Description:** ADDC simultaneously adds the byte variable indicated, the carry flag and the Accumulator contents, leaving the result in the Accumulator. The carry and auxiliary-carry flags are set, respectively, if there is a carry-out from bit 7 or bit 3, and cleared otherwise. When adding unsigned integers, the carry flag indicates an overflow occurred.

OV is set if there is a carry-out of bit 6 but not out of bit 7, or a carry-out of bit 7 but not out of bit 6; otherwise OV is cleared. When adding signed integers, OV indicates a negative number produced as the sum of two positive operands, or a positive sum from two negative operands.

Four source operand addressing modes are allowed: register, direct, register-indirect, or immediate.

**Example:** The Accumulator holds 0C3H (11000011B) and register 0 holds 0AAH (10101010B) with the carry flag set. The instruction,

ADDC A,R0

will leave 6EH (01101110B) in the Accumulator with AC cleared and both the Carry flag and OV set to 1.

**ADDC A,Rn**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 0 1 1 | 1 r r r |
|---|---|

**Operation:** ADDC

(A) ← (A) + (C) + ($R_n$)

**ADDC A,direct**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 0 1 1 | 0 1 0 1 | direct address |
|---|---|---|

**Operation:** ADDC

(A) ← (A) + (C) + (direct)

**ADDC A,@Ri**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 0 1 1 | 0 1 1 i |
|---|---|

**Operation:** ADDC

(A) ← (A) + (C) + (($R_i$))

**ADDC A,#data**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 0 1 1 | 0 1 0 0 | immediate data |
|---|---|---|

**Operation:** ADDC

(A) ← (A) + (C) + #data

**AJMP addr11**

**Function:** Absolute Jump

**Description:** AJMP transfers program execution to the indicated address, which is formed at run-time by concatenating the high-order five bits of the PC (*after* incrementing the PC twice), opcode bits 7-5, and the second byte of the instruction. The destination must therefore be within the same 2k block of program memory as the first byte of the instruction following AJMP.

**Example:** The label "JMPADR" is at program memory location 0123H. The instruction,

AJMP JMPADR

is at location 0345H and will load the PC with 0123H.

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| a10 a9 a8 0 | 0 0 0 1 | a7 a6 a5 a4 | a3 a2 a1 a0 |
|---|---|---|---|

**Operation:** AJMP
(PC) ← (PC) + 2
$(PC_{10\text{-}0})$ ← page address

**ANL <dest-byte>,<src-byte>**

**Function:** Logical-AND for byte variables

**Description:** ANL performs the bitwise logical-AND operation between the variables indicated and stores the results in the destination variable. No flags are affected.

The two operands allow six addressing mode combinations. When the destination is the Accumulator, the source can use register, direct, register-indirect, or immediate addressing; when the destination is a direct address, the source can be the Accumulator or immediate data.

*Note:* When this instruction is used to modify an output port, the value used as the original port data will be read from the output data latch, *not* the input pins.

**Example:** If the Accumulator holds 0C3H (11000011B) and register 0 holds 55H (01010101B) then the instruction,

ANL A,R0

will leave 41H (01000001B) in the Accumulator.

When the destination is a directly addressed byte, this instruction will clear combinations of bits in any RAM location or hardware register. The mask byte determining the pattern of bits to be cleared would either be a constant contained in the instruction or a value computed in the Accumulator at run-time. The instruction,

ANL P1,#01110011B

will clear bits 7, 3, and 2 of output port 1.

**ANL A,Rn**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 1 0 1 | 1 r r r |
|---|---|

**Operation:** ANL
$(A) \leftarrow (A) \wedge (R_n)$

**ANL A,direct**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 1 0 1 | 0 1 0 1 | direct address |
|---|---|---|

**Operation:** ANL
$(A) \leftarrow (A) \wedge$ (direct)

**ANL A,@Ri**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 1 0 1 | 0 1 1 i |
|---|---|

**Operation:** ANL
$(A) \leftarrow (A) \wedge ((R_i))$

**ANL A,#data**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 1 0 1 | 0 1 0 0 | immediate data |
|---|---|---|

**Operation:** ANL
$(A) \leftarrow (A) \wedge$ #data

**ANL direct,A**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 1 0 1 | 0 0 1 0 | direct address |
|---|---|---|

**Operation:** ANL
$(A) \leftarrow$ (direct) $\wedge$ (A)

**ANL direct,#data**

**Bytes:** 3

**Cycles:** 2

**Encoding:**

| 0 1 0 1 | 0 0 1 1 | direct address | immediate data |
|---|---|---|---|

**Operation:** ANL
(direct) $\leftarrow$ (direct) $\wedge$ #data

**ANL C,<src-bit>**

**Function:** Logical-AND for bit variables

**Description:** If the Boolean value of the source bit is a logical 0 then clear the carry flag; otherwise leave the carry flag in its current state. A slash ("/") preceding the operand in the assembly language indicates that the logical complement of the addressed bit is used as the source value, *but the source bit itself is not affected.* No other flags are affected.

Only direct addressing is allowed for the source operand.

**Example:** Set the carry flag if, and only if, P1.0 = 1, ACC.7 = 1, and OV = 0:

```
MOV     C,P1.0  ;LOAD CARRY WITH INPUT PIN STATE
ANL     C,ACC.7;AND CARRY WITH ACCUM. BIT 7
ANL     C,/OV   ;AND WITH INVERSE OF OVERFLOW FLAG
```

**ANL C,bit**

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| 1 0 0 0 | 0 0 1 0 | bit address |
|---|---|---|

**Operation:** ANL

(C) ← (C) ∧ (bit)

**ANL C,/bit**

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| 1 0 1 1 | 0 0 0 0 | bit address |
|---|---|---|

**Operation:** ANL

(C) ← (C) ∧ ⌝(bit)

**CJNE <dest-byte>,<src-byte>,rel**

**Function:** Compare and Jump if Not Equal

**Description:** CJNE compares the magnitudes of the first two operands, and branches if their values are not equal. The branch destination is computed by adding the signed relative-displacement in the last instruction byte to the PC, after incrementing the PC to the start of the next instruction. The carry flag is set if the unsigned integer value of <dest-byte> is less than the unsigned integer value of <src-byte>; otherwise, the carry is cleared. Neither operand is affected.

The first two operands allow four addressing mode combinations: the Accumulator may be compared with any directly addressed byte or immediate data, and any indirect RAM location or working register can be compared with an immediate constant.

**Example:** The Accumulator contains 34H. Register 7 contains 56H. The first instruction in the sequence,

```
        CJNE    R7,#60H,NOT_EQ
;       ...     ....        ;       R7 = 60H.
NOT_EQ  JC      REQ_LOW     ;       IF R7 < 60H.
;       ...     ....        ;       R7 > 60H.
```

sets the carry flag and branches to the instruction at label NOT_EQ. By testing the carry flag, this instruction determines whether R7 is greater or less than 60H.

If the data being presented to Port 1 is also 34H, then the instruction,

```
WAIT:  CJNE  A,P1,WAIT
```

clears the carry flag and continues with the next instruction in sequence, since the Accumulator does equal the data read from P1. (If some other value was being input on P1, the program will loop at this point until the P1 data changes to 34H.)

**CJNE A,direct,rel**

**Bytes:** 3

**Cycles:** 2

**Encoding:**

| 1 0 1 1 | 0 1 0 1 | direct address | rel. address |
|---|---|---|---|

**Operation:**

```
(PC) ← (PC) + 3
IF (A) < > (direct)
THEN
        (PC) ← (PC) + relative offset
IF (A) < (direct)
THEN.
        (C) ← 1
ELSE
        (C) ← 0
```

**CJNE A,#data,rel**

**Bytes:** 3

**Cycles:** 2

**Encoding:**

| 1 0 1 1 | 0 1 0 0 | immediate data | rel. address |
|---|---|---|---|

**Operation:**

(PC) ← (PC) + 3
IF (A) < > *data*
THEN
(PC) ← (PC) + *relative offset*
IF (A) < *data*
THEN
(C) ← 1
ELSE
(C) ← 0

**CJNE Rn,#data,rel**

**Bytes:** 3

**Cycles:** 2

**Encoding:**

| 1 0 1 1 | 1 r r r | immediate data | rel. address |
|---|---|---|---|

**Operation:**

(PC) ← (PC) + 3
IF ($R_n$) < > *data*
THEN
(PC) ← (PC) + *relative offset*
IF ($R_n$) < *data*
THEN
(C) ← 1
ELSE
(C) ← 0

**CJNE @Ri,#data,rel**

**Bytes:** 3

**Cycles:** 2

**Encoding:**

| 1 0 1 1 | 0 1 1 i | immediate data | rel. address |
|---|---|---|---|

**Operation:**

(PC) ← (PC) + 3
IF (($R_i$)) < > *data*
THEN
(PC) ← (PC) + *relative offset*
IF (($R_i$)) < *data*
THEN
(C) ← 1
ELSE
(C) ← 0

**CLR A**

**Function:** Clear Accumulator

**Description:** The Accumulator is cleared (all bits reset to zero). No flags are affected.

**Example:** The Accumulator contains 5CH (01011100B). The instruction,

CLR A

will leave the Accumulator set to 00H (00000000B).

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 1 1 0 | 0 1 0 0 |
|---|---|

**Operation:** CLR
(A) ← 0

**CLR bit**

**Function:** Clear bit

**Description:** The indicated bit is cleared (reset to zero). No other flags are affected. CLR can operate on the carry flag or any directly addressable bit.

**Example:** Port 1 has previously been written with 5DH (01011101B). The instruction,

CLR P1.2

will leave the port set to 59H (01011001B).

**CLR C**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 1 0 0 | 0 0 1 1 |
|---|---|

**Operation:** CLR
(C) ← 0

**CLR bit**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 1 1 0 0 | 0 0 1 0 | bit address |
|---|---|---|

**Operation:** CLR
(bit) ← 0

### CPL A

**Function:** Complement Accumulator

**Description:** Each bit of the Accumulator is logically complemented (one's complement). Bits which previously contained a one are changed to a zero and vice-versa. No flags are affected.

**Example:** The Accumulator contains 5CH (01011100B). The instruction,

CPL A

will leave the Accumulator set to 0A3H (10100011B).

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 1 1 1 | 0 1 0 0 |
|---|---|

**Operation:** CPL
(A) ← ⌝ (A)

### CPL bit

**Function:** Complement bit

**Description:** The bit variable specified is complemented. A bit which had been a one is changed to zero and vice-versa. No other flags are affected. CLR can operate on the carry or any directly addressable bit.

*Note:* When this instruction is used to modify an output pin, the value used as the original data will be read from the output data latch, *not* the input pin.

**Example:** Port 1 has previously been written with 5DH (01011101B). The instruction sequence,

CPL P1.1

CPL P1.2

will leave the port set to 5BH (01011011B).

### CPL C

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 0 1 1 | 0 0 1 1 |
|---|---|

**Operation:** CPL
(C) ← ⌝ (C)

### CPL bit

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 1 0 1 1 | 0 0 1 0 | bit address |
|---|---|---|

**Operation:** CPL
(bit) ← ⌝ (bit)

**DA A**

**Function:** Decimal-adjust Accumulator for Addition

**Description:** DA A adjusts the eight-bit value in the Accumulator resulting from the earlier addition of two variable (each in packed-BCD format), producing two four-bit digits. Any ADD or ADDC instruction may have been used to perform the addition.

If Accumulator bits 3-0 are greater than nine (xxx1010-xxx1111), or if the AC flag is one, six is added to the Accumulator, producing the proper BCD digit in the low-order nibble. This internal addition would set the carry flag if a carry-out of the low-order four-bit field propagated through all high-order bits, but it would not clear the carry flag otherwise.

If the carry flag is now set, or if the four high-order bits now exceed nine (1010xxx-111xxxx), these high-order bits are incremented by six, producing the proper BCD digit in the high-order nibble. Again, this would set the carry flag if there was a carry-out of the high-order bits, but wouldn't clear the carry. The carry flag thus indicates if the sum of the original two BCD variables is greater than 100, allowing multiple precision decimal addition. OV is not affected.

All of this occurs during the one instruction cycle. Essentially, this instruction performs the decimal conversion by adding 00H, 06H, 60H, or 66H to the Accumulator, depending on initial Accumulator and PSW conditions.

*Note*: DA A *cannot* simply convert a hexadecimal number in the Accumulator to BCD notation, nor does DA A apply to decimal subtraction.

**Example:** The Accumulator holds the value 56H (01010110B) representing the packed BCD digits of the decimal number 56. Register 3 contains the value 67H (01100111B) representing the packed BCD digits of the decimal number 67. The carry flag is set.. The instruction sequence,

```
ADDC    A,R3
DA      A
```

will first perform a standard two's-complement binary addition, resulting in the value 0BEH (10111110B) in the Accumulator. The carry and auxiliary carry flags will be cleared.

The Decimal Adjust instruction will then alter the Accumulator to the value 24H (00100100B), indicating the packed BCD digits of the decimal number 24, the low-order two digits of the decimal sum of 56, 67, and the carry-in. The carry flag will be set by the Decimal Adjust instruction, indicating that a decimal overflow occurred. The true sum 56, 67, and 1 is 124.

BCD variables can be incremented or decremented by adding 01H or 99H. If the Accumulator initially holds 30H (representing the digits of 30 decimal), the the instruction sequence,

```
ADD     A,#99H
DA      A
```

will leave the carry set and 29H in the Accumulator, since 30 + 99 = 129. The low-order byte of the sum can be interpreted to mean 30 – 1 = 29.

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
|---|---|---|---|---|---|---|---|

**Operation:**

```
DA
–contents of Accumulator are BCD
IF   [[(A3-0) > 9] ∨ [(AC) = 1]]
     THEN(A3-0) ← (A3-0) + 6
          AND
IF   [[(A7-4) > 9] ∨ [(C) = 1]]
     THEN(A7-4) ← (A7-4) + 6
```

## DEC byte

**Function:** Decrement

**Description:** The variable indicated is decremented by 1. An original value of 00H will underflow to 0FFH. No flags are affected. Four operand addressing modes are allowed: accumulator, register, direct, or register-indirect.

*Note:* When this instruction is used to modify an output port, the value used as the original data will be read from the output data latch, *not* the input pin.

**Example:** Register 0 contains 7FH (01111111B). Internal RAM locations 7EH and 7FH contain 00H and 40H, respectively. The instruction sequence,

```
DEC   @R0
DEC   R0
DEC   @R0
```

will leave register 0 set to 7EH and internal RAM locations 7EH and 7FH set to 0FFH and 3FH.

### DEC A

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 0 0 1 | 0 1 0 0 |
|---|---|

**Operation:** DEC

$(A) \leftarrow (A) - 1$

### DEC Rn

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 0 0 1 | 1 r r r |
|---|---|

**Operation:** DEC

$(R_n) \leftarrow (R_n) - 1$

### DEC direct

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 0 0 1 | 0 1 0 1 | direct address |
|---|---|---|

**Operation:** DEC

$(direct) \leftarrow (direct) - 1$

### DEC @Ri

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 0 0 1 | 0 1 1 i |
|---|---|

**Operation:** DEC

$((R_i)) \leftarrow ((R_i)) - 1$

**DIV AB**

**Function:** Divide

**Description:** DIV AB divides the unsigned eight-bit integer in the Accumulator by the unsigned eight-bit integer in register B.

The Accumulator receives the integer part of the quotient; register B receives the integer remainder. The carry and OV flags will be cleared.

*Exception:* if B had originally contained 00H, the values returned in the Accumulator and B-register will be undefined and the overflow flag will be set. The carry flag is cleared in any case.

**Example:** The Accumulator contains 251 (0FBH or 11111011B) and B contains 18 (12H or 00010010B). The instruction,

DIV AB

will leave 13 in the Accumulator (0DH or 00001101B) and the value 17 (11H or 00010001B) in B, since 251 = (13 x 18) + 17. Carry and OV will both be cleared.

**Bytes:** 1

**Cycles:** 4

**Encoding:**

| 1 0 0 0 | 0 1 0 0 |
|---|---|

**Operation:** DIV

$(A)_{15-8} \leftarrow (A)/(B)$

$(B)_{7-0}$

**DJNZ <byte>,<rel-addr>**

**Function:** Decrement and Jump if Not Zero

**Description:** DJNZ decrements the location indicated by 1, and branches to the address indicated by the second operand if the resulting value is not zero. An original value of 00H will underflow to 0FFH. No flags are affected. The branch destination would be computed by adding the signed relative-displacement value in the last instruction byte to the PC, after incrementing the PC to the first byte of the following instruction.

The location decremented may be a register or directly addressed byte.

*Note:* When this instruction is used to modify an output port, the value used as the original port data will be read from the output data latch, *not* the input pins.

**Example:** Internal RAM locations 40H, 50H, and 60H contain the values 01H, 70H, and 15H, respectively. The instruction sequence,

```
DJNZ    40H,LABEL_1
DJNZ    50H,LABEL_2
DJNZ    60H,LABEL_3
```

will cause a jump to the instruction at LABEL_2 with the values 00h, 6FH, and 15H in the three RAM locations. The first jump was *not* taken because the result was zero.

This instruction provides a simple was of executing a program loop a given number of times, or for adding a moderate time delay (from 2 to 512 machine cycles) with a single instruction. The instruction sequence,

```
        MOV     R2,#8
TOGGLE: CPL     P1.7
        DJNZ    R2,TOGGLE
```

will toggle P1.7 eight times, causing four output pulses to appear at bit 7 of output Port 1. Each pulse will last three machine cycles, two for DJNZ and one to alter the pin.

**DJNZ Rn,rel**

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| 1 1 0 1 | 1 r r r | rel. address |
|---|---|---|

**Operation:**

DJNZ
(PC) ← (PC) + 2
($R_n$) ← ($R_n$) – 1
IF ($R_n$) > 0 or ($R_n$) < 0
THEN
(PC) ← (PC) + rel

**DJNZ direct,rel**

**Bytes:** 3

**Cycles:** 2

**Encoding:**

| 1 1 0 1 | 0 1 0 1 | direct data | rel. address |
|---|---|---|---|

**Operation:**

DJNZ
(PC) ← (PC) + 2
(direct) ← (direct) – 1
IF (direct) > 0 or (direct) < 0
THEN
(PC) ← (PC) + rel

## INC <byte>

**Function:** Increment

**Description:** INC increments the indicated variable by 1. An original value of 0FFH will overflow to 00H. No flags are affected. Three addressing modes are allowed: register, direct, or register-indirect.

*Note:* When this instruction is used to modify an output port, the value used as the original port data will be read from the output data latch, *not* the input pins.

**Example:** Register 0 contains 7EH (01111110B). Internal RAM locations 7EH and 7FH contain 0FFH and 40H, respectively. The instruction sequence,

```
INC     @R0
INC     R0
INC     @R0
```

will leave register 0 set to 7FH and internal RAM locations 7EH and 7FH holding (respectively) 00H and 41H.

### INC A

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 0 0 0 | 0 1 0 0 |
|---|---|

**Operation:** INC

$(A) \leftarrow (A) + 1$

### INC Rn

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 0 0 0 | 1 r r r |
|---|---|

**Operation:** INC

$(R_n) \leftarrow (R_n) + 1$

### INC direct

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 0 0 0 | 0 1 0 1 | direct address |
|---|---|---|

**Operation:** INC

(direct) ← (direct) + 1

### INC @Ri

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 0 0 0 | 0 1 1 i |
|---|---|

**Operation:** INC

$((R_i)) \leftarrow ((R_i)) + 1$

### INC DPTR

**Function:** Increment Data Pointer

**Description:** Increment the 16-bit data pointer by 1. A 16-bit increment (modulo $2^{16}$) is performed; an overflow of the low-order byte of the data pointer (DPL) from 0FFH to 00H will increment the high-order byte (DPH). No flags are affected.

This is the only 16-bit register which can be incremented.

**Example:** Registers DPH and DPL contain 12H and 0FEH, respectively. The instruction sequence,

```
INC     DPTR
INC     DPTR
INC     DPTR
```

will change DPH and DPL to 13H and 01H.

**Bytes:** 1

**Cycles:** 2

**Encoding:**

| 1 0 1 0 | 0 0 1 1 |
|---|---|

**Operation:**

```
INC
(DPTR) ← (DPTR) + 1
```

### JB bit,rel

**Function:** Jump if Bit set

**Description:** If the indicated bit is a one, jump to the address indicated; otherwise proceed with the next instruction. The branch destination is computed by adding the signed relative-displacement in the third instruction byte to the PC, after incrementing the PC to the first byte of the next instruction. *The bit tested is not modified.* No flags are affected.

**Example:** The data present at input port 1 is 11001010B. The Accumulator holds 56 (01010110B). The instruction sequence,

```
JB      P1.2,LABEL1
JB      ACC.2,LABEL2
```

will cause program execution to branch to the instruction at label LABEL2.

**Bytes:** 3

**Cycles:** 2

**Encoding:**

| 0 0 1 0 | 0 0 0 0 | bit address | rel. address |
|---|---|---|---|

**Operation:**

```
JB
(PC) ← (PC) + 3
IF (bit) = 1
   THEN
        (PC) ← (PC) + rel
```

**JBC bit,rel**

**Function:** Jump if Bit is set and Clear bit

**Description:** If the indicated bit is a one, branch to the address indicated; otherwise proceed with the next instruction. *The bit will not be cleared if it is already a zero.* The branch destination is computed by adding the signed relative-displacement in the third instruction byte to the PC, after incrementing the PC to the first byte of the next instruction. No flags are affected.

Note: When this instruction is used to test an output pin, the value used as the original data will read from the output data latch, not the input pin.

**Example:** The Accumulator holds 56H (01010110B). The instruction sequence,

```
JBC     ACC.3,LABEL1
JBC     ACC.2,LABEL2
```

will cause program execution to continue at the instruction identified by the LABEL2, with the Accumulator modified to 52H (01010010B).

**Bytes:** 3

**Cycles:** 2

**Encoding:**

| 0 0 0 1 | 0 0 0 0 | bit address | rel. address |
|---|---|---|---|

**Operation:**

```
JBC
(PC) ← (PC) + 3
IF (bit) = 1
   THEN
(bit) ← 0
(PC) ← (PC) + rel
```

**JC rel**

**Function:** Jump if Carry is set

**Description:** If the carry flag is set, branch to the address indicated; otherwise proceed with the next instruction. The branch destination is computed by adding the signed relative-displacement in the second instruction byte to the PC, after incrementing the PC twice. No flags are affected.

**Example:** The carry flag is cleared. The instruction sequence,

```
JC      LABEL1
CPL     C
JC      LABEL2
```

will set the carry and cause program execution to continue at the instruction identified by the label LABEL2.

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| 0 1 0 0 | 0 0 0 0 | rel. address |
|---|---|---|

**Operation:**

```
JC
(PC) ← (PC) + 2
IF (C) = 1
   THEN
            (PC) ← (PC) + rel
```

### JMP @A+DPTR

**Function:** Jump indirect

**Description:** Add the eight-bit unsigned contents of the Accumulator with the sixteen-bit data pointer, and load the resulting sum to the program counter. This will be the address for subsequent instruction fetches. Sixteen-bit addition is performed (modulo $2^{16}$): a carry-out from the low-order eight bits propagates through the higher-order bits. Neither the Accumulator nor the Data Pointer is altered. No flags are affected.

**Example:** An even number from 0 to 6 is in the Accumulator. The following sequence of instructions will branch to one of four AJMP instructions in a jump table starting at JMP_TBL:

```
            MOV   DPTR,#JMP_TBL
            JMP   @A+DPTR
JMP_TBL:    AJMP  LABEL0
            AJMP  LABEL1
            AJMP  LABEL2
            AJMP  LABEL3
```

If the Accumulator equals 04H when starting this sequence, execution will jump to label LABEL2. Remember that AJMP is a two-byte instruction, so the jump instructions start at every other address.

**Bytes:** 1

**Cycles:** 2

**Encoding:**

| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|

**Operation:** JMP
(PC) ← (A) + (DPTR)

### JNB bit,rel

**Function:** Jump if Bit Not set

**Description:** If the indicated bit is a zero, branch to the indicated address; otherwise proceed with the next instruction. The branch destination is computed by adding the signed relative-displacement in the third instruction byte to the PC, after incrementing the PC to the first byte of the next instruction. *The bit tested is not modified.* No flags are affected.

**Example:** The data present at input port 1 is 11001010B. The Accumulator holds 56H (01010110B). The instruction sequence,

```
JNB     P1.3,LABEL1
JNB     ACC.3,LABEL2
```

will cause program execution to continue at the instruction at label LABEL2.

**Bytes:** 3

**Cycles:** 2

**Encoding:**

| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | bit address | rel. address |
|---|---|---|---|---|---|---|---|---|---|

**Operation:** JNB
(PC) ← (PC) + 3
IF (bit) = 0
THEN
(PC) ← (PC) + rel

**JNC rel**

**Function:** Jump if Carry Not set

**Description:** If the carry flag is a zero, branch to the address indicated; otherwise proceed with the next instruction. The branch destination is computed by adding the signed relative-displacement in the second instruction byte to the PC, after incrementing the PC twice to point to the next instruction. The carry flag is not modified.

**Example:** The carry flag is set. The instruction sequence,

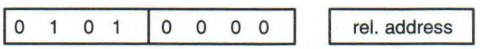
```
JNC     LABEL1
CPL     C
JNC     LABEL2
```

will clear the carry and cause program execution to continue at the instruction identified by the label LABEL2.

**Bytes:** 2

**Cycles:** 2

**Encoding:**




**Operation:**

```
JNC
(PC) ← (PC) + 2
IF (C) = 0
   THEN
(PC) ← (PC) + rel
```

**JNZ rel**

**Function:** Jump if Accumulator Not Zero

**Description:** If any bit of the Accumulator is a one, branch to the indicated address; otherwise proceed with the next instruction. The branch destination is computed by adding the signed relative-displacement in the second instruction byte to the PC, after incrementing the PC twice. The Accumulator is not modified. No flags are affected.

**Example:** The Accumulator originally holds 00H. The instruction sequence,

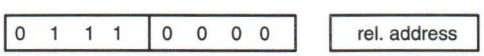
```
JNZ     LABEL1
INC     A
JNZ     LABEL2
```

will set the Accumulator to 01H and continue at label LABEL2.

**Bytes:** 2

**Cycles:** 2

**Encoding:**




**Operation:**

```
JNZ
(PC) ← (PC) + 2
IF A ≠ 0
   THEN  (PC) ← (PC) + rel
```

**JZ rel**

**Function:** Jump if Accumulator Zero

**Description:** If all bits of the Accumulator are zero, branch to the indicated address; otherwise proceed with the next instruction. The branch destination is computed by adding the signed relative-displacement in the second instruction byte to the PC, after incrementing the PC twice. The Accumulator is not modified. No flags are affected.

**Example:** The Accumulator originally holds 01H. The instruction sequence,

```
JZ      LABEL1
DEC     A
JZ      LABEL2
```

will change the Accumulator to 00H and cause program execution to continue at the instruction identified by the label LABEL2.

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| 0 1 1 0 | 0 0 0 0 | rel. address |
|---|---|---|

**Operation:**

```
JZ
(PC) ← (PC) + 2
IF A = 0
   THEN (PC) ← (PC) + rel
```

**LCALL addr16**

**Function:** Long Call

**Description:** LCALL calls a subroutine located at the indicated address. The instruction adds three to the program counter to generate the address of the next instruction and then pushes the 16-bit result onto the stack (low byte first), incrementing the Stack Pointer by two. The high-order and low-order bytes of the PC are then loaded, respectively, with the second and third bytes of the LCALL instruction. Program execution continues with the instruction at this address. The subroutine may therefore begin anywhere in the full 64k-byte program memory address space. No flags are affected.

**Example:** Initially the Stack Pointer equals 07H. The label "SUBRTN" is assigned to program memory location 1234H. After executing the instruction,

```
LCALL   SUBRTN
```

at location 0123H, the Stack Pointer will contain 09H, internal RAM locations 08H and 09H will contain 26H and 01H, and the PC will contain 1235H.

**Bytes:** 3

**Cycles:** 2

**Encoding:**

| 0 0 0 1 | 0 0 1 0 | addr15-addr8 | addr7-addr0 |
|---|---|---|---|

**Operation:**

```
LCALL
(PC) ← (PC) + 3
(SP) ← (SP) + 1
((SP)) ← (PC7-0)
(SP) ← (SP) + 1
((SP)) ← (PC15-8)
(PC) ← addr15-0
```

**LJMP addr16 (Implemented in 87C751 and 87C752 for in-circuit emulation only.)**

**Function:** Long Jump

**Description:** LJMP causes an unconditional branch to the indicated address, by loading the high-order and low-order bytes of the PC (respectively) with the second and third instruction bytes. The destination may therefore be anywhere in the full 64k program memory address space. No flags are affected.

**Example:** The label "JMPADR" is assigned to the instruction at program memory location 1234H. The instruction,

LJMP JMPADR

at location 0123H will load the program counter with 1234H.

**Bytes:** 3

**Cycles:** 2

**Encoding:**

| 0 0 0 0 | 0 0 1 0 | addr15-addr8 | addr7-addr0 |
|---|---|---|---|

**Operation:** LJMP

(PC) ← $addr_{15-0}$

**MOV <dest-byte>,<src-byte>**

**Function:** Move byte variable

**Description:** The byte variable indicated by the second operand is copied into the location specified by the first operand. The source byte is not affected. No other register or flag is affected.

This is by far the most flexible operation. Fifteen combinations of source and destination addressing modes are allowed.

**Example:** Internal RAM location 30H holds 40H. The value of RAM location 40H is 10H. The data present at input port 1 is 11001010B (0CAH). The instruction sequence,

```
MOV     R0,#30H     ;R0 < = 30H
MOV     A,@R0       ;A < = 40H
MOV     R1,A        ;R1 < = 40H
MOV     B,@R1       ;B < = 10H
MOV     @R1,P1      ;RAM (40H) < = 0CAH
MOV     P2,P1       ;P2 #0CAH
```

leaves the value 30H in register 0, 40H in both the Accumulator and register 1, 10H in register B, and 0CAH (11001010B) both in RAM location 40H and output on port 2.

**MOV A,Rn**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 1 1 0 | 1 r r r |
|---|---|

**Operation:** MOV

(A) ← ($R_n$)

***MOV A,direct**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 1 1 1 0 | 0 1 0 1 | direct address |
|---|---|---|

**Operation:** MOV
(A) ← (direct)

**MOV A,@Ri**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 1 1 0 | 0 1 1 i |
|---|---|

**Operation:** MOV
(A) ← ((Ri))

**MOV A,#data**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 1 1 1 | 0 1 0 0 | immediate data |
|---|---|---|

**Operation:** MOV
(A) ← #data

**MOV Rn,A**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 1 1 1 | 1 r r r |
|---|---|

**Operation:** MOV
(Rn) ← (A)

**MOV Rn,direct**

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| 1 0 1 0 | 1 r r r | direct address |
|---|---|---|

**Operation:** MOV
(Rn) ← (direct)

**MOV Rn,#data**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 1 1 1 | 1 r r r | immediate data |
|---|---|---|

**Operation:** MOV
(Rn) ← #data

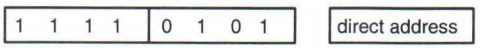
*MOV A,ACC is not a valid instruction.

**MOV direct,A**

**Bytes:** 2

**Cycles:** 1

**Encoding:**




**Operation:** MOV
(direct) ← (A)

**MOV direct,Rn**

**Bytes:** 2

**Cycles:** 2

**Encoding:**




**Operation:** MOV
(direct) ← ($R_n$)

**MOV direct,direct**

**Bytes:** 3

**Cycles:** 2

**Encoding:**



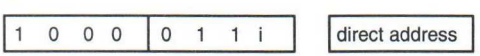

**Operation:** MOV
(direct) ← (direct)

**MOV direct,@Ri**

**Bytes:** 2

**Cycles:** 2

**Encoding:**




**Operation:** MOV
(direct) ← (($R_i$))

**MOV direct,#data**

**Bytes:** 3

**Cycles:** 2

**Encoding:**



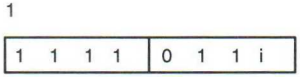

**Operation:** MOV
(direct) ← #data

**MOV @Ri,A**

**Bytes:** 1

**Cycles:** 1

**Encoding:**




**Operation:** MOV
(($R_i$)) ← (A)

**MOV @Ri,direct**

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| 1 0 1 0 | 0 1 1 i | direct address |
|---|---|---|

**Operation:** MOV
$((R_i)) \leftarrow$ (direct)

**MOV @Ri,#data**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 1 1 1 | 0 1 1 i | immediate data |
|---|---|---|

**Operation:** MOV
$((R_i)) \leftarrow$ #data

**MOV <dest-bit>,<src-bit>**

**Function:** Move bit data

**Description:** The Boolean variable indicated by the second operand is copied into the location specified by the first operand. One of the operands must be the carry flag; the other may be any directly addressable bit. No other register or flag is affected.

**Example:** The carry flag is originally set. The data present at input Port 3 is 11000101B. The data previously written to output Port 1 is 35H (00110101B). The instruction sequence,

```
MOV   P1.3,C
MOV   C,P3.3
MOV   P1.2,C
```

will leave the carry cleared and change Port 1 to 39H (00111001B).

**MOV C,bit**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 1 0 1 0 | 0 0 1 0 | bit address |
|---|---|---|

**Operation:** MOV
(C) $\leftarrow$ (bit)

**MOV bit,C**

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| 1 0 0 1 | 0 0 1 0 | bit address |
|---|---|---|

**Operation:** MOV
(bit) $\leftarrow$ (C)

### MOV DPTR,#data16

**Function:** Load Data Pointer with a 16-bit constant

**Description:** The Data Pointer is loaded with the 16-bit constant indicated. The 16-bit constant is loaded into the second and third bytes of the instruction. The second byte (DPH) is the high-order byte, while the third byte (DPL) holds the low-order byte. No flags are affected.

This is the only instruction which moves 16 bits of data at once.

**Example:** The instruction,

MOV DPTR,#1234H

will load the value 1234H into the Data Pointer: DPH will hold 12H and DPL will hold 34H.

**Bytes:** 3

**Cycles:** 2

**Encoding:**

| 1 0 0 1 | 0 0 0 0 | immed. data15-8 | immed. data7-0 |
|---|---|---|---|

**Operation:** MOV

(DPTR) ← (#data$_{15-0}$)

DPH ☐ DPL ← #data$_{15-8}$☐ #data$_{7-0}$

### MOVC A,@A+<base-reg>

**Function:** Move Code byte

**Description:** The MOVC instructions load the Accumulator with a code byte, or constant from program memory. The address of the byte fetched is the sum of the original unsigned eight-bit Accumulator contents and the contents of a sixteen-bit base register, which may be either the Data Pointer or the PC. In the latter case, the PC is incremented to the address of the following instruction before being added with the Accumulator; otherwise the base register is not altered. Sixteen-bit addition is performed so a carry-out from the low-order eight bits may propagate through higher-order bits. No flags are affected.

**Example:** A value between 0 and 3 is in the Accumulator. The following instructions will translate the value in the Accumulator to one of four values defined by the DB (define byte) directive:

```
REL_PC:   INC    A
          MOVC   A,@A+PC
          RET
          DB     66H
          DB     77H
          DB     88H
          DB     99H
```

If the subroutine is called with the Accumulator equal to 01H, it will return with 77H in the Accumulator. The INC A before the MOVC instruction is needed to "get around" the RET instruction above the table. If several bytes of code separated the MOVC from the table, the corresponding number would be added to the Accumulator instead.

### MOVC A,@A+DPTR

**Bytes:** 1

**Cycles:** 2

**Encoding:**

| 1 0 0 1 | 0 0 1 1 |
|---|---|

**Operation:** MOVC

(A) ← ((A) + (DPTR))

**MOVC A,@A+PC**

**Bytes:** 1

**Cycles:** 2

**Encoding:**

| 1 0 0 0 | 0 0 1 1 |
|---|---|

**Operation:** MOVC
(PC) ← (PC) + 1
(A) ← ((A) + (PC))

**MOVX <dest-byte>,<src-byte> (Not implemented in the 8XC752 or 8XC752)**

**Function:** Move External

**Description:** The MOVX instructions transfer data between the Accumulator and a byte of external data memory, hence the "X" appended to MOV. There are two types of instructions, differing in whether they provide an eight-bit or sixteen-bit indirect address to the external data RAM.

In the first type, the contents of R0 or R1 in the current register bank provide an eight-bit address multiplexed with data on P0. Eight bits are sufficient for external I/O expansion decoding or for a relatively small RAM array. For somewhat larger arrays, port pins can be used to output higher-order address bits. These pins would be controlled by an output instruction preceding the MOVX.

In the second type of MOVX instruction, The Data Pointer generates a sixteen-bit address. P2 outputs the high-order eight address bits (the contents of DPH) while P0 multiplexes the low-order eight bits (DPL) with data. The P2 Special Function Register retains its previous contents while the P2 output buffers are emitting the contents of DPH. This form is faster and more efficient when accessing very large data arrays (up to 64k bytes), since no additional instructions are needed to set up the output ports.

It is possible in some situations to mix the two MOVX types. A large RAM array with its high-order address lines driven by P2 can be addressed via the Data Pointer, or with code to output high-order address bits to P2 followed by a MOVX instruction using R0 or R1.

**Example:** An external 256 byte RAM using multiplexed address/data lines is connected to the 8051 Port 0. Port 3 provides control lines for the external RAM. Ports 1 and 2 are used for normal I/O. Registers 0 and 1 contain 12H and 34H. Location 34H of the external RAM holds the value 56H. The instruction sequence,

```
MOVX  A,@R1
MOVX  @R0,A
```

copies the value 56H into both the Accumulator and external RAM location 12H.

**MOVX A,@Ri**

**Bytes:** 1

**Cycles:** 2

**Encoding:**

| 1 1 1 0 | 0 0 1 i |
|---|---|

**Operation:** MOVX
(A) ← ($((R_i))$)

**MOVX A,@DPTR**

**Bytes:** 1

**Cycles:** 2

**Encoding:**

| 1 1 1 0 | 0 0 0 0 |
|---|---|

**Operation:** MOVX
(A) ← ((DPTR))

**MOVX @Ri,A**

**Bytes:** 1

**Cycles:** 2

**Encoding:**

| 1 1 1 1 | 0 0 1 i |
|---|---|

**Operation:** MOVX
$((R_i)) \leftarrow (A)$

**MOVX @DPTR,A**

**Bytes:** 1

**Cycles:** 2

**Encoding:**

| 1 1 1 1 | 0 0 0 0 |
|---|---|

**Operation:** MOVX
((DPTR)) ← (A)

**MUL AB**

**Function:** Multiply

**Description:** MUL AB multiplies the unsigned eight-bit integers in the Accumulator and register B. The low-order byte of the sixteen-bit product is left in the Accumulator, and the high-order byte in B. If the product is greater than 255 (0FFH) the overflow flag is set; otherwise it is cleared. The carry flag is always cleared.

**Example:** Originally the Accumulator holds the value 80 (50H). Register B holds the value 160 (0A0H).The instruction,

MUL AB

will give the product 12,800 (3200H), so B is changed to 32H (00110010B) and the Accumulator is cleared. The overflow flag is set, carry is cleared.

**Bytes:** 1

**Cycles:** 4

**Encoding:**

| 1 0 1 0 | 0 1 0 0 |
|---|---|

**Operation:** MUL
$(A)_{7\text{-}0} \leftarrow (A) \times (B)$
$(B)_{15\text{-}8}$

## NOP

**Function:** No Operation

**Description:** Execution continues at the following instruction. Other than the PC, no registers or flags are affected.

**Example:** It is desired to produce a low-going output pulse on bit 7 of Port 2 lasting exactly 5 cycles. A simple SETB/CLR sequence would generate a one-cycle pulse, so four additional cycles must be inserted. This may be done (assuming  are enabled) with the instruction sequence,

```
CLR     P2.7
NOP
NOP
NOP
NOP
SETB    P2.7
```

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Operation:** NOP

(PC) ← (PC) + 1

## ORL <dest-byte>,<src-byte>

**Function:** Logical-OR for byte variables

**Description:** ORL performs the bitwise logical-OR operation between the indicated variables, storing the results in the destination byte. No flags are affected.

The two operands allow six addressing mode combinations. When the destination is the Accumulator, the source can use register, direct, register-indirect, or immediate addressing; when the destination is a direct address, the source can be the Accumulator or immediate data.

*Note:* When this instruction is used to modify an output port, the value used as the original port data will be read from the output data latch, *not* the input pins.

**Example:** If the Accumulator holds 0C3H (11000011B) and R0 holds 55H (01010101B) then the instruction,

```
ORL   A,R0
```

will leave the Accumulator holding the value 0D7H (11010111B).

When the destination is a directly addressed byte, the instruction can set combinations of bits in any RAM location or hardware register. The pattern of bits to be set is determined by a mask byte, which may be either a constant data value in the instruction or a variable computed in the Accumulator at run-time. The instruction,

```
ORL   P1,#00110010B
```

will set bits 5, 4, and 1 of output Port 1.

## ORL A,Rn

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 1 | r | r | r |

**Operation:** ORL

(A) ← (A) ∨ ($R_n$)

**ORL A,direct**

**Bytes:** 2

**Cycles:** 1

**Encoding:**



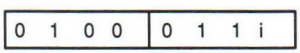

**Operation:** ORL
(A) ← (A) ∨ (direct)

**ORL A,@Ri**

**Bytes:** 1

**Cycles:** 1

**Encoding:**



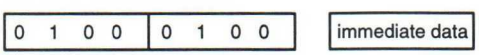

**Operation:** ORL
(A) ← (A) ∨ ((Ri))

**ORL A,#data**

**Bytes:** 2

**Cycles:** 1

**Encoding:**



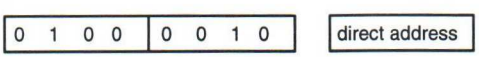

**Operation:** ORL
(A) ← (A) ∨ #data

**ORL direct,A**

**Bytes:** 2

**Cycles:** 1

**Encoding:**



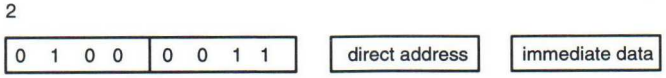

**Operation:** ORL
(direct) ← (direct) ∨ (A)

**ORL direct,#data**

**Bytes:** 3

**Cycles:** 2

**Encoding:**




**Operation:** ORL
(direct) ← (direct) ∨ #data

**ORL C,<src-bit>**

**Function:** Logical-OR for bit variables

**Description:** Set the carry flag if the Boolean value is a logical 1; leave the carry in its current state otherwise. A slash ("/") preceding the operand in the assembly language indicates that the logical complement of the addressed bit is used as the source value, but the source bit itself is not affected. No other flags are affected.

**Example:** Set the carry flag if and only if P1.0 = 1, ACC.7 = 1, or OV = 0:

```
ORL   C,P1.0     ;LOAD CARRY WITH INPUT PIN P10
ORL   C,ACC.7    ;OR CARRY WITH THE ACC. BIT 7
ORL   C,/OV      ;OR CARRY WITH THE INVERSE OF OV.
```

**ORL C,bit**

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| 0 1 1 1 | 0 0 1 0 | bit address |
|---|---|---|

**Operation:** ORL

$(C) \leftarrow (C) \vee (bit)$

**ORL C,/bit**

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| 1 0 1 0 | 0 0 0 0 | bit address |
|---|---|---|

**Operation:** ORL

$(C) \leftarrow (C) \vee (\overline{bit})$

**POP direct**

**Function:** Pop from stack

**Description:** The contents of the internal RAM location addressed by the Stack Pointer is read, and the Stack Pointer is decremented by one. The value read is then transferred to the directly addressed byte indicated. No flags are affected.

**Example:** The Stack Pointer originally contains the value 32H, and internal RAM locations 30H through 32H contain the values 20H, 23H, and 01H, respectively. The instruction sequence,

```
POP   DPH
POP   DPL
```

will leave the Stack Pointer equal to the value 30H and the Data Pointer set to 0123H. At this point the instruction,

```
POP   SP
```

will leave the Stack Pointer set to 20H. Note that in this special case the Stack Pointer was decremented to 2FH before being loaded with the value popped (20H).

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| 1 1 0 1 | 0 0 0 0 | direct address |
|---|---|---|

**Operation:**

```
POP
(direct) ← ((SP))
(SP) ← (SP) – 1
```

**PUSH direct**

**Function:** Push onto stack

**Description:** The Stack Pointer is incremented by one. The contents of the indicated variable is then copied into the internal RAM location addressed by the Stack Pointer. Otherwise no flags are affected.

**Example:** On entering an interrupt routine the Stack Pointer contains 09H. The Data Pointer holds the value 0123H. The instruction sequence,

```
PUSH  DPL
PUSH  DPH
```

will leave the Stack Pointer set to 0BH and store 23H and 01H in internal RAM locations 0AH and 0BH, respectively.

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| 1 1 0 0 | 0 0 0 0 | direct address |
|---|---|---|

**Operation:**

```
PUSH
(SP) ← (SP) + 1
((SP)) ← (direct)
```

**RET**

**Function:** Return from subroutine

**Description:** RET pops the high- and low-order bytes of the PC successively from the stack, decrementing the Stack Pointer by two. Program execution continues at the resulting address, generally the instruction immediately following an ACALL or LCALL. No flags are affected.

**Example:** The Stack Pointer originally contains the value 0BH. Internal RAM locations 0AH and 0BH contain the values 23H and 01H, respectively. The instruction,

RET

will leave the Stack Pointer equal to the value 09H. Program execution will continue at location 0123H.

**Bytes:** 1

**Cycles:** 2

**Encoding:**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |

**Operation:** RET
$(PC_{15-8}) \leftarrow ((SP))$
$(SP) \leftarrow (SP) - 1$
$(PC_{7-0}) \leftarrow ((SP))$
$(SP) \leftarrow (SP) - 1$

**RETI**

**Function:** Return from interrupt

**Description:** RETI pops the high- and low-order bytes of the PC successively from the stack, and restores the interrupt logic to accept additional interrupts at the same priority level as the one just processed. The Stack Pointer is left decremented by two. No other registers are affected; the PSW is not automatically restored to its pre-interrupt status. Program execution continues at the resulting address, which is generally the instruction immediately after the point at which the interrupt request was detected. If a lower- or same-level interrupt has been pending when the RETI instruction is executed, that one instruction will be executed before the pending interrupt is processed.

**Example:** The Stack Pointer originally contains the value 0BH. An interrupt was detected during the instruction ending at location 0122H. Internal RAM locations 0AH and 0BH contain the values 23H and 01H, respectively. The instruction,

RETI

will leave the Stack Pointer equal to 09H and return program execution to location 0123H.

**Bytes:** 1

**Cycles:** 2

**Encoding:**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |

**Operation:** RETI
$(PC_{15-8}) \leftarrow ((SP))$
$(SP) \leftarrow (SP) - 1$
$(PC_{7-0}) \leftarrow ((SP))$
$(SP) \leftarrow (SP) - 1$

**RL A**

**Function:** Rotate Accumulator Left

**Description:** The eight bits in the Accumulator are rotated one bit to the left. Bit 7 is rotated into the bit 0 position. No flags are affected.

**Example:** The Accumulator holds the value 0C5H (11000101B). The instruction,

RL A

leaves the Accumulator holding the value 8BH (10001011B) with the carry unaffected.

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |

**Operation:** RL

$(A_{n+1}) \leftarrow (A_n)$, n = 0 – 6

(A0) ← (A7)

**RLC A**

**Function:** Rotate Accumulator Left through the Carry flag

**Description:** The eight bits in the Accumulator and the carry flag are together rotated one bit to the left. Bit 7 moves into the carry flag; the original state of the carry flag moves into the bit 0 position. No other flags are affected.

**Example:** The Accumulator holds the value 0C5H (11000101B), and the carry is zero. The instruction,

RLC A

leaves the Accumulator holding the value 8AH (10001010B) with the carry set.

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |

**Operation:** RLC

$(A_{n+1}) \leftarrow (A_n)$, n = 0 – 6

(A0) ← (C)

(C) ← (A7)

**RR A**

**Function:** Rotate Accumulator Right

**Description:** The eight bits in the Accumulator are rotated one bit to the right. Bit 0 is rotated into the bit 7 position. No flags are affected.

**Example:** The Accumulator holds the value 0C5H (11000101B). The instruction,

RR A

leaves the Accumulator holding the value 0E2H (11100010B) with the carry unaffected.

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

**Operation:** RR

$(A_n) \leftarrow (A_{n+1})$, n = 0 – 6

(A7) ← (A0)

**RRC A**

**Function:** Rotate Accumulator Right through the Carry flag

**Description:** The eight bits in the Accumulator and the carry flag are together rotated one bit to the right. Bit 0 moves into the carry flag; the original state of the carry flag moves into the bit 7 position. No other flags are affected.

**Example:** The Accumulator holds the value 0C5H (11000101B), and the carry is zero. The instruction,

RRC A

leaves the Accumulator holding the value 62 (01100010B) with the carry set.

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |

**Operation:** RRC

$(A_n) \leftarrow (A_{n+1})$, n = 0 – 6

(A7) ← (C)

(C) ← (A0)

**SETB <bit>**

**Function:** Set Bit

**Description:** SETB sets the indicated bit to one. SETB can operate on the carry flag or any directly addressable bit. No other flags are affected.

**Example:** The carry flag is cleared. Output Port 1 has been written with the value 34H (00110100B). The instructions,

```
SETB   C
SETB   P1.0
```

will leave the carry flag set to 1 and change the data output on Port 1 to 35H (00110101B).

**SETB C**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 1 0 1 | 0 0 1 1 |
|---|---|

**Operation:**
SETB
(C) ← 1

**SETB bit**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 1 1 0 1 | 0 0 1 0 | bit address |
|---|---|---|

**Operation:**
SETB
(bit) ← 1

**SJMP rel**

**Function:** Short Jump

**Description:** Program control branches unconditionally to the address indicated. The branch destination is computed by adding the signed displacement in the second instruction byte to the PC, after incrementing the PC twice. Therefore, the range of destinations allowed is from 128 bytes preceding this instruction to 127 bytes following it.

**Example:** The label "RELADR" is assigned to an instruction at program memory location 0123H. The instruction,

```
SJMP   RELADR
```

will assemble into location 0100H. After the instruction is executed, the PC will contain the value 0123H.

(*Note:* Under the above conditions the instruction following SJMP will be at 102H. Therefore, the displacement byte of the instruction will be the relative offset (0123H-0102H) = 21H. Put another way, an SJMP with a displacement of 0FEH would be a one-instruction infinite loop.)

**Bytes:** 2

**Cycles:** 2

**Encoding:**

| 1 0 0 0 | 0 0 0 0 | rel. address |
|---|---|---|

**Operation:**
SJMP
(PC) ← (PC) + 2
(PC) ← (PC) + rel

**SUBB A, <src-byte>**

**Function:** Subtract with borrow

**Description:** SUBB subtracts the indicated variable and the carry flag together from the Accumulator, leaving the result in the Accumulator. SUBB sets the carry (borrow) flag if a borrow is needed for bit 7, and clears C otherwise. (If C was set *before* executing a SUBB instruction, this indicates that a borrow was needed for the previous step in a multiple precision subtraction, so the carry is subtracted from the Accumulator along with the source operand.) AC is set if a borrow is needed for bit 3, and cleared otherwise. OV is set if a borrow is needed into bit 6, but not into bit 7, or into bit 7, but not bit 6.

When subtracting signed integers OV indicates a negative number produced when a negative value is subtracted from a positive value, or a positive result when a positive number is subtracted from a negative number.

The source operand allows four addressing modes: register, direct, register-indirect, or immediate.

**Example:** The Accumulator holds 0C9H (11001001B), register 2 holds 54H (01010100B), and the carry flag is set. The instruction,

SUBB A,R2

will leave the value 74H (01110100B) in the Accumulator, with the carry flag and AC cleared but OV set.

Notice that 0C9H minus 54H is 75H The difference between this and the above result is due to the carry (borrow) flag being set before the operation. If the state of the carry is not known before starting a single or multiple-precision subtraction, it should be explicitly cleared by a CLR C instruction

**SUBB A,Rn**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 | 0 | 0 | 1 | 1 | r | r | r |
|---|---|---|---|---|---|---|---|

**Operation:** SUBB
(A) ← (A) – (C) – ($R_n$)

**SUBB A,direct**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | direct address |
|---|---|---|---|---|---|---|---|---|

**Operation:** SUBB
(A) ← (A) – (C) – (direct)

**SUBB A,@Ri**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 | 0 | 0 | 1 | 0 | 1 | 1 | i |
|---|---|---|---|---|---|---|---|

**Operation:** SUBB
(A) ← (A) – (C) – ($R_i$)

**SUBB A,#data**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | immediate data |
|---|---|---|---|---|---|---|---|---|

**Operation:** SUBB
(A) ← (A) – (C) – (#data)

**SWAP A**

**Function:** Swap nibbles within the Accumulator

**Description:** SWAP A interchanges the low- and high-order nibbles (four-bit fields) of the Accumulator (bits 3-0 and bits 7-4). The operation can also be thought of as a four-bit rotate instruction. No flags are affected.

**Example:** The Accumulator holds the value 0C5H (11000101B). The instruction,

SWAP A

leaves the Accumulator holding the value 5CH (01011100B).

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 1 0 0 | 0 1 0 0 |
|---|---|

**Operation:** SWAP
$(A_{3-0}) \rightleftarrows (A_{7-4})$

**XCH A,<byte>**

**Function:** Exchange Accumulator with byte variable

**Description:** XCH loads the Accumulator with the contents of the indicated variable, at the same time writing the original Accumulator contents to the indicated variable. The source/destination operand can use register, direct, or register-indirect addressing.

**Example:** R0 contains the address 20H. The Accumulator holds the value 3FH (00111111B). Internal RAM location 20H holds the value 75H (01110101B). The instruction,

XCH A,@R0

will leave the RAM location 20H holding the values 3FH (00111111B) and 75H (01110101B) in the Accumulator.

**XCH A,Rn**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 1 0 0 | 1 r r r |
|---|---|

**Operation:** XCH
$(A) \rightleftarrows (R_n)$

**XCH A,direct**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 1 1 0 0 | 0 1 0 1 | direct address |
|---|---|---|

**Operation:** XCH
(A) ⇄ (direct)

**XCH A,@Ri**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 1 0 0 | 0 1 1 i |
|---|---|

**Operation:** XCH
$(A) \rightleftarrows ((R_i))$

### XCHD A,@Ri

**Function:** Exchange Digit

**Description:** XCHD exchanges the low-order nibble of the Accumulator (bits 3-0), generally representing a hexadecimal or BCD digit, with that of the internal RAM location indirectly addressed by the specified register. The high-order nibbles (bits 7-4) of each register are not affected. No flags are affected.

**Example:** R0 contains the address 20H. The Accumulator holds the value 36H (00110110B). Internal RAM location 20H holds the value 75H (01110101B). The instruction,

XCHD A,@R0

will leave RAM location 20H holding the value 76H (01110110B) and 35H (00110101B) in the Accumulator.

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 1 1 0 1 | 0 1 1 i |
|---|---|

**Operation:** XCHD

$(A_{3\text{-}0}) \rightleftarrows ((Ri_{3\text{-}0}))$

### XRL <dest-byte>,<src-byte>

**Function:** Logical Exclusive-OR for byte variables

**Description:** XRL performs the bitwise logical Exclusive-OR operation between the indicated variables, storing the results in the destination. No flags are affected.

The two operands allow six addressing mode combinations. When the destination is the Accumulator, the source can use register, direct, register-indirect, or immediate addressing; when the destination is a direct address, the source can be the Accumulator or immediate data.

(*Note:* When this instruction is used to modify an output port, the value used as the original port data will be read from the output data latch, *not* the input pins.)

**Example:** If the Accumulator holds 0C3H (11000011B) and register 0 holds 0AAH (10101010B) then the instruction,

XRL A,R0

will leave the Accumulator holding the value 69H (01101001B).

When the destination is a directly addressed byte, this instruction can complement combinations of bits in any RAM location or hardware register. The pattern of bits to be complemented is then determined by a mask byte, either a constant contained in the instruction or a variable computed in the Accumulator at run-time. The instruction,

XRL P1,#00110001B

will complement bits 5, 4, and 0 of output Port 1.

**XRL A,Rn**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 | 1 | 1 | 0 | 1 | r | r | r |
|---|---|---|---|---|---|---|---|

**Operation:** XRL
(A) ← (A) ⩛ ($R_n$)

**XRL A,direct**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | direct address |
|---|---|---|---|---|---|---|---|---|

**Operation:** XRL
(A) ← (A) ⩛ (direct)

**XRL A,@Ri**

**Bytes:** 1

**Cycles:** 1

**Encoding:**

| 0 | 1 | 1 | 0 | 0 | 1 | 1 | i |
|---|---|---|---|---|---|---|---|

**Operation:** XRL
(A) ← (A) ⩛ ($R_i$)

**XRL A,#data**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | immediate data |
|---|---|---|---|---|---|---|---|---|

**Operation:** XRL
(A) ← (A) ⩛ #data

**XRL direct,A**

**Bytes:** 2

**Cycles:** 1

**Encoding:**

| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | direct address |
|---|---|---|---|---|---|---|---|---|

**Operation:** XRL
(direct) ← (direct) ⩛ (A)

**XRL direct,#data**

**Bytes:** 3

**Cycles:** 2

**Encoding:**

| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | direct address | immediate data |
|---|---|---|---|---|---|---|---|---|---|

**Operation:** XRL
(direct) ← (direct) ⩛ #data

## EPROM PRODUCTS

Most of the 80C51 derivative products offered by Philips are supported with an EPROM version.

All EPROM products are available in both windowed DIP and OTP package configurations. The windowed DIP package allows the EPROM to be erased under a strong UV light source, making program development easier and faster. The OTP (One Time Programmable) version cannot be erased because there is no window through which the die could be exposed to UV light. While the EPROM can only be programmed once in the OTP package, the part costs less than in windowed DIP and therefore offers an advantage for those not desiring to use the masked ROM version of the part.

The EPROM products are fully supported on the industry standard EPROM programmers.

## Programming the 87C51

The setup for programming the microcontroller is shown in Figure 1. Note that the part is running with a 4 to 6 MHz oscillator. The clock must be running because the device is executing internal address and program data transfers during the programming.

To program the 87C51, the address of the EPROM location to be programmed is applied to ports 1 and 2 as shown in Figure 1. The code byte to be programmed into this location is applied to port 0. RST, $\overline{PSEN}$, and the pins of ports 2 and 3 specified in Table 1 are held at the "Program Code Data" levels specified in the table. The ALE/$\overline{PROG}$ is then pulsed low 25 times to program the addressed location.

### Encryption Table

The encryption table is a feature of the 87C51, and its derivatives, that protects the code from being easily read by anyone other than the programmer. The encryption table is 16 to 64 bytes of code, depending on the microcontroller, that are exclusive NORed with the program code data as it is read out. The first byte is XNORed with the first location read, the second with the second read, etc. through the sixteenth byte read. The seventeenth byte is XNORed with the first byte of the encryption table, the eighteenth with the second, etc. and on in sixteen-byte groups.

After the Encryption table has been programmed the user has to know its contents in order to correctly decode the program code data. The encryption table itself cannot be read out.

The encryption table is programmed in the same manner as the program memory, but using the "Pgm Encryption Table" levels specified in Table 1. After the encryption table is programmed, verification cycles will produce only encrypted information.

### Security Bit

There are two security bits on the 87C51 that, when set, prevent the program data memory from being read out or programmed further. To program the security bits, repeat the programming sequence using the "Pgm Security Bit" levels specified in Table 1.

After the first security bit is programmed, further programming of the code memory or the encryption table is disabled. The other security bit can of course still be programmed. With only security bit one programmed, the memory can still be read out for program verification. After the second security bit is programmed, it is no longer possible to read out (verify) the program memory.

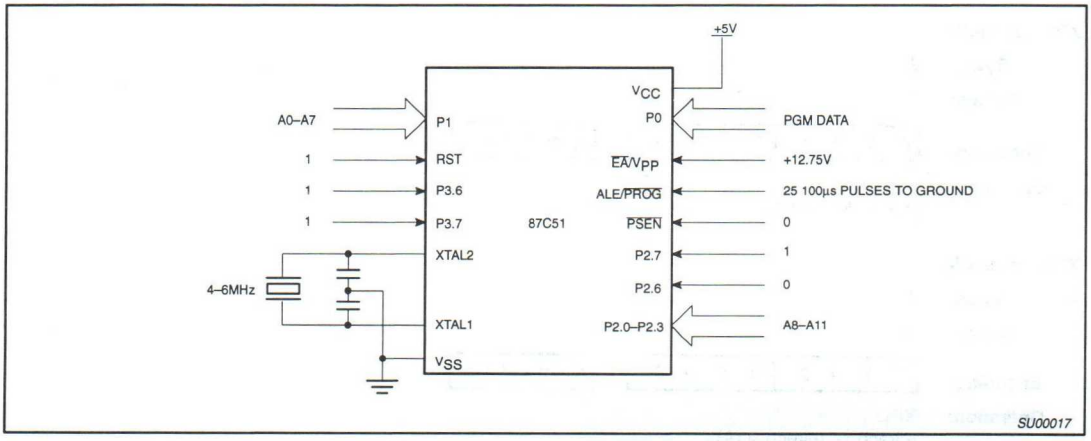



**Figure 1. Programming Configuration**

**Table 1. EPROM Programming Modes**

| MODE | RST | PSEN | ALE/PROG | EA/$V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm security bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm security bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |

**NOTES:**
1. "0" = valid low for that pin, "1" = valid high for that pin.
2. $V_{PP}$ = 12.75 ±0.25V.
3. $V_{CC}$ = 5V ±10% during programming and verification.
* ALE/PROG receives 25 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100ms (±10µs) and high for a minimum of 10µs.

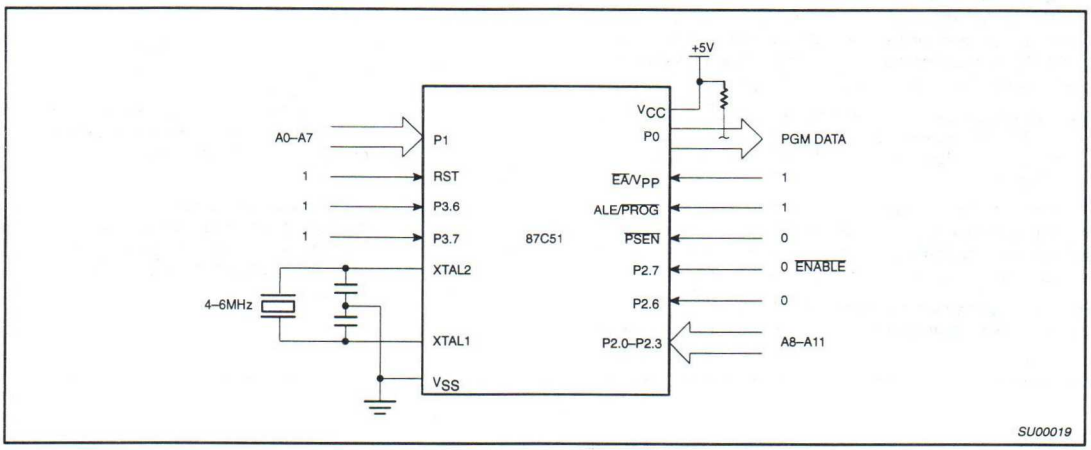



**Figure 2. Program Verification**

## Program Verification

If security bit 2 has not been programmed the on-chip program memory can be read out for program verification. To verify the contents of the program memory, the address of the location to be read is applied to ports 1 and 2 as shown in Figure 2. The other pins are held at the "Verify Code Data" levels indicated in Table 1. The contents of the addressed location will appear on port 0. For this operation external pull-ups are required on port 0 as shown in Figure 2. Note that if the encryption table has been programmed the data presented at port 0 will be the exclusive NOR of the program byte with a byte from the encryption table.

### Signature Bytes

The 87C51 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C51 manufactured by Philips.

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:

| | | |
|---|---|---|
| (030H) | = | 15H indicates the part is made by Philips |
| (031H) | = | 90H 87C451 |
| | | 92H 87C51 |
| | | 94H 87C552 |
| | | 96H 87C550 |
| | | 97H 87C52 |
| | | 99H 87C654/87C652 |
| | | 9BH 87C528 |
| | | 9CH 87C592 |
| | | 9DH 87C524 |
| | | 9EH 87C598 |
| | | B0H 87C575 |
| | | B1H 87C51FA |
| | | B2H 87C51FB |
| | | B3H 87C51FC |
| | | B5H 89CE558 |
| | | B6H 87C576 |
| | | BBH 87C504 |
| | | 4BH 87C055 |
| (060H) | | FCH 87C51FC |

## EPROM Erasure

Erasure of the EPROM occurs when the chip is exposed to light with wavelengths shorter than 4000 angstroms. Sunlight and fluorescent lighting have wavelengths in this range, so exposure to these light sources over an extended period of time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. It is recommended, for this reason, that an opaque label be placed over the window. If the part is subject to elevated temperatures or an environment where solvents are used, Kapton tape (Fluorglas part number 2345-5 or its equivalent) can be used.

The recommended erasure procedure is to expose the chip to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-sec/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 40 minutes, at a distance of 1 inch, is adequate.

## Programming the 87C750, 87C751 and 87C752

The 87C750, 87C751 and 87C752 are programmed using a Quick-pulse programming algorithm that is similar to that used for the 87C51. It differs from the 87C51 in that a serial data stream is used to place the 87C751 in the programming mode.

Figure 3 shows a block diagram of the programming configuration for the 87C751. Port pin P0.2 is used for the programming voltage supply input ($V_{PP}$ signal). Port pin P0.1 is used for the program (PGM) signal.

Port 3 accepts the address input for the EPROM location to be programmed. Both the high and low components of the eleven-bit address are presented to the part through port 3. Multiplexing of the address components is performed using ASEL (P0.0).

Port 1 is used as a bidirectional data bus during programming and verify operations. During the programming mode, it accepts the byte to be programmed. In the verify mode, it returns the contents of the specified address location.

The X1 pin is the oscillator input and receives the master system clock. This clock should be between 1.2 and 6MHz.

The RESET pin is used to accept the serial data stream that places the 87C751 into various programming modes. This pattern consists of a 10-bit code with the LSB send first. Each bit is synchronized to the clock input X1.

To program the 87C751 the part must be put into the programming mode by presenting the proper serial code (see Table 2) to the RESET pin. To do this RESET should be held high for at least two machine cycles. Port pins P0.1 and P0.2 will be at VOH as a result of this, but they must be driven high prior to sending the serial data stream on the RESET pin. The serial data bits can now be transmitted over the RESET pin placing the 87C751 into one of the programming modes. Following the transmission of the last data bit, the reset pin should be held low.

Next the address information for the location to be programmed is placed on Port 3 and ASEL is used to perform the address multiplexing. ASEL should be driven high and then Port 3 driven with the high-order address bits. ASEL is then driven low, latching the high-order bits internally. Port 3 can now be driven with the low 8 bits of the address, completing the addressing of the location to be programmed.

A high-voltage $V_{PP}$ level is now applied to the $V_{PP}$ input. This sets Port 1 as an input port. The data to be programmed to the EPROM array should be placed on Port 1. A series of 25 programming pulses is now applied to the PGM pin (P0.1) to program the addressed EPROM location.

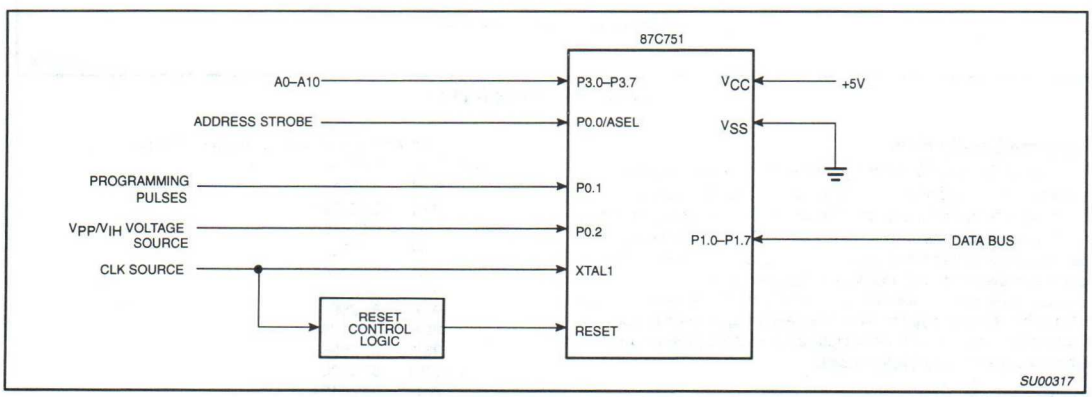



**Figure 3. Programming Configuration**

**Table 2. 87C750, 87C751, and 87C752 Serial Codes**

| OPERATION | SERIAL CODE | P0.1 (PGM/) | P0.2 ($V_{PP}$) |
|---|---|---|---|
| Program user EPROM | 296H | –* | $V_{PP}$ |
| Verify user EPROM | 296H | $V_{IH}$ | $V_{IH}$ |
| Program key EPROM | 292H | –* | $V_{PP}$ |
| Verify key EPROM | 292H | $V_{IH}$ | $V_{IH}$ |
| Program security bit 1 | 29AH | –* | $V_{PP}$ |
| Program security bit 2 | 298H | –* | $V_{PP}$ |
| Verify security bits | 29AH | $V_{IH}$ | $V_{IH}$ |
| Read signature bytes | 294H | $V_{IH}$ | $V_{IH}$ |

**NOTE:**
* Pulsed from $V_{IH}$ to $V_{IL}$ and returned to $V_{IH}$.

## Program Verification

The EPROM array can be verified by placing the part in the programming mode as described above and forcing the $V_{PP}$ pin to the $V_{OH}$ level. Four machine cycles after addressing a location the contents of the addressed location will appear on Port 1.

**87C750, 87C751 and 87C752 Signature Bytes**

The signature bytes for the 87C750, 87C751 and 87C752 are read differently and are in different locations than those on the 87C51. Due to its reduced pin count, the part has to be put into "Signature Byte Read Mode" by placing a 10-bit serial data stream on the Reset pin. The proper code and the conditions of P0.1 and P0.2, for this mode, are shown in Table 2.

Once the part has been placed into the Signature Byte Read Mode, the signature bytes can be read by the same procedure as a normal verification of locations 01EH and 01FH. The values are:

01EH = 15H indicates the part is made by Philips
01FH = 91H - 87C751
01FH = 95H - 87C752

**Programming Features**

The 87C751 has all of the special programming features incorporated within its EPROM array that the 87C51 has. It has an encryption key table and two security bits. These function exactly as they do in the 87C51. They are programmed or verified by sending the proper code over the RESET pin (see Table 2) and then following the 87C751 programming procedure as described previously.

**Erasure Characteristics**

The erasure procedure is exactly the same as that described for the 87C51.

# Section 3
## 80C51 Family Derivatives

### 80C51-Based 8-Bit Microcontrollers

CONTENTS


| | | |
|---|---|---|
| **80C31/80C51/87C51** | CMOS single-chip 8-bit microcontrollers | 3-3 |
| **83C51FA/83C51FB/ 83C51FC/80C51FA** | CMOS single-chip 8-bit microcontrollers | 3-25 |
| **87C51FA/87C51FB** | CMOS single-chip 8-bit microcontrollers | 3-61 |
| **87C51FC** | CMOS single-chip 8-bit microcontroller | 3-89 |
| 80CL31/80CL51 | Low-voltage single-chip 8-bit microcontrollers | 3-117 |
| **87L51FA/87L51FB** | CMOS single-chip 3.0V 8-bit microcontrollers | 3-150 |
| **80C32/87C52** | CMOS single-chip 8-bit microcontrollers | 3-166 |
| **80C52/80C54/80C58** | CMOS single-chip 8-bit microcontrollers | 3-188 |
| **87C54/87C58** | CMOS single-chip 8-bit microcontrollers | 3-215 |
| **83C145; 83C845 83C055; 87C055** | Microcontrollers for TV and video (MTV) | 3-236 |
| 80CL410/83CL410 | Low voltage/low power single-chip 8-bit microcontroller with $I^2C$ | 3-270 |
| **80C451/83C451/87C451** | CMOS single-chip 8-bit microcontrollers | 3-292 |
| **80C453/83C453/87C453** | CMOS single-chip 8-bit microcontrollers | 3-311 |
| **83C508/87C508** | CMOS single-chip 8-bit microcontrollers | 3-334 |
| **P83C524** | 8-bit microcontroller | 3-351 |
| 87C524 | CMOS single-chip 8-bit microcontroller | 3-372 |
| 80C528/83C528 | CMOS single-chip 8-bit microcontrollers | 3-392 |
| **87C528** | CMOS single-chip 8-bit microcontroller | 3-410 |
| **P8xCE528** | 8-bit microcontroller with EMC | 3-430 |
| 80C550/83C550/87C550 | CMOS single-chip 8-bit microcontroller with A/D and watchdog timer | 3-450 |
| **8XC552/562 overview** | | 3-473 |
| 8XC552 OVERVIEW | | 3-473 |
| 83C562 OVERVIEW | | 3-473 |
| Differences From the 80C51 | | 3-473 |
| Program Memory | | 3-473 |
| Data Memory | | 3-473 |
| Special Function Registers | | 3-473 |
| Timer T2 | | 3-474 |
| Timer T3, The Watchdog Timer | | 3-480 |
| Serial I/O | | 3-481 |
| Reset Circuitry | | 3-512 |
| Interrupts | | 3-513 |
| I/O Port Structure | | 3-517 |
| Port 1 Operation | | 3-517 |
| Port 5 Operation | | 3-517 |
| Pulse Width Modulated Outputs | | 3-517 |
| Analog-to-Digital Converter | | 3-517 |
| Power Reduction Modes | | 3-523 |
| Memory Organization | | 3-525 |
| **80C552/83C552** | Single-chip 8-bit microcontroller with 10-bit A/D, capture/compare timer, high-speed outputs, PWM | 3-530 |
| 87C552 | Single-chip 8-bit microcontroller with 10-bit A/D, capture/compare timer, high-speed outputs, PWM | 3-550 |
| **P83CE558/P80CE558/ P89CE558** | Single-chip 8-bit microcontroller | 3-571 |
| **P83CE559/P80CE559** | Single-chip 8-bit microcontroller | 3-640 |
| 80C562/83C562 | Single-chip 8-bit microcontroller with 8-bit A/D, capture/compare timer, high-speed outputs, PWM | 3-708 |

**CONTENTS** (Continued)


| | | |
|---|---|---|
| **80C575/83C575/87C575** | CMOS single-chip 8-bit microcontrollers | 3-721 |
| **83C576/87C576** | CMOS single-chip 8-bit microcontrollers | 3-756 |
| **P80CL580; P83CL580** | Low voltage 8-bit microcontrollers | 3-798 |
| **P8xC592** | 8-bit microcontroller with on-chip CAN | 3-869 |
| **P8xCE598** | 8-bit microcontroller with on-chip CAN | 3-969 |
| **80C652/83C652** | CMOS single-chip 8-bit microcontrollers | 3-992 |
| **87C652** | CMOS single-chip 8-bit microcontroller | 3-1010 |
| **83C654** | CMOS single-chip 8-bit microcontroller | 3-1030 |
| **87C654** | CMOS single-chip 8-bit microcontroller | 3–1048 |
| **83CE654** | CMOS single-chip 8-bit microcontroller with Electromagnetic Compatibility improvements | 3-1069 |
| **83C748/87C748** | CMOS single-chip 8-bit microcontrollers | 3-1083 |
| **83C749/87C749** | CMOS single-chip 8-bit microcontrollers | 3-1095 |
| **TPM749** | Microcontroller with TrackPoint™ microcode from IBM | 3-1110 |
| **83C750/87C750** | CMOS single-chip 8-bit microcontrollers | 3-1119 |
| **83C751/87C751** | CMOS single-chip 8-bit microcontrollers | 3-1129 |
| **83C752/87C752** | CMOS single-chip 8-bit microcontroller with A/D, PWM | 3-1147 |
| **83C754/87C754** | CMOS single-chip 8-bit microcontrollers | 3-1165 |
| **TPM754** | Microcontroller with TrackPoint™ microcode from IBM | 3-1187 |
| **P83CL781; P83CL782** | Low voltage 8-bit microcontrollers | 3-1195 |
| 80C851/83C851 | CMOS single-chip 8-bit microcontroller with on-chip EEPROM | 3-1251 |

# CMOS single-chip 8-bit microcontrollers

# 80C31/80C51/87C51

## DESCRIPTION

The Philips 80C31/80C51/87C51 is a high-performance microcontroller fabricated with Philips high-density CMOS technology. The CMOS 8XC51 is functionally compatible with the NMOS 8031/8051 microcontrollers. The Philips CMOS technology combines the high speed and density characteristics of HMOS with the low power attributes of CMOS. Philips epitaxial substrate minimizes latch-up sensitivity.

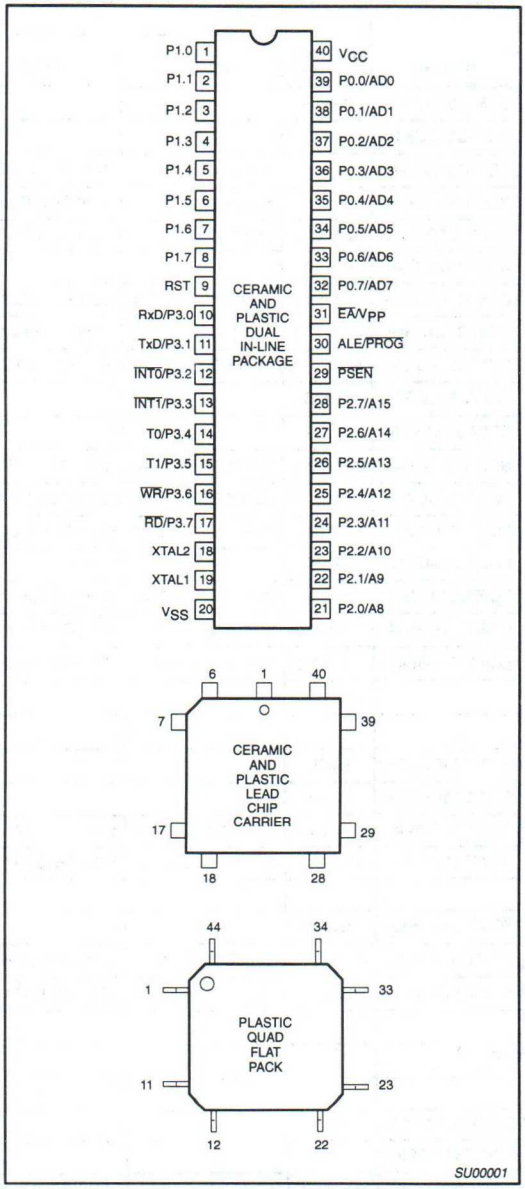
The 8XC51 contains a 4k × 8 ROM (80C51) EPROM (87C51), a 128 × 8 RAM, 32 I/O lines, two 16-bit counter/timers, a five-source, two-priority level nested interrupt structure, a serial I/O port for either multi-processor communications, I/O expansion or full duplex UART, and on-chip oscillator and clock circuits.

In addition, the device has two software selectable modes of power reduction—idle mode and power-down mode. The idle mode freezes the CPU while allowing the RAM, timers, serial port, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

## FEATURES

- 8031/8051 compatible
  - 4k × 8 ROM (80C51)
  - 4k × 8 EPROM (87C51)
  - ROMless (80C31)
  - 128 × 8 RAM
  - Two 16-bit counter/timers
  - Full duplex serial channel
  - Boolean processor
- Memory addressing capability
  - 64k ROM and 64k RAM
- Power control modes:
  - Idle mode
  - Power-down mode
- CMOS and TTL compatible
- Five speed ranges at $V_{CC}$ = 5V
  - 12MHz
  - 16MHz
  - 24MHz
  - 33MHz
- Five package styles
- Extended temperature ranges
- OTP package available

## PIN CONFIGURATIONS




SEE PAGE 3-6 FOR QFP AND LCC PIN FUNCTIONS.

## ORDERING INFORMATION

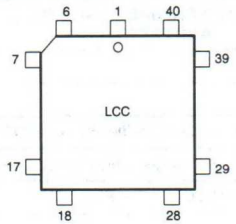
| EPROM | DRAWING NUMBER | PHILIPS NORTH AMERICA | | | | |
|---|---|---|---|---|---|---|
| | | ROMless | ROM | DRAWING NUMBER | TEMPERATURE RANGE °C AND PACKAGE[1] | Freq MHz |
| SC87C51CCF40 | 0590B | | | | 0 to +70, Ceramic Dual In-line Package, UV | 3.5 to 12 |
| SC87C51CCK44 | 1472A | | | | 0 to +70, Ceramic Leaded Chip Carrier, UV | 3.5 to 12 |
| SC87C51CCN40 | SOT129-1 | SC80C31BCCN40 | SC80C51BCCN40 | SOT129-1 | 0 to +70, Plastic Dual In-line Package, OTP | 3.5 to 12 |
| SC87C51CCA44 | SOT187-2 | SC80C31BCCA44 | SC80C51BCCA44 | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier, OTP | 3.5 to 12 |
| SC87C51CCB44 | SOT307-2 | SC80C31BCCB44 | SC80C51BCCB44 | SOT307-2 | 0 to +70, Plastic Quad Flat Pack, OTP | 3.5 to 12 |
| SC87C51ACF40 | 0590B | | | | –40 to +85, Ceramic Dual In-line Package, UV | 3.5 to 12 |
| SC87C51ACN40 | SOT129-1 | SC80C31BACN40 | SC80C51BACN40 | SOT129-1 | –40 to +85, Plastic Dual In-line Package, OTP | 3.5 to 12 |
| SC87C51ACA44 | SOT187-2 | SC80C31BACA44 | SC80C51BACA44 | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier, OTP | 3.5 to 12 |
| SC87C51ACB44 | SOT307-2 | SC80C31BACB44 | SC80C51BACB44 | SOT307-2 | –40 to +85, Plastic Quad Flat Pack, OTP | 3.5 to 12 |
| SC87C51CGF40 | 0590B | | | | 0 to +70, Ceramic Dual In-line Package, UV | 3.5 to 16 |
| SC87C51CGK44 | 1472A | | | | 0 to +70, Ceramic Leaded Chip Carrier, UV | 3.5 to 16 |
| SC87C51CGN40 | SOT129-1 | SC80C31BCGN40 | SC80C51BCGN40 | SOT129-1 | 0 to +70, Plastic Dual In-line Package, OTP | 3.5 to 16 |
| SC87C51CGA44 | SOT187-2 | SC80C31BCGA44 | SC80C51BCGA44 | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier, OTP | 3.5 to 16 |
| SC87C51CGB44 | SOT307-2 | SC80C31BCGB44 | SC80C51BCGB44 | SOT307-2 | 0 to +70, Plastic Quad Flat Pack, OTP | 3.5 to 16 |
| SC87C51AGF40 | 0590B | | | | –40 to +85, Ceramic Dual In-line Package, UV | 3.5 to 16 |
| SC87C51AGN40 | SOT129-1 | SC80C31BAGN40 | SC80C51BAGN40 | SOT129-1 | –40 to +85, Plastic Dual In-line Package, OTP | 3.5 to 16 |
| SC87C51AGA44 | SOT187-2 | SC80C31BAGA44 | SC80C51BAGA44 | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier, OTP | 3.5 to 16 |
| SC87C51AGB44 | SOT307-2 | SC80C31BAGB44 | SC80C51BAGB44 | SOT307-2 | –40 to +85, Plastic Quad Flat Pack, OTP | 3.5 to 16 |
| | | | | | | |
| | | | | | | |
| SC87C51CPF40 | 0590B | | | | 0 to +70, Ceramic Dual In-line Package, UV | 3.5 to 24 |
| SC87C51CPK44 | 1472A | | | | 0 to +70, Ceramic Leaded Chip Carrier, UV | 3.5 to 24 |
| SC87C51CPN40 | SOT129-1 | SC80C31BCPN40 | SC80C51BCPN40 | SOT129-1 | 0 to +70, Plastic Dual In-line Package, OTP | 3.5 to 24 |
| SC87C51CPA44 | SOT187-2 | SC80C31BCPA44 | SC80C51BCPA44 | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier, OTP | 3.5 to 24 |
| | | | | | | |
| SC87C51APF40 | 0590B | | | | –40 to +85, Ceramic Dual In-line Package, UV | |
| SC87C51APN40 | SOT129-1 | SC80C31BAPN40 | SC80C51BAPN40 | SOT129-1 | –40 to +85, Plastic Dual In-line Package, OTP | 3.5 to 24 |
| SC87C51APA44 | SOT187-2 | SC80C31BAPA44 | SC80C51BAPA44 | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier, OTP | 3.5 to 24 |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| SC87C51CYF40 | 0590B | | | | 0 to +70, Ceramic Dual In-line Package, UV | 3.5 to 33 |
| SC87C51CYK44 | 1472A | | | | 0 to +70, Ceramic Leaded Chip Carrier, UV | 3.5 to 33 |
| SC87C51CYN40 | SOT129-1 | SC80C31BCYN40 | SC80C51BCYN40 | SOT129-1 | 0 to +70, Plastic Dual In-line Package, OTP | 3.5 to 33 |
| SC87C51CYA44 | SOT187-2 | SC80C31BCYA44 | SC80C51BCYA44 | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier, OTP | 3.5 to 33 |
| | | | | | | |

1. OTP = One Time Programmable EPROM. UV = UV Erasable EPROM
2. SOT311 replaced by SOT307-2.

**ORDERING INFORMATION** (Continued)

| PHILIPS | | | | | |
|---|---|---|---|---|---|
| ROMless (ORDER NUMBER) | ROMless (MARKING NUMBER) | ROM | DRAWING NUMBER | TEMPERATURE RANGE °C AND PACKAGE[1] | Freq MHz |
| | | | | | |
| | | | | | |
| PCB80C31-2 N | PCB80C31BH2-12P | PCB80C51BH-2P | SOT129-1 | 0 to +70, Plastic Dual In-line Package, OTP | 0.5 to 12 |
| PCB80C31-2 A | PCB80C31BH2-12WP | PCB80C51BH-2WP | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier, OTP | 0.5 to 12 |
| | PCB80C31BH2-12H | PCB80C51BH-2H | SOT307-2[2] | 0 to +70, Plastic Quad Flat Pack, OTP | 0.5 to 12 |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| PCB80C31-3 N | PCB80C31BH3-16P | PCB80C51BH-3P | SOT129-1 | 0 to +70, Plastic Dual In-line Package, OTP | 1.2 to 16 |
| PCB80C31-3 A | PCB80C31BH3-16WP | PCB80C51BH-3WP | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier, OTP | 1.2 to 16 |
| | PCB80C31BH3-16H | PCB80C51BH-3H | SOT307-2[2] | 0 to +70, Plastic Quad Flat Pack, OTP | 1.2 to 16 |
| | | | | | |
| PCF80C31-3 N | PCF80C31BH3-16P | PCF80C51BH-3P | SOT129-1 | –40 to +85, Plastic Dual In-line Package, OTP | 1.2 to 16 |
| PCF80C31-3 A | PCF80C31BH3-16WP | PCF80C51BH-3WP | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier, OTP | 1.2 to 16 |
| | PCF80C31BH3-16H | PCF80C51BH-3H | SOT307-2[2] | –40 to +85, Plastic Quad Flat Pack, OTP | 1.2 to 16 |
| | PCA80C31BH3-16P | PCA80C51BH-3P | SOT129-1 | –40 to +125, Plastic Dual In-line Package | 1.2 to 16 |
| | PCA80C31BH3-16WP | PCA80C51BH-3WP | SOT187-2 | –40 to +125, Plastic Leaded Chip Carrier | 1.2 to 16 |
| | | | | | |
| | | | | | |
| PCB80C31-4 N | PCB80C31BH4-24P | PCB80C51BH-4P | SOT129-1 | 0 to +70, Plastic Dual In-line Package, OTP | 1.2 to 24 |
| PCB80C31-4 A | PCB80C31BH4-24WP | PCB80C51BH-4WP | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier, OTP | 1.2 to 24 |
| | PCB80C31BH4-24H | PCB80C51BH-4H | SOT307-2[2] | 0 to +70, Plastic Quad Flat Pack, OTP | 1.2 to 24 |
| | | | | | |
| PCF80C31-4 N | PCF80C31BH4-24P | PCF80C51BH-4P | SOT129-1 | –40 to +85, Plastic Dual In-line Package, OTP | 1.2 to 24 |
| PCF80C31-4 A | PCF80C31BH4-24WP | PCF80C51BH-4WP | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier, OTP | 1.2 to 24 |
| | PCF80C31BH4-24H | PCF80C51BH-4H | SOT307-2[2] | –40 to +85, Plastic Leaded Chip Carrier, OTP | 1.2 to 24 |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| PCB80C31-5 N | PCB80C31BH5-30P | PCB80C51BH-5P | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 1.2 to 33 |
| PCB80C31-5 A | PCB80C31BH5-30WP | PCB80C51BH-5WP | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 1.2 to 33 |
| PCB80C31-5 B | PCB80C31BH5-30H | PCB80C51BH-5H | SOT307-2[2] | 0 to +70, Plastic Quad Flat Pack | 1.2 to 33 |

## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

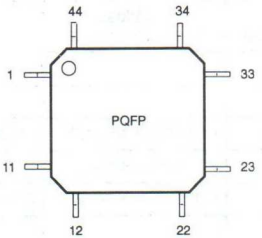



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | NC* | 16 | P3.4/T0 | 31 | P2.7/A15 |
| 2 | P1.0 | 17 | P3.5/T1 | 32 | $\overline{PSEN}$ |
| 3 | P1.1 | 18 | P3.6/$\overline{WR}$ | 33 | ALE/$\overline{PROG}$ |
| 4 | P1.2 | 19 | P3.7/$\overline{RD}$ | 34 | NC* |
| 5 | P1.3 | 20 | XTAL2 | 35 | $\overline{EA}$/$V_{PP}$ |
| 6 | P1.4 | 21 | XTAL1 | 36 | P0.7/AD7 |
| 7 | P1.5 | 22 | $V_{SS}$ | 37 | P0.6/AD6 |
| 8 | P1.6 | 23 | NC* | 38 | P0.5/AD5 |
| 9 | P1.7 | 24 | P2.0/A8 | 39 | P0.4/AD4 |
| 10 | RST | 25 | P2.1/A9 | 40 | P0.3/AD3 |
| 11 | P3.0/RxD | 26 | P2.2/A10 | 41 | P0.2/AD2 |
| 12 | NC* | 27 | P2.3/A11 | 42 | P0.1/AD1 |
| 13 | P3.1/TxD | 28 | P2.4/A12 | 43 | P0.0/AD0 |
| 14 | P3.2/$\overline{INT0}$ | 29 | P2.5/A13 | 44 | $V_{CC}$ |
| 15 | P3.3/$\overline{INT1}$ | 30 | P2.6/A14 | | |

* DO NOT CONNECT

SU00002

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS

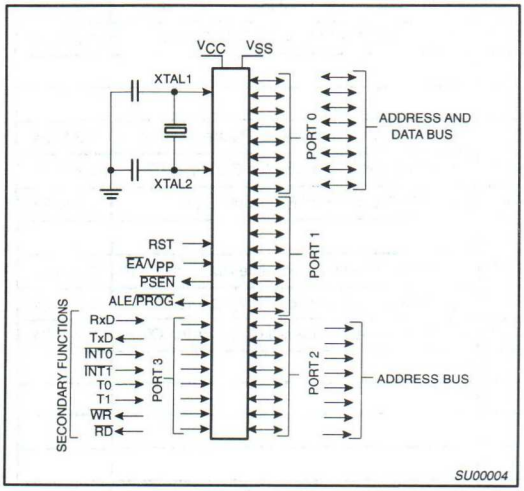



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | P1.5 | 16 | $V_{SS}$ | 31 | P0.6/AD6 |
| 2 | P1.6 | 17 | NC* | 32 | P0.5/AD5 |
| 3 | P1.7 | 18 | P2.0/A8 | 33 | P0.4/AD4 |
| 4 | RST | 19 | P2.1/A9 | 34 | P0.3/AD3 |
| 5 | P3.0/RxD | 20 | P2.2/A10 | 35 | P0.2/AD2 |
| 6 | NC* | 21 | P2.3/A11 | 36 | P0.1/AD1 |
| 7 | P3.1/TxD | 22 | P2.4/A12 | 37 | P0.0/AD0 |
| 8 | P3.2/$\overline{INT0}$ | 23 | P2.5/A13 | 38 | $V_{CC}$ |
| 9 | P3.3/$\overline{INT1}$ | 24 | P2.6/A14 | 39 | NC* |
| 10 | P3.4/T0 | 25 | P2.7/A15 | 40 | P1.0 |
| 11 | P3.5/T1 | 26 | $\overline{PSEN}$ | 41 | P1.1 |
| 12 | P3.6/$\overline{WR}$ | 27 | ALE/$\overline{PROG}$ | 42 | P1.2 |
| 13 | P3.7/$\overline{RD}$ | 28 | NC* | 43 | P.13 |
| 14 | XTAL2 | 29 | $\overline{EA}$/$V_{PP}$ | 44 | P1.4 |
| 15 | XTAL1 | 30 | P0.7/AD7 | | |

* DO NOT CONNECT

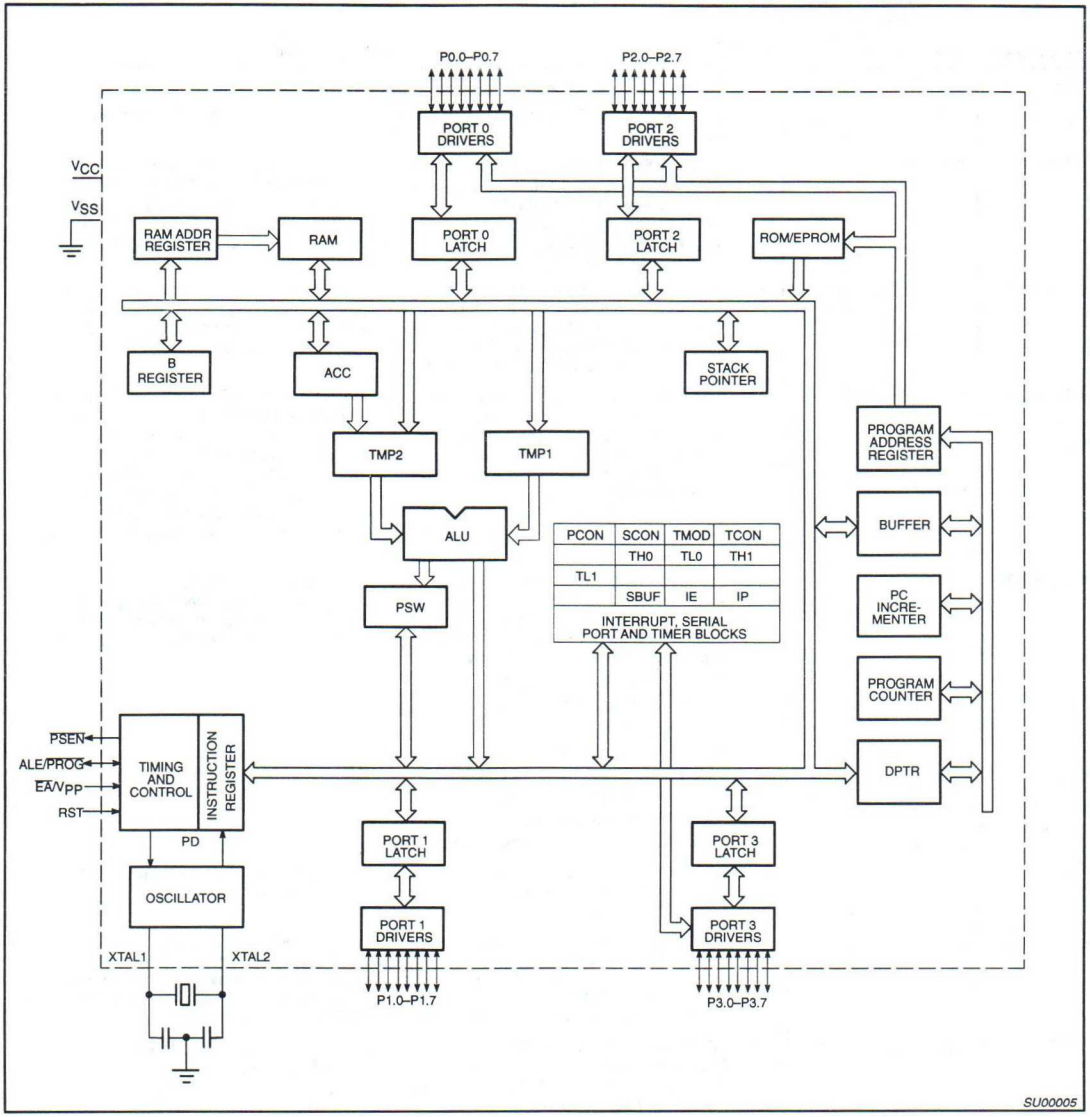
SU00003

## LOGIC SYMBOL




SU00004

## BLOCK DIAGRAM



## PIN DESCRIPTION

| MNEMONIC | PIN NO. DIP | PIN NO. LCC | PIN NO. QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification in the 87C51. External pull-ups are required during program verification. |
| P1.0–P1.7 | 1–8 | 2–9 | 40-44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also receives the low-order address byte during program memory verification. |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{CC}$. |
| ALE/$\overline{PROG}$ | 30 | 33 | 27 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the device is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}$/$V_{PP}$ | 31 | 35 | 29 | I | **External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H to 0FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 0FFFH. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**Table 1. 80C52/80C54/80C58 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| AUXR# | Auxiliary | 8EH | – | – | – | – | – | – | – | AO | xxxxxxx0B |
| AUXR1# | Auxiliary 1 (Note 2) | A2H | – | – | – | – | WUPD | 0 | – | DPS | xxxx00x0B |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data Pointer (2 bytes) | | | | | | | | | | |
| DPH | Data Pointer High | 83H | | | | | | | | | 00H |
| DPL | Data Pointer Low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE* | Interrupt Enable | A8H | EA | EC | ET2 | ES | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP* | Interrupt Priority | B8H | – | – | PT2 | PS | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| IPH# | Interrupt Priority High | B7H | – | – | PT2H | PSH | PT1H | PX1H | PT0H | PX0H | x0000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | – | – | – | – | – | – | T2EX | T2 | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | AD15 | AD14 | AD13 | AD12 | AD11 | AD10 | AD9 | AD8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TxD | RxD | FFH |
| | | | | | | | | | | | |
| PCON#[1] | Power Control | 87H | SMOD1 | SMOD0 | – | – | GF1 | GF0 | PD | IDL | 00xx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program Status Word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| SADDR# | Slave Address | A9H | | | | | | | | | 00H |
| SADEN# | Slave Address Mask | B9H | | | | | | | | | 00H |
| SBUF | Serial Data Buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial Control | 98H | SM0/FE | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack Pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer Control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| T2MOD# | Timer 2 Mode Control | C9H | – | – | – | – | – | – | T2OE | DCEN | xxxxxx00B |
| TH0 | Timer High 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer High 1 | 8DH | | | | | | | | | 00H |
| TL0 | Timer Low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer Low 1 | 8BH | | | | | | | | | 00H |
| TMOD | Timer Mode | 89H | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
– Reserved bits.
1. Reset value depends on reset source.
2. Available only on SC80C51.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the logic symbol.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-up reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles.

## IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. the control bits for the reduced power modes are in the special function register PCON.

Table 2 shows the state of I/O ports during low current operating modes.

**Table 2. External Pin Status During Idle and Power-Down Modes**

| MODE | PROGRAM MEMORY | ALE | PSEN | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

## ROM CODE SUBMISSION

When submitting ROM code for the 80C51, the following must be specified:

1. 4k byte user ROM data
2. 64 byte ROM encryption key (SC80C51 only)
3. ROM security bits (SC80C51 only).

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 0FFFH | DATA | 7:0 | User ROM Data |
| 1000H to 101FH | KEY | 7:0 | ROM Encryption Key |
| 1020H | SEC | 0 | ROM Security Bit 1 |
| 1020H | SEC | 1 | ROM Security Bit 2 |

**Security Bit 1:** When programmed, this bit has two effects on masked ROM parts:

1. External MOVC is disabled, and
2. EA# is latched on Reset.

**Security Bit 2:** When programmed, this bit inhibits Verify User ROM.

**Electrical Deviations from Commercial Specifications for Extended Temperature Range (87C51)**
DC and AC parameters not included here are the same as in the commercial temperature range table.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V (Philips North America SC87C51);
For SC87C51 (33MHz only), $T_{amb}$ = 0°C to +70°C, $V_{CC}$ = 5V ±5%
$T_{amb}$ = –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V (PCB80C31/51 and PCF80C31/51 Philips Parts Only)

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$ (Philips North America) | | –0.5 | $0.2V_{CC}$–0.15 | V |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$ (Philips) | | –0.5 | $0.2V_{CC}$–0.25 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | | –0.5 | $0.2V_{CC}$–0.45 | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST | | $0.2V_{CC}$+1 | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage to XTAL1, RST | | $0.7V_{CC}$+0.1 | $V_{CC}$+0.5 | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3 | $V_{IN}$ = 0.45V | | –75 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3 | $V_{IN}$ = 2.0V | | –750 | µA |
| $I_{CC}$ | Power supply current: | $V_{CC}$ = 4.5–5.5V | | | |
| | Active mode[1] @ 16MHz (Philips PCB80C31/51, PCF80C31/51) | | | 25 | mA |
| | Active mode @ 12MHz (Philips North America SC87C51) | | | 20 | mA |
| | Idle mode[2] @ 16MHz (Philips PCB80C31/51, PCF80C31/51) | | | 6.5 | mA |
| | Idle mode @ 12MHz (Philips North America SC87C51) | | | 5 | mA |
| | Power-down mode[3] (Philips PCB80C31/51, PCF80C31/51) | | | 75 | µA |
| | Power-down mode (Philips North America SC87C51) | | | 50 | µA |

**NOTES:**
1. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 10ns; $V_{IL} = V_{SS}$ + 0.5V; $V_{IH} = V_{CC}$ – 0.5V; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = $V_{CC}$.
2. The idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 10ns; $V_{IL} = V_{SS}$ + 0.5V; $V_{IH} = V_{CC}$ – 0.5V; XTAL2 not connected; $\overline{EA}$ = Port 0 = $V_{CC}$; RST = $V_{SS}$.
3. The power-down current is measured with all output pins disconnected, XTAL2 not connected, $\overline{EA}$ = Port 0 = $V_{CC}$; RST = $V_{SS}$.

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | 0 to +70 or –40 to +85 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Maximum $I_{OL}$ per I/O pin | 15 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.5 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±20%, $V_{SS}$ = 0V (PCB80C31/51 and PCF80C31/51) (12, 16, and 24MHz versions)
$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V (87C51 12, 16, and 24MHz versions) (PCB80C31/51 33MHz version);
For SC87C51 (33MHz only) $T_{amb}$ = 0°C to +70°C, $V_{CC}$ = 5V ±5%

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYPICAL[1] | MAX | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$[7] | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$[7] | | 0 | | $0.2V_{CC}$–0.3 | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST[7] | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST[7] | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3[11] | $I_{OL}$ = 1.6mA[2] | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$[11] | $I_{OL}$ = 3.2mA[2] | | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3, ALE, $\overline{PSEN}$[3] | $I_{OH}$ = –60µA,<br>$I_{OH}$ = –25µA<br>$I_{OH}$ = –10µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$ | | | V<br>V<br>V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode) | $I_{OH}$ = –800µA,<br>$I_{OH}$ = –300µA<br>$I_{OH}$ = –80µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$ | | | V<br>V<br>V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3[7] | $V_{IN}$ = 0.45V | | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3[7] | See note 4 | | | –650 | µA |
| $I_{LI}$ | Input leakage current, port 0 | $V_{IN}$ = $V_{IL}$ or $V_{IH}$ | | | ±10 | µA |
| $I_{CC}$ | Power supply current:[7]<br>Active mode @ 12MHz[8] (Philips)<br>Active mode @ 12MHz[5] (Philips North America)<br>Idle mode @ 12MHz[9] (Philips)<br>Idle mode @ 12MHz (Philips North America)<br>Power-down mode[10] (Philips and Philips North America) | See note 6 | | <br><br>11.5<br><br>1.3<br>3 | <br>18<br>19<br>4.4<br>4<br>50 | <br>mA<br>mA<br>mA<br>mA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor<br>(Philips North America)<br>(Philips) | | <br>50<br>50 | | <br>300<br>150 | <br>kΩ<br>kΩ |
| $C_{IO}$ | Pin capacitance[12] | | | | 10 | pF |

**NOTES:**

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
5. $I_{CCMAX}$ at other frequencies (for Philips North America parts) is given by: Active mode: $I_{CCMAX}$ = 1.43 X FREQ + 1.90;
   Idle mode: $I_{CCMAX}$ = 0.14 X FREQ +2.31, where FREQ is the external oscillator frequency in MHz. $I_{CCMAX}$ is given in mA. See Figure 8.
6. See Figures 9 through 12 for $I_{CC}$ test conditions.
7. For Philips North America parts when $T_{amb}$ = –40°C to +85°C or Philips parts when $T_{amb}$ = –40°C to +125°C, see DC Electrical Characteristics table on previous page.
8. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 10ns; $V_{IL} = V_{SS}$ + 0.5V; $V_{IH} = V_{CC}$ – 0.5V; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = $V_{CC}$.
9. The idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 10ns; $V_{IL} = V_{SS}$ + 0.5V; $V_{IH} = V_{CC}$ – 0.5V; XTAL2 not connected; $\overline{EA}$ = Port 0 = $V_{CC}$; RST = $V_{SS}$.
10. The power-down current is measured with all output pins disconnected, XTAL2 not connected, $\overline{EA}$ = Port 0 = $V_{CC}$; RST = $V_{SS}$.
11. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
    Maximum $I_{OL}$ per port pin: 15mA
    Maximum $I_{OL}$ per 8-bit port: 26mA
    Maximum $I_{OL}$ total for all outputs: 67mA
    If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
12. Pin capacitance for the ceramic DIP package is 15pF maximum.

## DC ELECTRICAL CHARACTERISTICS FOR PHILIPS NORTH AMERICA DEVICES (SC80C31 AND SC80C51)

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%; $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $V_{IL}$ | Input low voltage | 4.5V < $V_{CC}$ < 5.5V | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IH}$ | Input high voltage (ports 0, 1, 2, 3, $\overline{EA}$) | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3[8] | $V_{CC}$ = 4.5V<br>$I_{OL}$ = 1.6mA[2] | | | 0.4 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$[8, 7] | $V_{CC}$ = 4.5V<br>$I_{OL}$ = 3.2mA[2] | | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3 [3] | $V_{CC}$ = 4.5V<br>$I_{OH}$ = –30µA | $V_{CC}$ – 0.7 | | | V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode), ALE[9], $\overline{PSEN}$[3] | $V_{CC}$ = 4.5V<br>$I_{OH}$ = –3.2mA | $V_{CC}$ – 0.7 | | | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3 | $V_{IN}$ = 0.4V | –1 | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3[6] | $V_{IN}$ = 2.0V<br>See note 4 | | | –650 | µA |
| $I_{LI}$ | Input leakage current, port 0 | 0.45 < $V_{IN}$ < $V_{CC}$ – 0.3 | | | ±10 | µA |
| $I_{CC}$ | Power supply current (see Figure 8):<br>Active mode @ 16MHz[5]<br>Idle mode @ 16MHz[5]<br>Power-down mode | See note 5<br><br><br>$T_{amb}$ = 0 to +70°C<br>$T_{amb}$ = –40 to +85°C | | <br>11.5<br>1.3<br>3 | <br>32<br>5<br>50<br>75 | <br>µA<br>µA<br>µA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | 40 | | 225 | kΩ |
| $C_{IO}$ | Pin capacitance[10] (except $\overline{EA}$) | | | | 15 | pF |

**NOTES:**

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the ($V_{CC}$–0.7) specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
5. See Figures 9 through 12 for $I_{CC}$ test conditions.
   Active Mode: $I_{CC}$ = 1.5 × FREQ + 8.0;
   Idle Mode: $I_{CC}$ = 0.14 × FREQ +2.31; See Figure 8.
6. This value applies to $T_{amb}$ = 0°C to +70°C. For $T_{amb}$ = –40°C to +85°C, $I_{TL}$ = –750µA.
7. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
8. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 15mA (*NOTE: This is 85°C specification.)
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 71mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
9. ALE is tested to $V_{OH1}$, except when ALE is off then $V_{OH}$ is the voltage specification.
10. Pin capacitance is characterized but not tested. Pin capacitance is less than 25pF. Pin capacitance of ceramic package is less than 15pF (except $\overline{EA}$ it is 25pF).

## AC ELECTRICAL CHARACTERISTICS FOR SC87C51 12–33MHz PHILIPS NORTH AMERICA DEVICES

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V (SC87C51 12, 16 and 24MHz versions);
For SC87C51 (33MHz only) $T_{amb}$ = = 0°C to +70°C, $V_{CC}$ = 5V ±5%

| SYMBOL | FIGURE | PARAMETER | VARIABLE CLOCK[3] MIN | VARIABLE CLOCK[3] MAX | UNIT |
|---|---|---|---|---|---|
| $1/t_{CLCL}$ | | Oscillator frequency: **Speed Versions**<br>SC87C51 C<br>G<br>P<br>Y | <br>3.5<br>3.5<br>3.5<br>3.5 | <br>12<br>16<br>24<br>33 | <br>MHz<br>MHz<br>MHz<br>MHz |
| $t_{LHLL}$ | 1 | ALE pulse width | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 1 | Address valid to ALE low | $t_{CLCL}-13$ | | ns |
| $t_{LLAX}$ | 1 | Address hold after ALE low | $t_{CLCL}-20$ | | ns |
| $t_{LLIV}$ | 1 | ALE low to valid instruction in | | $4t_{CLCL}-65$ | ns |
| $t_{LLPL}$ | 1 | ALE low to $\overline{PSEN}$ low | $t_{CLCL}-13$ | | ns |
| $t_{PLPH}$ | 1 | $\overline{PSEN}$ pulse width | $3t_{CLCL}-20$ | | ns |
| $t_{PLIV}$ | 1 | $\overline{PSEN}$ low to valid instruction in | | $3t_{CLCL}-45$ | ns |
| $t_{PXIX}$ | 1 | Input instruction hold after $\overline{PSEN}$ | 0 | | ns |
| $t_{PXIZ}$ | 1 | Input instruction float after $\overline{PSEN}$ | | $t_{CLCL}-10$ | ns |
| $t_{AVIV}$ | 1 | Address to valid instruction in | | $5t_{CLCL}-55$ | ns |
| $t_{PLAZ}$ | 1 | $\overline{PSEN}$ low to address float | | 10 | ns |
| **Data Memory** | | | | | |
| $t_{RLRH}$ | 2, 3 | $\overline{RD}$ pulse width | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 2, 3 | $\overline{WR}$ pulse width | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 2, 3 | $\overline{RD}$ low to valid data in | | $5t_{CLCL}-90$ | ns |
| $t_{RHDX}$ | 2, 3 | Data hold after $\overline{RD}$ | 0 | | ns |
| $t_{RHDZ}$ | 2, 3 | Data float after $\overline{RD}$ | | $2t_{CLCL}-28$ | ns |
| $t_{LLDV}$ | 2, 3 | ALE low to valid data in | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 2, 3 | Address to valid data in | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 2, 3 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 2, 3 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | $4t_{CLCL}-75$ | | ns |
| $t_{QVWX}$ | 2, 3 | Data valid to $\overline{WR}$ transition | $t_{CLCL}-20$ | | ns |
| $t_{WHQX}$ | 2, 3 | Data hold after $\overline{WR}$ | $t_{CLCL}-20$ | | ns |
| $t_{RLAZ}$ | 2, 3 | $\overline{RD}$ low to address float | | 0 | ns |
| $t_{WHLH}$ | 2, 3 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | $t_{CLCL}-20$ | $t_{CLCL}+25$ | ns |
| **External Clock** | | | | | |
| $t_{CHCX}$ | 5 | High time | 12 | | ns |
| $t_{CLCX}$ | 5 | Low time | 12 | | ns |
| $t_{CLCH}$ | 5 | Rise time | | 20 | ns |
| $t_{CHCL}$ | 5 | Fall time | | 20 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. For all Philips North America speed versions only.
4. Interfacing the 87C51 to devices with float times up to 50ns is permitted. This limited bus contention will not cause damage to port 0 drivers.

## AC ELECTRICAL CHARACTERISTICS FOR PHILIPS DEVICES

$T_{amb}$ = 0°C to +70°C, $V_{CC}$ = 5V ±20%, $V_{SS}$ = 0V (PCB80C31/51, PCF80C31/51)[1, 2, 4, 5]

| SYMBOL | FIGURE | PARAMETER | VARIABLE CLOCK[3] | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $1/t_{CLCL}$ | | Oscillator frequency: **Speed Versions**<br>PCB8031/51 –2<br>PCA/PCB/PCF80C31/51 –3<br>PCB/PCF80C31/51 –4<br>PCB/FB80C31/51 –5 | <br>0.5<br>1.2<br>1.2<br>1.2 | <br>12<br>16<br>24<br>33 | <br>MHz<br>MHz<br>MHz<br>MHz |
| $t_{LHLL}$ | 1 | ALE pulse width | $2t_{CLCL}$–40 | | ns |
| $t_{AVLL}$ | 1 | Address valid to ALE low | $t_{CLCL}$–25 | | ns |
| $t_{LLAX}$ | 1 | Address hold after ALE low | $t_{CLCL}$–25 | | ns |
| $t_{LLIV}$ | 1 | ALE low to valid instruction in | | $4t_{CLCL}$–65 | ns |
| $t_{LLPL}$ | 1 | ALE low to $\overline{PSEN}$ low | $t_{CLCL}$–25 | | ns |
| $t_{PLPH}$ | 1 | $\overline{PSEN}$ pulse width | $3t_{CLCL}$–45 | | ns |
| $t_{PLIV}$ | 1 | $\overline{PSEN}$ low to valid instruction in | | $3t_{CLCL}$–60 | ns |
| $t_{PXIX}$ | 1 | Input instruction hold after $\overline{PSEN}$ | 0 | | ns |
| $t_{PXIZ}$ | 1 | Input instruction float after $\overline{PSEN}$ | | $t_{CLCL}$–25 | ns |
| $t_{AVIV}$ | 1 | Address to valid instruction in | | $5t_{CLCL}$–80 | ns |
| $t_{PLAZ}$ | 1 | $\overline{PSEN}$ low to address float | | 10 | ns |
| **Data Memory** | | | | | |
| $t_{RLRH}$ | 2, 3 | $\overline{RD}$ pulse width | $6t_{CLCL}$–100 | | ns |
| $t_{WLWH}$ | 2, 3 | $\overline{WR}$ pulse width | $6t_{CLCL}$–100 | | ns |
| $t_{RLDV}$ | 2, 3 | $\overline{RD}$ low to valid data in | | $5t_{CLCL}$–90 | ns |
| $t_{RHDX}$ | 2, 3 | Data hold after $\overline{RD}$ | 0 | | ns |
| $t_{RHDZ}$ | 2, 3 | Data float after $\overline{RD}$ | | $2t_{CLCL}$–28 | ns |
| $t_{LLDV}$ | 2, 3 | ALE low to valid data in | | $8t_{CLCL}$–150 | ns |
| $t_{AVDV}$ | 2, 3 | Address to valid data in | | $9t_{CLCL}$–165 | ns |
| $t_{LLWL}$ | 2, 3 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | $3t_{CLCL}$–50 | $3t_{CLCL}$+50 | ns |
| $t_{AVWL}$ | 2, 3 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | $4t_{CLCL}$–75 | | ns |
| $t_{QVWX}$ | 2, 3 | Data valid to $\overline{WR}$ transition | $t_{CLCL}$–30 | | ns |
| $t_{WHQX}$ | 2, 3 | Data hold after $\overline{WR}$ | $t_{CLCL}$–25 | | ns |
| $t_{RLAZ}$ | 2, 3 | $\overline{RD}$ low to address float | | 0 | ns |
| $t_{WHLH}$ | 2, 3 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | $t_{CLCL}$–25 | $t_{CLCL}$+25 | ns |
| **External Clock** | | | | | |
| $t_{CHCX}$ | 5 | High time | 15 | | ns |
| $t_{CLCX}$ | 5 | Low time | 15 | | ns |
| $t_{CLCH}$ | 5 | Rise time | | 20 | ns |
| $t_{CHCL}$ | 5 | Fall time | | 20 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. For all Philips speed versions only.
4. Interfacing the 80C31/51 to devices with float times up to 30ns is permitted. This limited bus contention will not cause damage to port 0 drivers.
5. $V_{CC}$ = 5V ±10% for 33MHz.

## AC ELECTRICAL CHARACTERISTICS FOR PHILIPS NORTH AMERICA DEVICES (SC80C31 AND SC80C51)

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 1 | Oscillator frequency<br>Speed versions : C, G | | | 3.5 | 16 | MHz |
| $t_{LHLL}$ | 1 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 1 | Address valid to ALE low | 22 | | $t_{CLCL}-40$ | | ns |
| $t_{LLAX}$ | 1 | Address hold after ALE low | 32 | | $t_{CLCL}-30$ | | ns |
| $t_{LLIV}$ | 1 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 1 | ALE low to $\overline{PSEN}$ low | 32 | | $t_{CLCL}-30$ | | ns |
| $t_{PLPH}$ | 1 | $\overline{PSEN}$ pulse width | 142 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 1 | $\overline{PSEN}$ low to valid instruction in[4] | | 82 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 1 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 1 | Input instruction float after $\overline{PSEN}$ | | 37 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 1 | Address to valid instruction in[4] | | 207 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 1 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 2, 3 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 2, 3 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 2, 3 | $\overline{RD}$ low to valid data in | | 147 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 2, 3 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 2, 3 | Data float after $\overline{RD}$ | | 65 | | $2t_{CLCL}-60$ | ns |
| $t_{LLDV}$ | 2, 3 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 2, 3 | Address to valid data in | | 397 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 2, 3 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 137 | 239 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 2, 3 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 122 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 2, 3 | Data valid to $\overline{WR}$ transition | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{WHQX}$ | 2, 3 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{QVWH}$ | 3 | Data valid to $\overline{WR}$ high | 287 | | $7t_{CLCL}-150$ | | ns |
| $t_{RLAZ}$ | 2, 3 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 2, 3 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 5 | High time | 20 | | 20 | $t_{CLCL}-t_{CLCX}$ | ns |
| $t_{CLCX}$ | 5 | Low time | 20 | | 20 | $t_{CLCL}-t_{CHCX}$ | ns |
| $t_{CLCH}$ | 5 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 5 | Fall time | | 20 | | 20 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 4 | Serial port clock cycle time | 750 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 4 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 4 | Output data hold after clock rising edge | 8 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 4 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 4 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}-133$ | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 80C31/51 to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.
4. See application note AN457 for external memory interfacing.

## AC ELECTRICAL CHARACTERISTICS FOR PHILIPS NORTH AMERICA DEVICES (SC80C31 AND SC80C51)

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 24MHz CLOCK | | VARIABLE CLOCK[4] | | 33MHz CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 1 | Oscillator frequency<br>Speed versions : P (24MHz)<br>: Y (33MHz) | 3.5 | 24 | 3.5 | 33 | <br>3.5 | <br>33 | MHz |
| $t_{LHLL}$ | 1 | ALE pulse width | 43 | | $2t_{CLCL}$–40 | | 21 | | ns |
| $t_{AVLL}$ | 1 | Address valid to ALE low | 17 | | $t_{CLCL}$–25 | | 5 | | ns |
| $t_{LLAX}$ | 1 | Address hold after ALE low | 17 | | $t_{CLCL}$–25 | | | | ns |
| $t_{LLIV}$ | 1 | ALE low to valid instruction in | | 102 | | $4t_{CLCL}$–65 | | 55 | ns |
| $t_{LLPL}$ | 1 | ALE low to $\overline{PSEN}$ low | 17 | | $t_{CLCL}$–25 | | 5 | | ns |
| $t_{PLPH}$ | 1 | $\overline{PSEN}$ pulse width | 80 | | $3t_{CLCL}$–45 | | 45 | | ns |
| $t_{PLIV}$ | 1 | $\overline{PSEN}$ low to valid instruction in | | 65 | | $3t_{CLCL}$–60 | | 30 | ns |
| $t_{PXIX}$ | 1 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 1 | Input instruction float after $\overline{PSEN}$ | | 17 | | $t_{CLCL}$–25 | | 5 | ns |
| $t_{AVIV}$ | 1 | Address to valid instruction in | | 128 | | $5t_{CLCL}$–80 | | 70 | ns |
| $t_{PLAZ}$ | 1 | $\overline{PSEN}$ low to address float | | 10 | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | | | |
| $t_{RLRH}$ | 2, 3 | $\overline{RD}$ pulse width | 150 | | $6t_{CLCL}$–100 | | 82 | | ns |
| $t_{WLWH}$ | 2, 3 | $\overline{WR}$ pulse width | 150 | | $6t_{CLCL}$–100 | | 82 | | ns |
| $t_{RLDV}$ | 2, 3 | $\overline{RD}$ low to valid data in | | 118 | | $5t_{CLCL}$–90 | | 60 | ns |
| $t_{RHDX}$ | 2, 3 | Data hold after $\overline{RD}$ | 0 | | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 2, 3 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}$–28 | | 32 | ns |
| $t_{LLDV}$ | 2, 3 | ALE low to valid data in | | 183 | | $8t_{CLCL}$–150 | | 90 | ns |
| $t_{AVDV}$ | 2, 3 | Address to valid data in | | 210 | | $9t_{CLCL}$–165 | | 105 | ns |
| $t_{LLWL}$ | 2, 3 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 75 | 175 | $3t_{CLCL}$–50 | $3t_{CLCL}$+50 | 40 | 140 | ns |
| $t_{AVWL}$ | 2, 3 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 92 | | $4t_{CLCL}$–75 | | 45 | | ns |
| $t_{QVWX}$ | 2, 3 | Data valid to $\overline{WR}$ transition | 12 | | $t_{CLCL}$–30 | | 0 | | ns |
| $t_{WHQX}$ | 2, 3 | Data hold after $\overline{WR}$ | 17 | | $t_{CLCL}$–25 | | 5 | | ns |
| $t_{QVWH}$ | 3 | Data valid to $\overline{WR}$ high | 162 | | $7t_{CLCL}$–130 | | 80 | | ns |
| $t_{RLAZ}$ | 2, 3 | $\overline{RD}$ low to address float | | 0 | | 0 | | 0 | ns |
| $t_{WHLH}$ | 2, 3 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 17 | 67 | $t_{CLCL}$–25 | $t_{CLCL}$+25 | 5 | 55 | ns |
| **External Clock** | | | | | | | | | |
| $t_{CHCX}$ | 5 | High time | 17 | | 17 | $t_{CLCL}$–$t_{CLCX}$ | | | ns |
| $t_{CLCX}$ | 5 | Low time | 17 | | 17 | $t_{CLCL}$–$t_{CHCX}$ | | | ns |
| $t_{CLCH}$ | 5 | Rise time | | 5 | | 5 | | | ns |
| $t_{CHCL}$ | 5 | Fall time | | 5 | | 5 | | | ns |
| **Shift Register** | | | | | | | | | |
| $t_{XLXL}$ | 4 | Serial port clock cycle time | 505 | | $12t_{CLCL}$ | | 360 | | ns |
| $t_{QVXH}$ | 4 | Output data setup to clock rising edge | 283 | | $10t_{CLCL}$–133 | | 167 | | ns |
| $t_{XHQX}$ | 4 | Output data hold after clock rising edge | 3 | | $2t_{CLCL}$–80 | | | | ns |
| $t_{XHDX}$ | 4 | Input data hold after clock rising edge | 0 | | 0 | | 0 | | ns |
| $t_{XHDV}$ | 4 | Clock rising edge to input data valid | | 283 | | $10t_{CLCL}$–133 | | 167 | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the SC80C31/51 to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.
4. Variable clock is specified for oscillator frequencies greater than 16MHz to 33MHz. For frequencies equal or less than 16MHz, see 16MHz "AC Electrial Characteristics", page 3-16.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{PSEN}$ low.

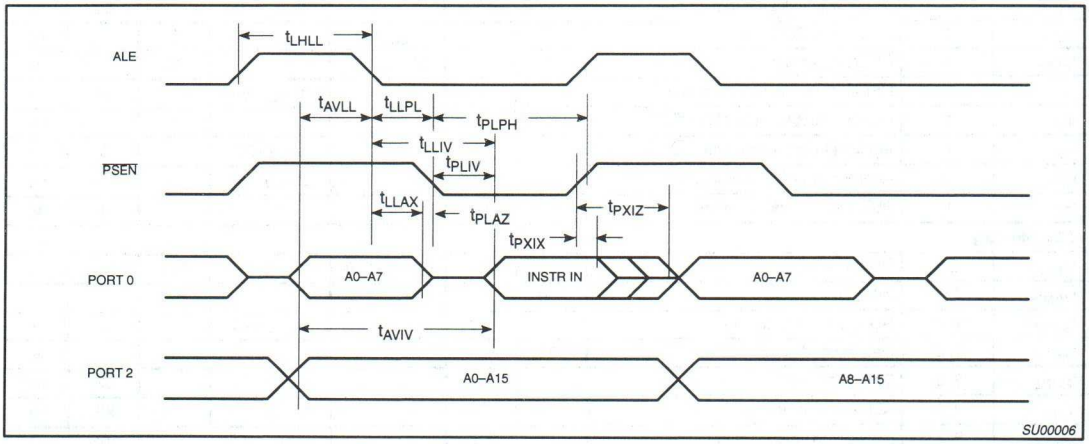



**Figure 1. External Program Memory Read Cycle**

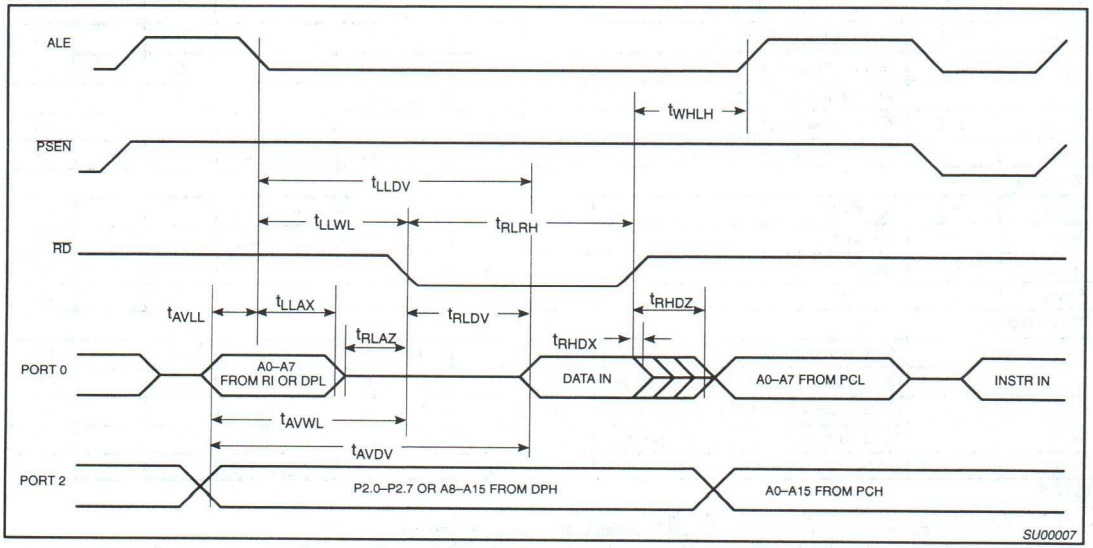



**Figure 2. External Data Memory Read Cycle**

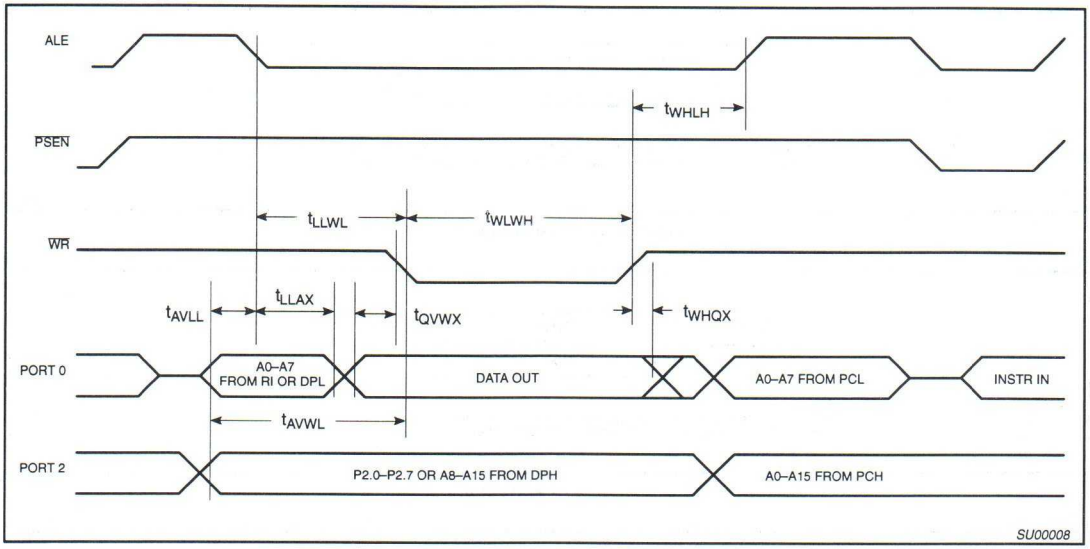



**Figure 3. External Data Memory Write Cycle**

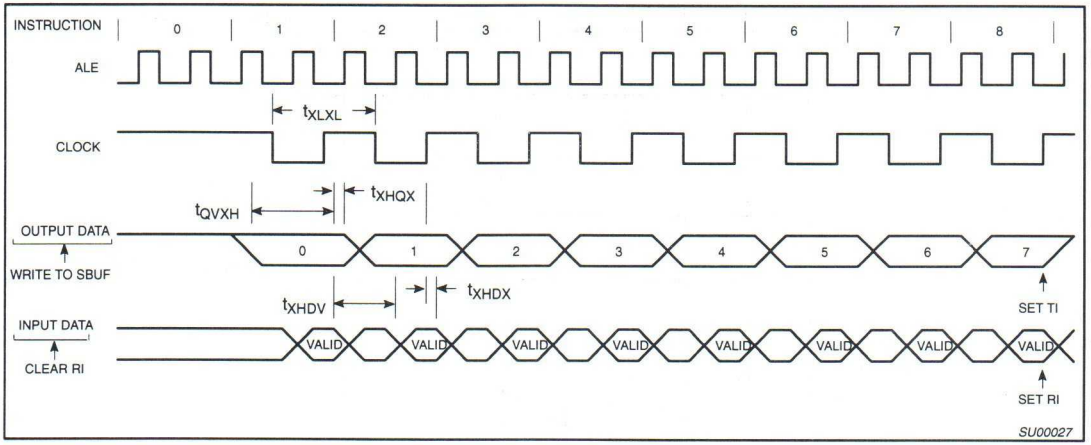



**Figure 4. Shift Register Mode Timing**

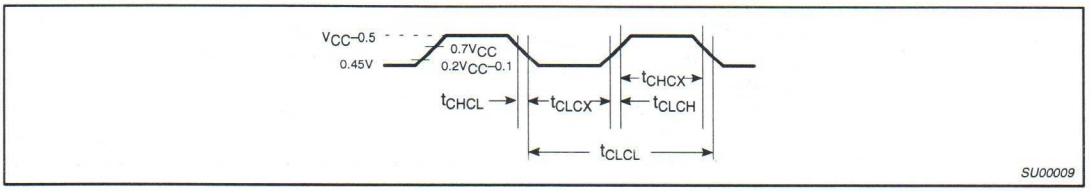
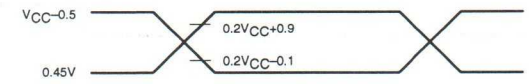


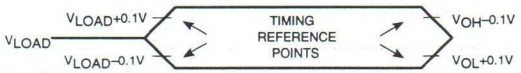

**Figure 5. External Clock Drive**




NOTE:
AC inputs during testing are driven at $V_{CC}$ –0.5 for a logic '1' and 0.45V for a logic '0'.
Timing measurements are made at $V_{IH}$ min for a logic '1' and $V_{IL}$ max for a logic '0'.

SU00010

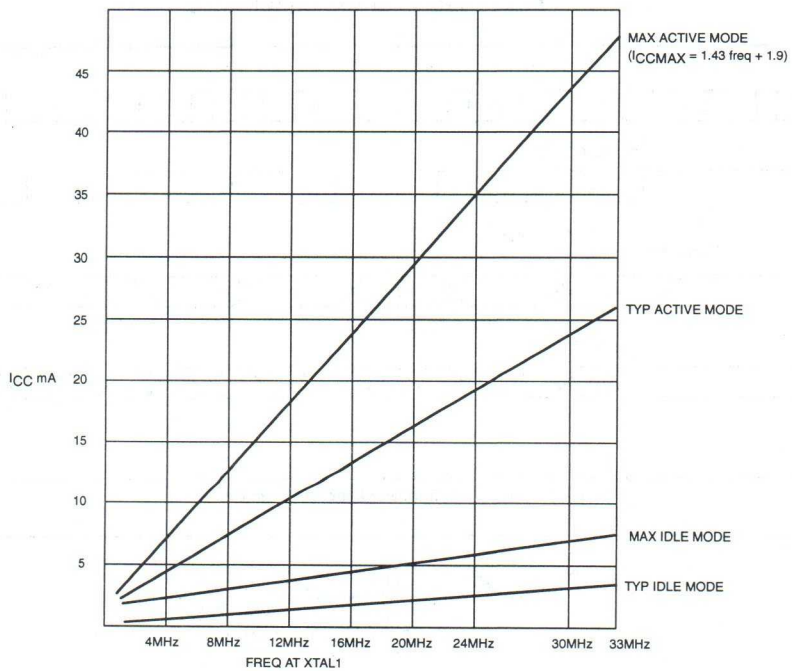
**Figure 6. AC Testing Input/Output**




NOTE:
For timing purposes, a port is no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded $V_{OH}/V_{OL}$ level occurs. $I_{OH}/I_{OL} \geq \pm 20$mA.

SU00011

**Figure 7. Float Waveform**




SU00012

**Figure 8. $I_{CC}$ vs. FREQ**
**Valid only within frequency specifications of the device under test**

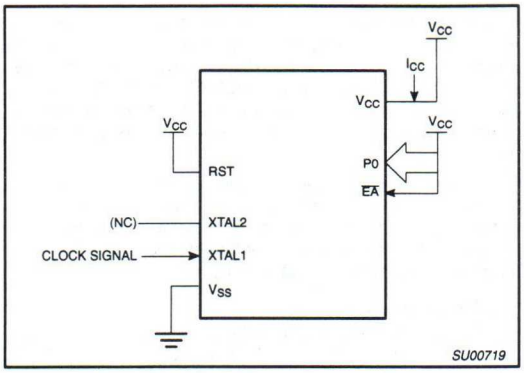



**Figure 9. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

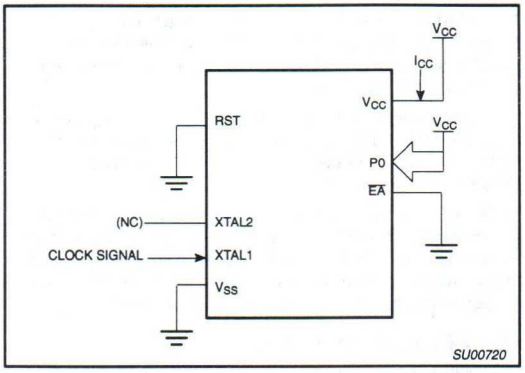



**Figure 10. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

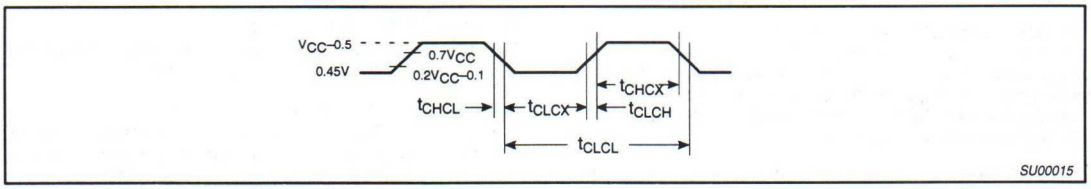



**Figure 11. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

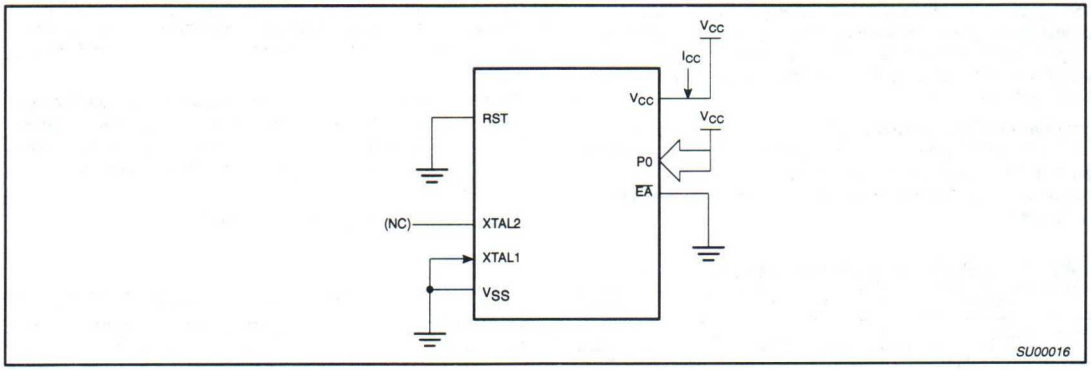



**Figure 12. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2V to 5.5V**

## EPROM CHARACTERISTICS

The 87C51 is programmed by using a modified Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C51 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C51 manufactured by Philips Corporation.

Table 3 shows the logic levels for reading the signature bytes, and for programming the program memory, the encryption table, and the security bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 13 and 14. Figure 15 shows the circuit configuration for normal program memory verification.

### Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 13. Note that the 87C51 is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1 and 2, as shown in Figure 13. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 3 specified in Table 3 are held at the 'Program Code Data' levels indicated in Table 3. The ALE/$\overline{PROG}$ is pulsed low 25 times as shown in Figure 14.

To program the encryption table, repeat the 25 pulse programming sequence for addresses 0 through 1FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the security bits, repeat the 25 pulse programming sequence using the 'Pgm Security Bit' levels. After one security bit is programmed, further programming of the code memory and encryption table is disabled. However, the other security bit can still be programmed.

Note that the $\overline{EA}$/$V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

### Program Verification

If security bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 1 and 2 as shown in Figure 15. The other pins are held at the 'Verify Code Data' levels indicated in Table 3. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

#### Reading the Signature Bytes

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:
(030H) = 15H indicates manufactured by Philips
(031H) = 92H indicates 87C51

### Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 3, and which satisfies the timing specifications, is suitable.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-sec/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

**Table 3. EPROM Programming Modes**

| MODE | RST | $\overline{PSEN}$ | ALE/$\overline{PROG}$ | $\overline{EA}$/$V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm security bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm security bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V±10% during programming and verification.
4. *ALE/$\overline{PROG}$ receives 25 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

™Trademark phrase of Intel Corporation.

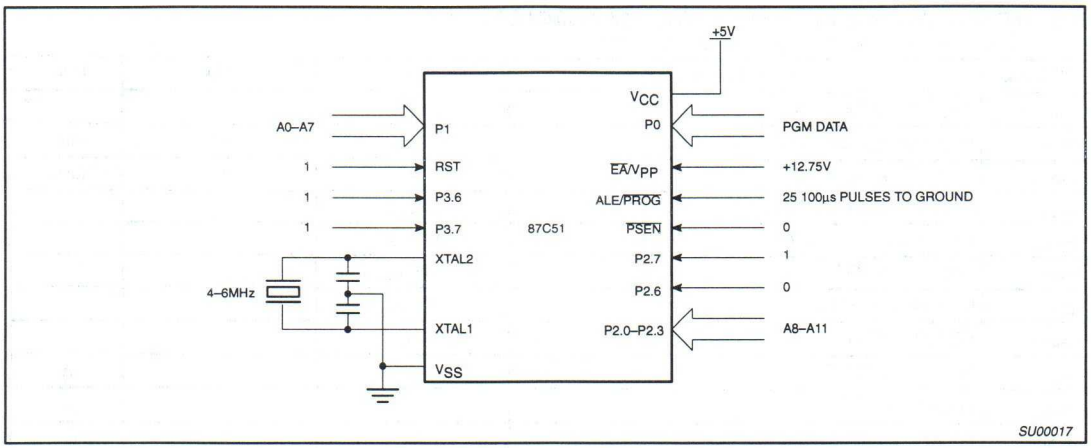



**Figure 13. Programming Configuration**

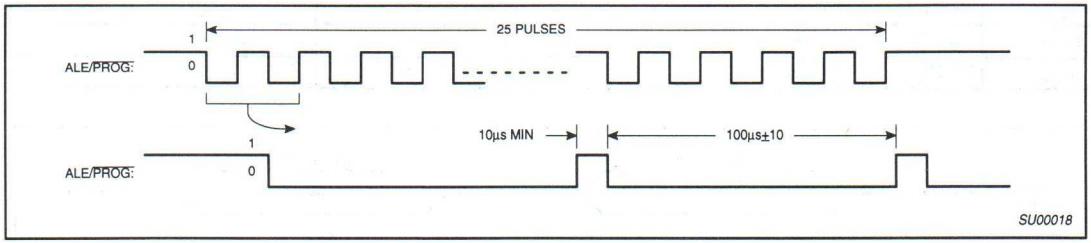



**Figure 14. PROG Waveform**

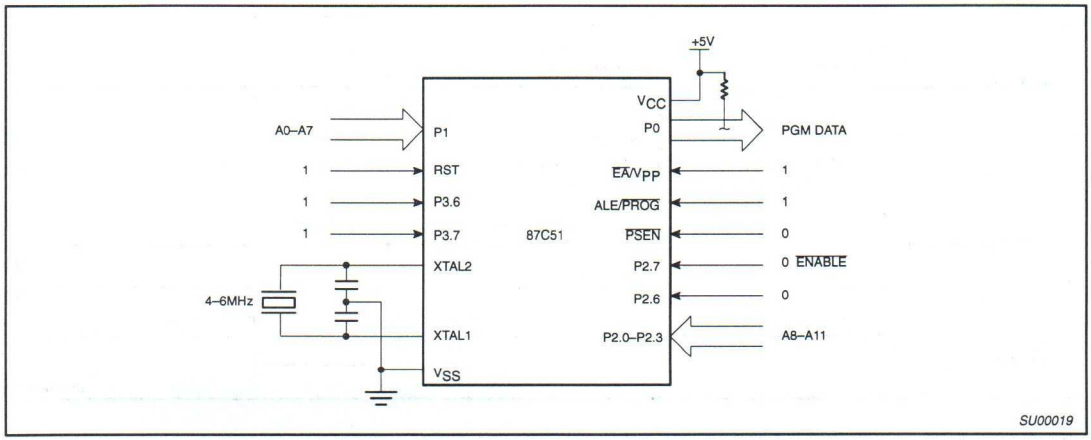
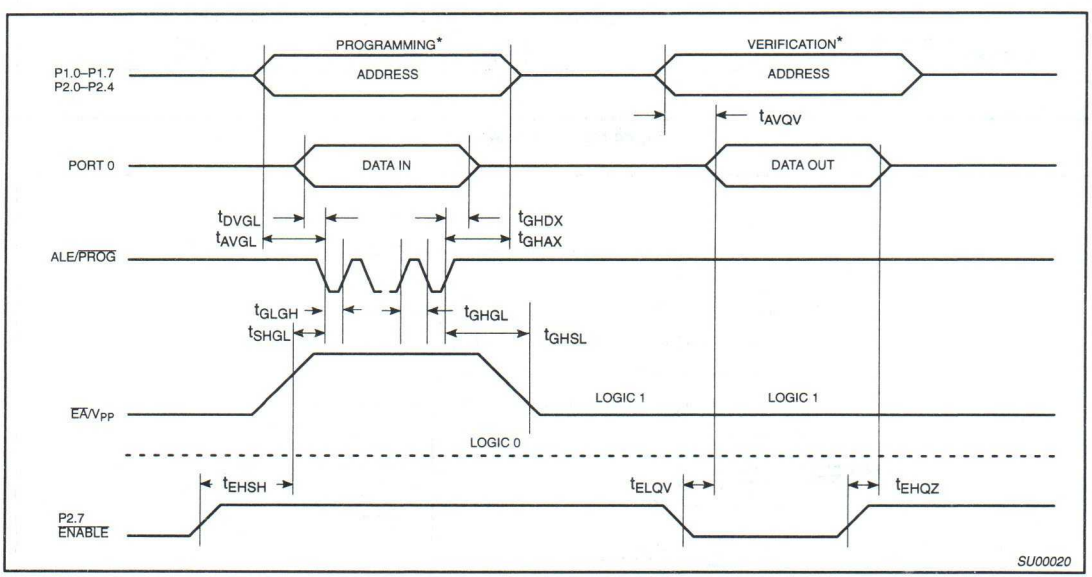



**Figure 15. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (See Figure 16)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




**NOTE:**

* FOR PROGRAMMING VERIFICATION SEE FIGURE 13.
FOR VERIFICATION CONDITIONS SEE FIGURE 15.

**Figure 16. EPROM Programming and Verification**

# CMOS single-chip 8-bit microcontrollers

# 83C51FA/83C51FB/ 83C51FC/80C51FA

## DESCRIPTION

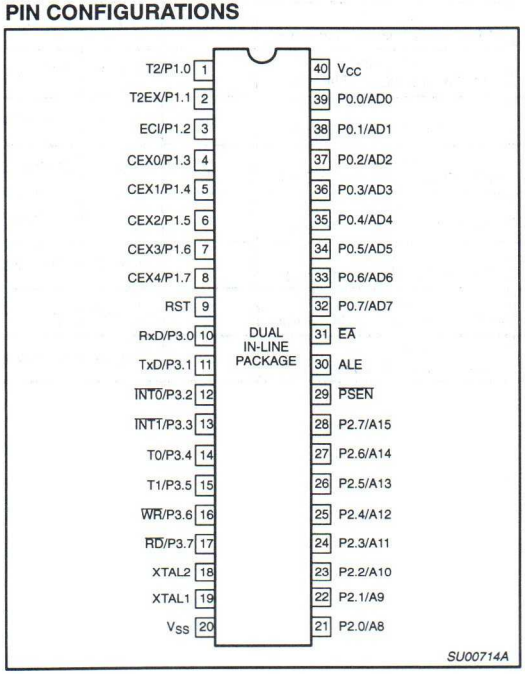
The 83C51FA/83C51FB/80C51FA Single-Chip 8-Bit Microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 83C51FA/83C51FB/80C51FA has the same instruction set as the 80C51.

This device provides architectural enhancements that make it applicable in a variety of applications for general control systems. The 83C51FA contains 8k × 8 ROM memory (80C51FA ROMless version addresses up to 64k of external memory), the 83C51FB contains 16k × 8 ROM memory, the 83C51FC contains 32k × 8 ROM memory, a volatile 256 × 8 read/write data memory, four 8-bit I/O ports, three 16-bit timer/event counters, a Programmable Counter Array (PCA), a multi-source, four-priority-level, nested interrupt structure, an enhanced UART and on-chip oscillator and timing circuits. For systems that require extra capability, the 83C51FA/FB can be expanded using standard TTL compatible memories and logic.

Its added features make it an even more powerful microcontroller for applications that require pulse width modulation, high-speed I/O and up/down counting capabilities such as motor control. It also has a more versatile serial channel that facilitates multiprocessor communications.

See 87C51FA/87C51FB datasheet for EPROM/OTP specifications.

## PIN CONFIGURATIONS




## FEATURES

- 80C51 central processing unit
- Full static operation
- 8k × 8 ROM: 83C51FA;
  16k × 8 ROM: 83C51FB;
  32k × 8 ROM: 83C51FC
  ROMless: 80C51FA
  all capable of addressing external memory to 64k bytes
  - Two level program security system
  - 64 byte encryption array
- 256 × 8 RAM, expandable externally to 64k bytes
- Speed range up to 33MHz
- Three 16-bit timer/counters
  - T2 is an up/down counter
- 7 interrupt sources
- 4 level priority
- Programmable Counter Array (PCA)
  - High speed output
  - Capture/compare
  - Pulse Width Modulator
  - Watchdog Timer
- Four 8-bit I/O ports
- Full-duplex enhanced UART
  - Framing error detection
  - Automatic address recognition
- Power control modes
  - Idle mode
  - Power-down mode
- Once (On Circuit Emulation) Mode
- Five package styles
- Programmable clock out
- Low EMI (inhibit ALE)
- Second DPTR register (ROM only)
- Asynchronous port Reset

## ORDERING INFORMATION

| ROMless | ROM 8K × 8 | ROM 16K × 8 | ROM 32K × 8 | TEMPERATURE RANGE °C AND PACKAGE | VOLTAGE RANGE (V)[1] | FREQ. (MHZ) | DWG. # |
|---|---|---|---|---|---|---|---|
| S80C51FA–4N40 | S83C51FA–4N40 | S83C51FB–4N40 | S83C51FC–4N40 | 0 to +70, 40-Pin Plastic Dual In-line Pkg. | 2.7 to 5.5[1] | 3.5 to 16 | SOT129-1 |
| S80C51FA–4A44 | S83C51FA–4A44 | S83C51FB–4A44 | S83C51FC–4A44 | 0 to +70, 44-Pin Plastic Leaded Chip Carrier | 2.7 to 5.5[1] | 3.5 to 16 | SOT187-2 |
| S80C51FA–4B44 | S83C51FA–4B44 | S83C51FB–4B44 | S83C51FC–4B44 | 0 to +70, 44-Pin Plastic Quad Flat Pack | 2.7 to 5.5[1] | 3.5 to 16 | SOT307-2 |
| S80C51FA–5N40 | S83C51FA–5N40 | S83C51FB–5N40 | S83C51FC–5N40 | –40 to +85, 40-Pin Plastic Dual In-line Pkg. | 2.7 to 5.5[1] | 3.5 to 16 | SOT129-1 |
| S80C51FA–5A44 | S83C51FA–5A44 | S83C51FB–5A44 | S83C51FC–5A44 | –40 to +85, 44-Pin Plastic Leaded Chip Carrier | 2.7 to 5.5[1] | 3.5 to 16 | SOT187-2 |
| S80C51FA–5B44 | S83C51FA–5B44 | S83C51FB–5B44 | S83C51FC–5B44 | –40 to +85, 44-Pin Plastic Quad Flat Pack | 2.7 to 5.5[1] | 3.5 to 16 | SOT307-2 |
| S80C51FA–AN40 | S83C51FA–AN40 | S83C51FB–AN40 | S83C51FC–AN40 | 0 to +70, 40-Pin Plastic Dual In-line Pkg. | 5 | 3.5 to 24 | SOT129-1 |
| S80C51FA–AA44 | S83C51FA–AA44 | S83C51FB–AA44 | S83C51FC–AA44 | 0 to +70, 44-Pin Plastic Leaded Chip Carrier | 5 | 3.5 to 24 | SOT187-2 |
| S80C51FA–AB44 | S83C51FA–AB44 | S83C51FB–AB44 | S83C51FC–AB44 | 0 to +70, 44-Pin Plastic Quad Flat Pack | 5 | 3.5 to 24 | SOT307-2 |
| S80C51FA–BN40 | S83C51FA–BN40 | S83C51FB–BN40 | S83C51FC–BN40 | –40 to +85, 40-Pin Plastic Dual In-line Pkg. | 5 | 3.5 to 24 | SOT129-1 |
| S80C51FA–BA44 | S83C51FA–BA44 | S83C51FB–BA44 | S83C51FC–BA44 | –40 to +85, 44-Pin Plastic Leaded Chip Carrier | 5 | 3.5 to 24 | SOT187-2 |
| S80C51FA–BB44 | S83C51FA–BB44 | S83C51FB–BB44 | S83C51FC–BB44 | –40 to +85, 44-Pin Plastic Quad Flat Pack | 5 | 3.5 to 24 | SOT307-2 |
| S80C51FA–IN40 | S83C51FA–IN40 | S83C51FB–IN40 | S83C51FC–IN40 | 0 to +70, 40-Pin Plastic Dual In-line Pkg. | 5 | 3.5 to 33 | SOT129-1 |
| S80C51FA–IA44 | S83C51FA–IA44 | S83C51FB–IA44 | S83C51FC–IA44 | 0 to +70, 44-Pin Plastic Leaded Chip Carrier | 5 | 3.5 to 33 | SOT187-2 |
| S80C51FA–IB44 | S83C51FA–IB44 | S83C51FB–IB44 | S83C51FC–IB44 | 0 to +70, 44-Pin Plastic Quad Flat Pack | 5 | 3.5 to 33 | SOT307-2 |
| S80C51FA–JN40 | S83C51FA–JN40 | S83C51FB–JN40 | S83C51FC–JN40 | –40 to +85, 40-Pin Plastic Dual In-line Pkg. | 5 | 3.5 to 33 | SOT129-1 |
| S80C51FA–JA44 | S83C51FA–JA44 | S83C51FB–JA44 | S83C51FC–JA44 | –40 to +85, 44-Pin Plastic Leaded Chip Carrier | 5 | 3.5 to 33 | SOT187-2 |
| S80C51FA–JB44 | S83C51FA–JB44 | S83C51FB–JB44 | S83C51FC–JB44 | –40 to +85, 44-Pin Plastic Quad Flat Pack | 5 | 3.5 to 33 | SOT307-2 |

**NOTE:**
1. S80C51FA devices are specified for 5V only.

## BLOCK DIAGRAM

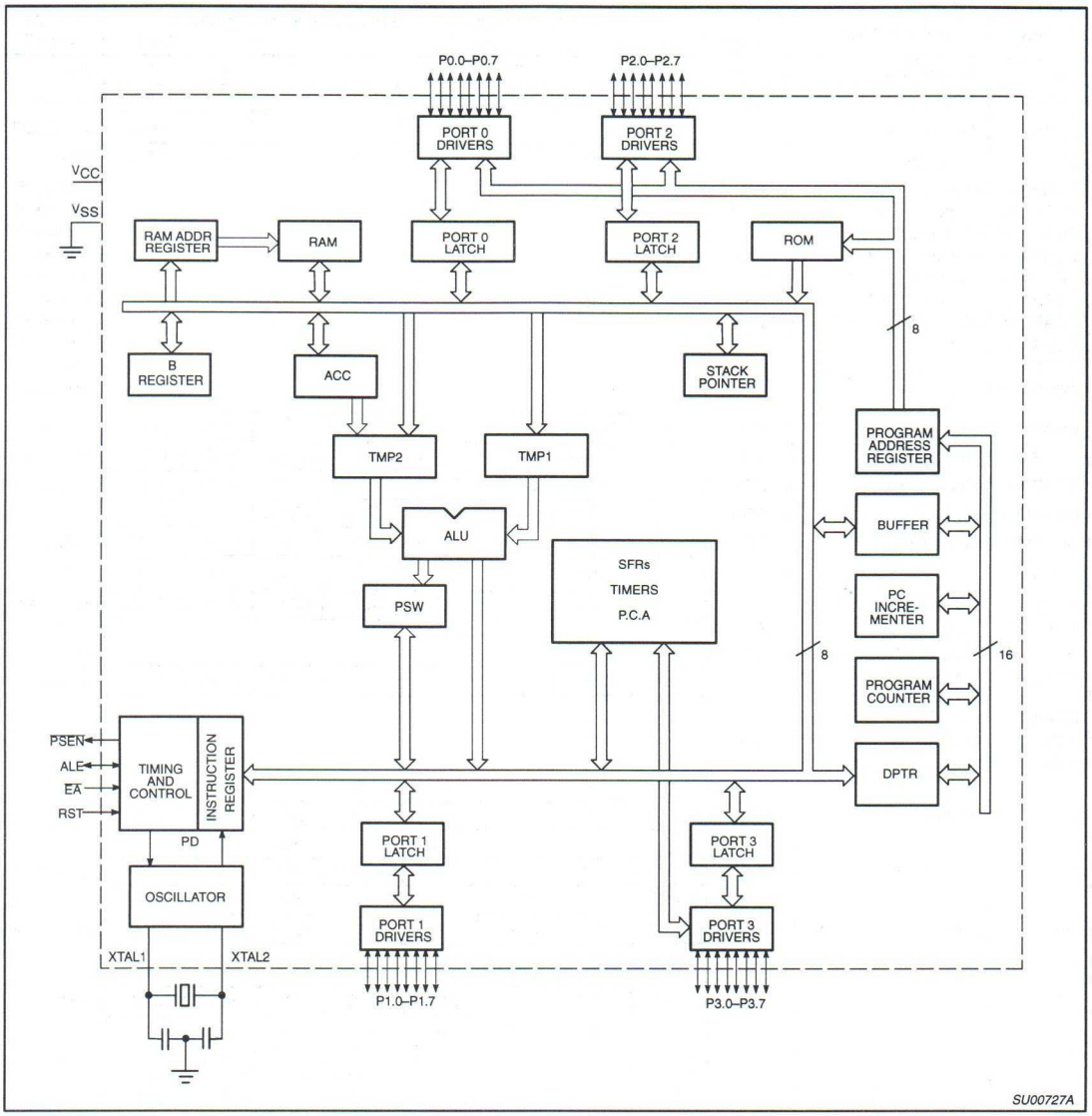


**Table 1. 83C51FA/83C51FB/83C51FC/80C51FA Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| AUXR# | Auxiliary | 8EH | – | – | – | – | – | – | – | AO | xxxxxxx0B |
| AUXR1# | Auxiliary 1 [3] | A2H | – | – | – | – | – | – | – | DPS0 | xxxx0xx0B |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| CCAP0H# | Module 0 Capture High | FAH | | | | | | | | | xxxxxxxxB |
| CCAP1H# | Module 1 Capture High | FBH | | | | | | | | | xxxxxxxxB |
| CCAP2H# | Module 2 Capture High | FCH | | | | | | | | | xxxxxxxxB |
| CCAP3H# | Module 3 Capture High | FDH | | | | | | | | | xxxxxxxxB |
| CCAP4H# | Module 4 Capture High | FEH | | | | | | | | | xxxxxxxxB |
| CCAP0L# | Module 0 Capture Low | EAH | | | | | | | | | xxxxxxxxB |
| CCAP1L# | Module 1 Capture Low | EBH | | | | | | | | | xxxxxxxxB |
| CCAP2L# | Module 2 Capture Low | ECH | | | | | | | | | xxxxxxxxB |
| CCAP3L# | Module 3 Capture Low | EDH | | | | | | | | | xxxxxxxxB |
| CCAP4L# | Module 4 Capture Low | EEH | | | | | | | | | xxxxxxxxB |
| CCAPM0# | Module 0 Mode | DAH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM1# | Module 1 Mode | DBH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM2# | Module 2 Mode | DCH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM3# | Module 3 Mode | DDH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM4# | Module 4 Mode | DEH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| CCON*# | PCA Counter Control | D8H | CF | CR | – | CCF4 | CCF3 | CCF2 | CCF1 | CCF0 | 00x00000B |
| CH# | PCA Counter High | F9H | | | | | | | | | 00H |
| CL# | PCA Counter Low | E9H | | | | | | | | | 00H |
| CMOD# | PCA Counter Mode | D9H | CIDL | WDTE | – | – | – | CPS1 | CPS0 | ECF | 00xxx000B |
| DPTR: | Data Pointer (2 bytes) | | | | | | | | | | |
| DPH | Data Pointer High | 83H | | | | | | | | | 00H |
| DPL | Data Pointer Low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE* | Interrupt Enable | A8H | EA | EC | ET2 | ES | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP* | Interrupt Priority | B8H | – | PPC | PT2 | PS | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| IPH# | Interrupt Priority High | B7H | – | PPCH | PT2H | PSH | PT1H | PX1H | PT0H | PX0H | x0000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | CEX4 | CEX3 | CEX2 | CEX1 | CEX0 | ECI | T2EX | T2 | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | AD15 | AD14 | AD13 | AD12 | AD11 | AD10 | AD9 | AD8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TxD | RxD | FFH |
| PCON# | Power Control | 87H | SMOD1 | SMOD0 | – | POF[2] | GF1 | GF0 | PD | IDL | 00xx0000B |

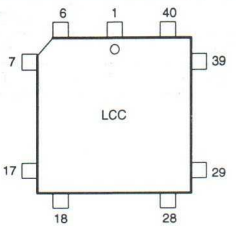
* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
– Reserved bits.
1. Reset value depends on reset source.
2. Bit will not be affected by Reset.
3. Not available on 80C51FA (ROMless) at this time.

**Table 1. 83C51FA/83C51FB/83C51FC/80C51FA Special Function Registers** (Continued)

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program Status Word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| RACAP2H# | Timer 2 Capture High | CBH | | | | | | | | | 00H |
| RACAP2L# | Timer 2 Capture Low | CAH | | | | | | | | | 00H |
| SADDR# | Slave Address | A9H | | | | | | | | | 00H |
| SADEN# | Slave Address Mask | B9H | | | | | | | | | 00H |
| SBUF | Serial Data Buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial Control | 98H | SM0/FE | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack Pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer Control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| T2CON* | Timer 2 Control | C8H | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/$\overline{T2}$ | CP/$\overline{RL2}$ | 00H |
| T2MOD# | Timer 2 Mode Control | C9H | – | – | – | – | – | – | T2OE | DCEN | xxxxxx00B |
| TH0 | Timer High 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer High 1 | 8DH | | | | | | | | | 00H |
| TH2# | Timer High 2 | CDH | | | | | | | | | 00H |
| TL0 | Timer Low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer Low 1 | 8BH | | | | | | | | | 00H |
| TL2# | Timer Low 2 | CCH | | | | | | | | | 00H |
| TMOD | Timer Mode | 89H | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
– Reserved bits.

## PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

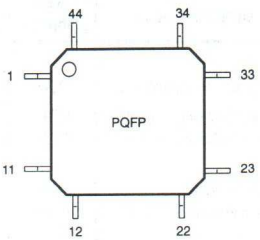



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | NC* | 16 | P3.4/T0 | 31 | P2.7/A15 |
| 2 | P1.0/T2 | 17 | P3.5/T1 | 32 | $\overline{PSEN}$ |
| 3 | P1.1/T2EX | 18 | P3.6/$\overline{WR}$ | 33 | ALE |
| 4 | P1.2/ECI | 19 | P3.7/$\overline{RD}$ | 34 | NC* |
| 5 | P1.3/CEX0 | 20 | XTAL2 | 35 | $\overline{EA}$ |
| 6 | P1.4/CEX1 | 21 | XTAL1 | 36 | P0.7/AD7 |
| 7 | P1.5/CEX2 | 22 | $V_{SS}$ | 37 | P0.6/AD6 |
| 8 | P1.6/CEX3 | 23 | NC* | 38 | P0.5/AD5 |
| 9 | P1.7/CEX4 | 24 | P2.0/A8 | 39 | P0.4/AD4 |
| 10 | RST | 25 | P2.1/A9 | 40 | P0.3/AD3 |
| 11 | P3.0/RxD | 26 | P2.2/A10 | 41 | P0.2/AD2 |
| 12 | NC* | 27 | P2.3/A11 | 42 | P0.1/AD1 |
| 13 | P3.1/TxD | 28 | P2.4/A12 | 43 | P0.0/AD0 |
| 14 | P3.2/$\overline{INT0}$ | 29 | P2.5/A13 | 44 | $V_{CC}$ |
| 15 | P3.3/$\overline{INT1}$ | 30 | P2.6/A14 | | |

* DO NOT CONNECT

SU00715B

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS




| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | P1.5/CEX2 | 16 | $V_{SS}$ | 31 | P0.6/AD6 |
| 2 | P1.6/CEX3 | 17 | NC* | 32 | P0.5/AD5 |
| 3 | P1.7/CEX4 | 18 | P2.0/A8 | 33 | P0.4/AD4 |
| 4 | RST | 19 | P2.1/A9 | 34 | P0.3/AD3 |
| 5 | P3.0/RxD | 20 | P2.2/A10 | 35 | P0.2/AD2 |
| 6 | NC* | 21 | P2.3/A11 | 36 | P0.1/AD1 |
| 7 | P3.1/TxD | 22 | P2.4/A12 | 37 | P0.0/AD0 |
| 8 | P3.2/$\overline{INT0}$ | 23 | P2.5/A13 | 38 | $V_{CC}$ |
| 9 | P3.3/$\overline{INT1}$ | 24 | P2.6/A14 | 39 | NC* |
| 10 | P3.4/T0 | 25 | P2.7/A15 | 40 | P1.0/T2 |
| 11 | P3.5/T1 | 26 | $\overline{PSEN}$ | 41 | P1.1/T2EX |
| 12 | P3.6/$\overline{WR}$ | 27 | ALE | 42 | P1.2/ECI |
| 13 | P3.7/$\overline{RD}$ | 28 | NC* | 43 | P1.3/CEX0 |
| 14 | XTAL2 | 29 | $\overline{EA}$ | 44 | P1.4/CEX1 |
| 15 | XTAL1 | 30 | P0.7/AD7 | | |

* DO NOT CONNECT

SU00716B

## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER DIP | LCC | QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification. External pull-ups are required during program verification. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also receives the low-order address byte during program memory verification. Alternate functions include: |
| | 1 | 2 | 40 | I/O | **T2 (P1.0):** Timer/Counter 2 external count input/Clockout (see Programmable Clock-Out) |
| | 2 | 3 | 41 | I | **T2EX (P1.1):** Timer/Counter 2 Reload/Capture/Direction Control |
| | 3 | 4 | 42 | I | **ECI (P1.2):** External Clock Input to the PCA |
| | 4 | 5 | 43 | I/O | **CEX0 (P1.3):** Capture/Compare External I/O for PCA module 0 |
| | 5 | 6 | 44 | I/O | **CEX1 (P1.4):** Capture/Compare External I/O for PCA module 1 |
| | 6 | 7 | 1 | I/O | **CEX2 (P1.5):** Capture/Compare External I/O for PCA module 2 |
| | 7 | 8 | 2 | I/O | **CEX3 (P1.6):** Capture/Compare External I/O for PCA module 3 |
| | 8 | 9 | 3 | I/O | **CEX4 (P1.7):** Capture/Compare External I/O for PCA module 4 |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |

## PIN DESCRIPTIONS (Continued)

| MNEMONIC | PIN NUMBER DIP | LCC | QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{CC}$. |
| ALE | 30 | 33 | 27 | O | **Address Latch Enable:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. ALE can be disabled by setting SFR auxiliary.0. With this bit set, ALE will be active only during a MOVX instruction. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the 8XC51FX is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}$ | 31 | 35 | 29 | I | **External Access Enable:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H and 7FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 7FFFH. If security bit 1 is programmed, $\overline{EA}$ will be internally latched on Reset. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTE:**
To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher than $V_{CC}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

## TIMER 2 OPERATION

### Timer 2

Timer 2 is a 16-bit Timer/Counter which can operate as either an event timer or an event counter, as selected by C/T2* in the special function register T2CON (see Figure 1). Timer 2 has three operating modes:Capture, Auto-reload (up or down counting) ,and Baud Rate Generator, which are selected by bits in the T2CON as shown in Table 2.

### Capture Mode

In the capture mode there are two options which are selected by bit EXEN2 in T2CON. If EXEN2=0, then timer 2 is a 16-bit timer or counter (as selected by C/T2* in T2CON) which, upon overflowing sets bit TF2, the timer 2 overflow bit. This bit can be used to generate an interrupt (by enabling the Timer 2 interrupt bit in the IE register/SFR table). If EXEN2= 1, Timer 2 operates as described above, but with the added feature that a 1- to -0 transition at external input T2EX causes the current value in the Timer 2 registers, TL2 and TH2, to be captured into registers RCAP2L and RCAP2H, respectively. In addition, the transition at T2EX causes bit EXF2 in T2CON to be set, and EXF2 like TF2 can generate an interrupt (which vectors to the same location as Timer 2 overflow interrupt. The Timer 2 interrupt service routine can interrogate TF2 and EXF2 to determine which event caused the interrupt). The capture mode is illustrated in Figure 2 (There is no reload value for TL2 and TH2 in this mode. Even when a capture event occurs from T2EX, the counter keeps on counting T2EX pin transitions or osc/12 pulses.).

### Auto-Reload Mode (Up or Down Counter)

In the 16-bit auto-reload mode, Timer 2 can be configured (as either a timer or counter (C/T2* in T2CON)) then programmed to count up or down. The counting direction is determined by bit DCEN(Down Counter Enable) which is located in the T2MOD register (see Figure 3). When reset is applied the DCEN=0 which means Timer 2 will default to counting up. If DCEN bit is set, Timer 2 can count up or down depending on the value of the T2EX pin.

Figure 4 shows Timer 2 which will count up automatically since DCEN=0. In this mode there are two options selected by bit EXEN2 in T2CON register. If EXEN2=0, then Timer 2 counts up to 0FFFFH and sets the TF2 (Overflow Flag) bit upon overflow. This causes the Timer 2 registers to be reloaded with the 16-bit value in RCAP2L and RCAP2H.

The values in RCAP2L and RCAP2H are preset by software means. If EXEN2=1, then a 16-bit reload can be triggered either by an overflow or by a 1-to-0 transition at input T2EX. This transition also sets the EXF2 bit. The Timer 2 interrupt, if enabled, can be generated when either TF2 or EXF2 are 1.

In Figure 5 DCEN=1 which enables Timer 2 to count up or down. This mode allows pin T2EX to control the direction of count. When a logic 1 is applied at pin T2EX Timer 2 will count up. Timer 2 will overflow at 0FFFFH and set the TF2 flag, which can then generate an interrupt, if the interrupt is enabled. This timer overflow also causes the 16–bit value in RCAP2L and RCAP2H to be reloaded into the timer registers TL2 and TH2.

When a logic 0 is applied at pin T2EX this causes Timer 2 to count down. The timer will underflow when TL2 and TH2 become equal to the value stored in RCAP2L and RCAP2H. Timer 2 underflow sets the TF2 flag and causes 0FFFFH to be reloaded into the timer registers TL2 and TH2.

The external flag EXF2 toggles when Timer 2 underflows or overflows. This EXF2 bit can be used as a 17th bit of resolution if needed. The EXF2 flag does not generate an interrupt in this mode of operation.

(MSB) (LSB)

| TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/$\overline{T2}$ | CP/$\overline{RL2}$ |
|---|---|---|---|---|---|---|---|

| Symbol | Position | Name and Significance |
|---|---|---|
| TF2 | T2CON.7 | Timer 2 overflow flag set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK or TCLK = 1. |
| EXF2 | T2CON.6 | Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software. EXF2 does not cause an interrupt in up/down counter mode (DCEN = 1). |
| RCLK | T2CON.5 | Receive clock flag. When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in modes 1 and 3. RCLK = 0 causes Timer 1 overflow to be used for the receive clock. |
| TCLK | T2CON.4 | Transmit clock flag. When set, causes the serial port to use Timer 2 overflow pulses for its transmit clock in modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock. |
| EXEN2 | T2CON.3 | Timer 2 external enable flag. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX. |
| TR2 | T2CON.2 | Start/stop control for Timer 2. A logic 1 starts the timer. |
| C/$\overline{T2}$ | T2CON.1 | Timer or counter select. (Timer 2)<br>0 = Internal timer (OSC/12)<br>1 = External event counter (falling edge triggered). |
| CP/$\overline{RL2}$ | T2CON.0 | Capture/Reload flag. When set, captures will occur on negative transitions at T2EX if EXEN2 = 1. When cleared, auto-reloads will occur either with Timer 2 overflows or negative transitions at T2EX when EXEN2 = 1. When either RCLK = 1 or TCLK = 1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow. |

SU00728

**Figure 1. Timer/Counter 2 (T2CON) Control Register**

**Table 2. Timer 2 Operating Modes**

| RCLK + TCLK | CP/$\overline{RL2}$ | TR2 | MODE |
|---|---|---|---|
| 0 | 0 | 1 | 16-bit Auto-reload |
| 0 | 1 | 1 | 16-bit Capture |
| 1 | X | 1 | Baud rate generator |
| X | X | 0 | (off) |

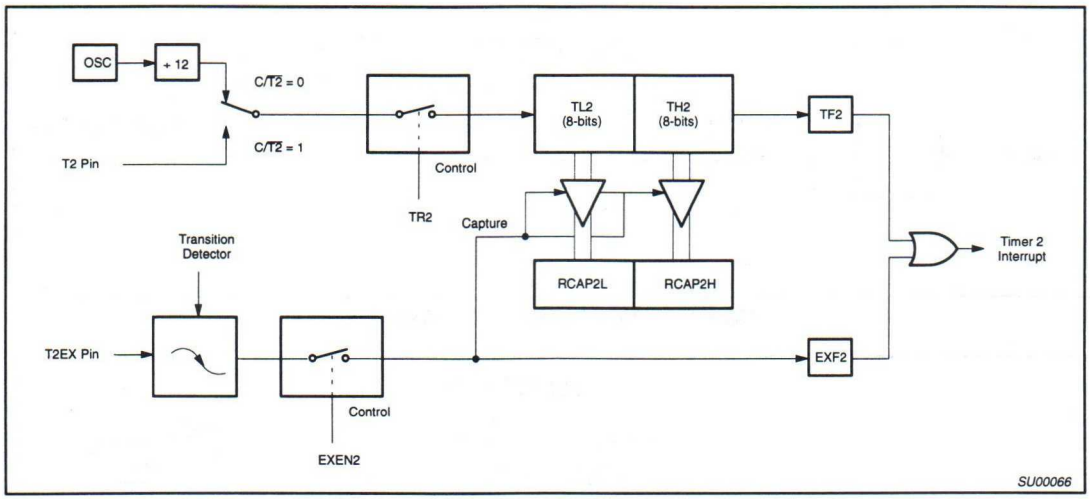



**Figure 2. Timer 2 in Capture Mode**

**T2MOD** Address = 0C9H Reset Value = XXXX XX00B

Not Bit Addressable

| — | — | — | — | — | — | T2OE | DCEN |
|---|---|---|---|---|---|---|---|
| Bit 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Symbol | Function |
|---|---|
| — | Not implemented, reserved for future use.* |
| T2OE | Timer 2 Output Enable bit. |
| DCEN | Down Count Enable bit. When set, this allows Timer 2 to be configured as an up/down counter. |

* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

SU00729

**Figure 3. Timer 2 Mode (T2MOD) Control Register**

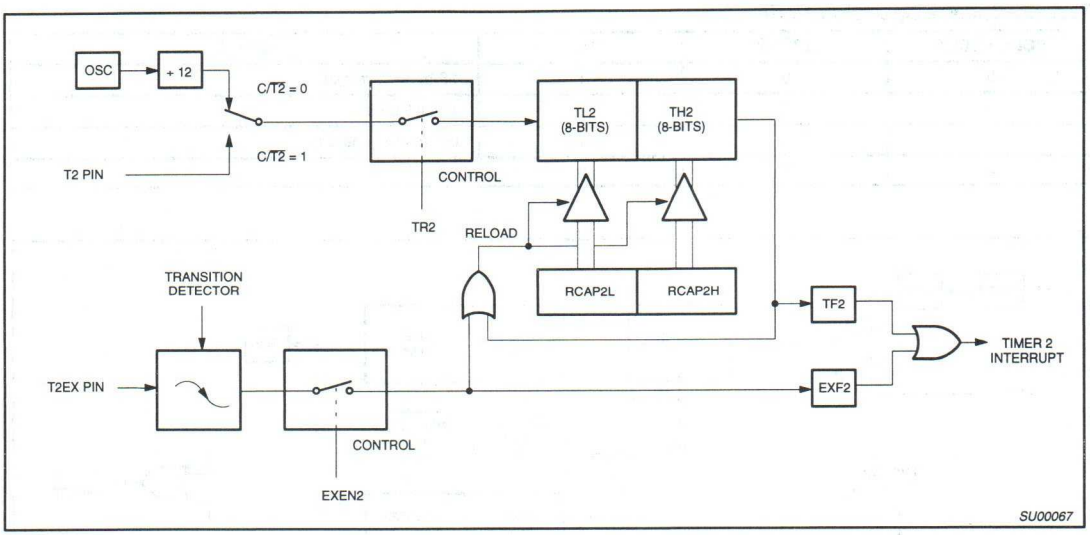



**Figure 4. Timer 2 in Auto-Reload Mode (DCEN = 0)**

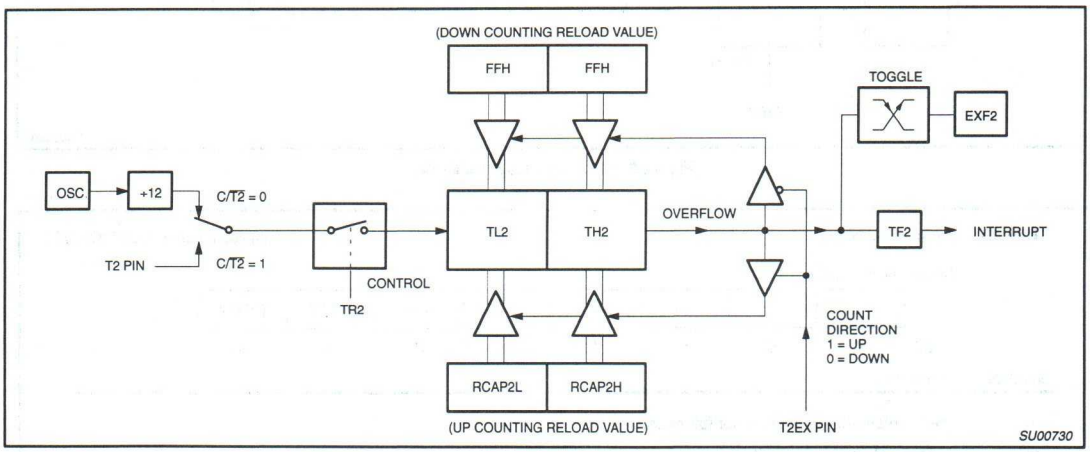



**Figure 5. Timer 2 Auto Reload Mode (DCEN = 1)**

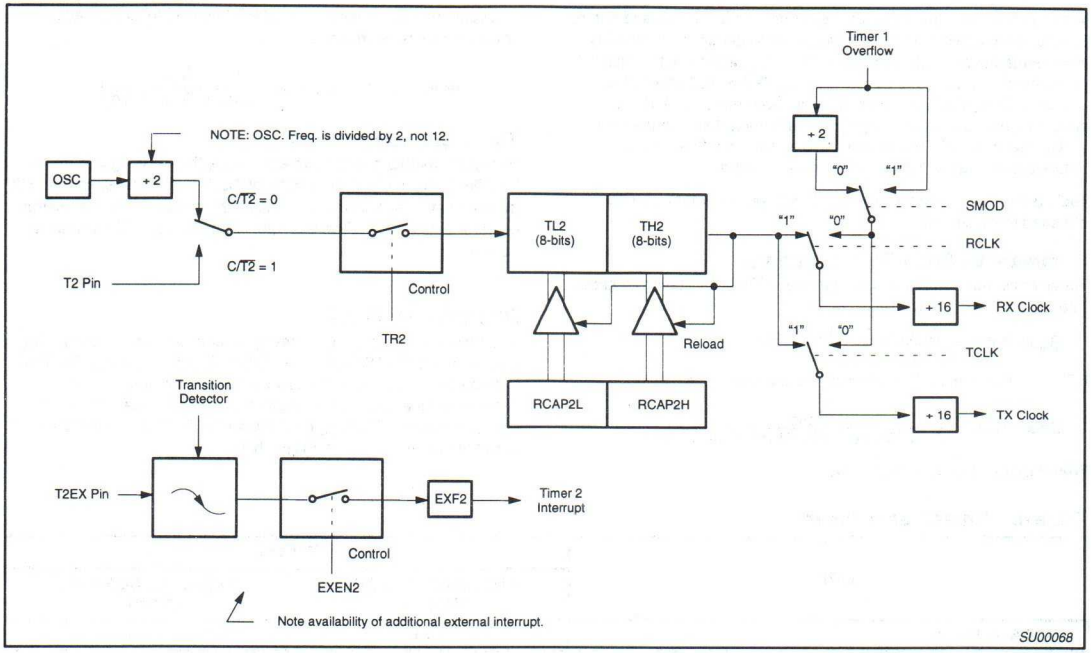



**Figure 6. Timer 2 in Baud Rate Generator Mode**

## Table 3. Timer 2 Generated Commonly Used Baud Rates

| Baud Rate | Osc Freq | Timer 2 | |
|---|---|---|---|
| | | RCAP2H | RCAP2L |
| 375K | 12MHz | FF | FF |
| 9.6K | 12MHz | FF | D9 |
| 2.8K | 12MHz | FF | B2 |
| 2.4K | 12MHz | FF | 64 |
| 1.2K | 12MHz | FE | C8 |
| 300 | 12MHz | FB | 1E |
| 110 | 12MHz | F2 | AF |
| 300 | 6MHz | FD | 8F |
| 110 | 6MHz | F9 | 57 |

## Baud Rate Generator Mode

Bits TCLK and/or RCLK in T2CON (Table 2) allow the serial port transmit and receive baud rates to be derived from either Timer 1 or Timer 2. When TCLK= 0, Timer 1 is used as the serial port transmit baud rate generator. When TCLK= 1, Timer 2 is used as the serial port transmit baud rate generator. RCLK has the same effect for the serial port receive baud rate. With these two bits, the serial port can have different receive and transmit baud rates – one generated by Timer 1, the other by Timer 2.

Figure 6 shows the Timer 2 in baud rate generation mode. The baud rate generation mode is like the auto-reload mode,in that a rollover in TH2 causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2H and RCAP2L, which are preset by software.

The baud rates in modes 1 and 3 are determined by Timer 2's overflow rate given below:

$$\text{Modes 1 and 3 Baud Rates} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

The timer can be configured for either "timer" or "counter" operation. In many applications, it is configured for "timer" operation (C/T2*=0). Timer operation is different for Timer 2 when it is being used as a baud rate generator.

Usually, as a timer it would increment every machine cycle (i.e., 1/12 the oscillator frequency). As a baud rate generator, it increments every state time (i.e., 1/2 the oscillator frequency). Thus the baud rate formula is as follows:

$$\text{Modes 1 and 3 Baud Rates} = \frac{\text{Oscillator Frequency}}{[32 \times [65536 - (\text{RCAP2H, RCAP2L})]]}$$

Where: (RCAP2H, RCAP2L)= The content of RCAP2H and RCAP2L taken as a 16-bit unsigned integer.

The Timer 2 as a baud rate generator mode shown in Figure 6, is valid only if RCLK and/or TCLK = 1 in T2CON register. Note that a rollover in TH2 does not set TF2, and will not generate an interrupt. Thus, the Timer 2 interrupt does not have to be disabled when Timer 2 is in the baud rate generator mode. Also if the EXEN2 (T2 external enable flag) is set, a 1-to-0 transition in T2EX (Timer/counter 2 trigger input) will set EXF2 (T2 external flag) but will not cause a reload from (RCAP2H, RCAP2L) to (TH2,TL2). Therefore when Timer 2 is in use as a baud rate generator, T2EX can be used as an additional external interrupt, if needed.

When Timer 2 is in the baud rate generator mode, one should not try to read or write TH2 and TL2. As a baud rate generator, Timer 2 is incremented every state time (osc/2) or asynchronously from pin T2; under these conditions, a read or write of TH2 or TL2 may not be accurate. The RCAP2 registers may be read, but should not be written to, because a write might overlap a reload and cause write and/or reload errors. The timer should be turned off (clear TR2) before accessing the Timer 2 or RCAP2 registers.

Table 3 shows commonly used baud rates and how they can be obtained from Timer 2.

## Summary Of Baud Rate Equations

Timer 2 is in baud rate generating mode. If Timer 2 is being clocked through pin T2(P1.0) the baud rate is:

$$\text{Baud Rate} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

If Timer 2 is being clocked internally , the baud rate is:

$$\text{Baud Rate} = \frac{f_{OSC}}{[32 \times [65536 - (\text{RCAP2H, RCAP2L})]]}$$

Where $f_{OSC}$= Oscillator Frequency

To obtain the reload value for RCAP2H and RCAP2L, the above equation can be rewritten as:

$$\text{RCAP2H, RCAP2L} = 65536 - \left(\frac{f_{OSC}}{32 \times \text{Baud Rate}}\right)$$

## Timer/Counter 2 Set-up

Except for the baud rate generator mode, the values given for T2CON do not include the setting of the TR2 bit. Therefore, bit TR2 must be set, separately, to turn the timer on. see Table 4 for set-up of Timer 2 as a timer. Also see Table 5 for set-up of Timer 2 as a counter.

## POWER OFF FLAG

The Power Off Flag (POF) is set by on-chip circuitry when the $V_{CC}$ level on the 8XC51FA/83C51FB/83C51FC rises from 0 to 5V. The POF bit can be set or cleared by software allowing a user to determine if the reset is the result of a power-on or a warm start after powerdown. The $V_{CC}$ level must remain above 3V for the POF to remain unaffected by the $V_{CC}$ level.

**Table 4. Timer 2 as a Timer**

| MODE | T2CON | |
|---|---|---|
| | INTERNAL CONTROL (Note 1) | EXTERNAL CONTROL (Note 2) |
| 16-bit Auto-Reload | 00H | 08H |
| 16-bit Capture | 01H | 09H |
| Baud rate generator receive and transmit same baud rate | 34H | 36H |
| Receive only | 24H | 26H |
| Transmit only | 14H | 16H |

**Table 5. Timer 2 as a Counter**

| MODE | TMOD | |
|---|---|---|
| | INTERNAL CONTROL (Note 1) | EXTERNAL CONTROL (Note 2) |
| 16-bit | 02H | 0AH |
| Auto-Reload | 03H | 0BH |

**NOTES:**
1. Capture/reload occurs only on timer/counter overflow.
2. Capture/reload occurs on timer/counter overflow and a 1-to-0 transition on T2EX (P1.1) pin except when Timer 2 is used in the baud rate generator mode.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

### Reset

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up. Ports 1, 2, and 3 will asynchronously be driven to their reset condition when a voltage above $V_{IH1}$ (min.) is applied to RESET.

### Idle Mode

In the idle mode (see Table 6), the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

### Power-Down Mode

To save even more power, a Power Down mode (see Table 6) can be invoked by software. In this mode, the oscillator is stopped and the instruction that invoked Power Down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the Power Down mode is terminated.

Either a hardware reset or external interrupt can be used to exit from Power Down. Reset redefines all the SFRs but does not change the on-chip RAM. An external interrupt allows both the SFRs and the on-chip RAM to retain their values.

To properly terminate Power Down the reset or external interrupt should not be executed before $V_{CC}$ is restored to its normal operating level and must be held active long enough for the oscillator to restart and stabilize (normally less than 10ms).

With an external interrupt, INT0 and INT1 must be enabled and configured as level-sensitive. Holding the pin low restarts the oscillator but bringing the pin back high completes the exit. Once the interrupt is serviced, the next instruction to be executed after RETI will be the one following the instruction that put the device into Power Down.

### Design Consideration

- When the idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal rest algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

### ONCE™ Mode

The ONCE ("On-Circuit Emulation") Mode facilitates testing and debugging of systems using the 8XC51FA/FB without the 8XC51FA/FB/FC having to be removed from the circuit. The ONCE Mode is invoked by:

1. Pull ALE low while the device is in reset and $\overline{PSEN}$ is high;
2. Hold ALE low as RST is deactivated.

While the device is in ONCE Mode, the Port 0 pins go into a float state, and the other port pins and ALE and $\overline{PSEN}$ are weakly pulled high. The oscillator circuit remains active. While the 8XC51FA/FB is in this mode, an emulator or test CPU can be used to drive the circuit. Normal operation is restored when a normal reset is applied.

### Programmable Clock-Out

The 8XC51FA/83C51FB has a new feature. A 50% duty cycle clock can be programmed to come out on P1.0. This pin, besides being a regular I/O pin, has two alternate functions. It can be programmed:

1. to input the external clock for Timer/Counter 2, or
2. to output a 50% duty cycle clock ranging from 61Hz to 4MHz at a 16MHz operating frequency.

To configure the Timer/Counter 2 as a clock generator, bit C/T2 (in T2CON) must be cleared and bit T20E in T2MOD must be set. Bit TR2 (T2CON.2) also must be set to start the timer.

The Clock-Out frequency depends on the oscillator frequency and the reload value of Timer 2 capture registers (RCAP2H, RCAP2L) as shown in this equation:

$$\frac{\text{Oscillator Frequency}}{4 \times (65536 - \text{RCAP2H, RCAP2L})}$$

In the Clock-Out mode Timer 2 roll-overs will not generate an interrupt. This is similar to when it is used as a baud-rate generator. It is possible to use Timer 2 as a baud-rate generator and a clock generator simultaneously. Note, however, that the baud-rate and the Clock-Out frequency will be the same.

**Table 6. External Pin Status During Idle and Power-Down Mode**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

## Programmable Counter Array (PCA)

The Programmable Counter Array is a special Timer that has five 16-bit capture/compare modules associated with it. Each of the modules can be programmed to operate in one of four modes: rising and/or falling edge capture, software timer, high-speed output, or pulse width modulator. Each module has a pin associated with it in port 1. Module 0 is connected to P1.3(CEX0), module 1 to P1.4(CEX1), etc. The basic PCA configuration is shown in Figure 7.

The PCA timer is a common time base for all five modules and can be programmed to run at: 1/12 the oscillator frequency, 1/4 the oscillator frequency, the Timer 0 overflow, or the input on the ECI pin (P1.2). The timer count source is determined from the CPS1 and CPS0 bits in the CMOD SFR as follows (see Figure 10):

| CPS1 | CPS0 | PCA Timer Count Source |
|---|---|---|
| 0 | 0 | 1/12 oscillator frequency |
| 0 | 1 | 1/4 oscillator frequency |
| 1 | 0 | Timer 0 overflow |
| 1 | 1 | External Input at ECI pin |

In the CMOD SFR are three additional bits associated with the PCA. They are CIDL which allows the PCA to stop during idle mode, WDTE which enables or disables the watchdog function on module 4, and ECF which when set causes an interrupt and the PCA overflow flag CF (in the CCON SFR) to be set when the PCA timer overflows. These functions are shown in Figure 8.

The watchdog timer function is implemented in module 4 (see Figure 17).

The CCON SFR contains the run control bit for the PCA and the flags for the PCA timer (CF) and each module (refer to Figure 11). To run the PCA the CR bit (CCON.6) must be set by software. The PCA is shut off by clearing this bit. The CF bit (CCON.7) is set when the PCA counter overflows and an interrupt will be generated if the ECF bit in the CMOD register is set, The CF bit can only be cleared by software. Bits 0 through 4 of the CCON register are the flags for the modules (bit 0 for module 0, bit 1 for module 1, etc.) and are set by hardware when either a match or a capture occurs. These flags also can only be cleared by software. The PCA interrupt system shown in Figure 9.

Each module in the PCA has a special function register associated with it. These registers are: CCAPM0 for module 0, CCAPM1 for module 1, etc. (see Figure 12). The registers contain the bits that control the mode that each module will operate in. The ECCF bit (CCAPMn.0 where n=0, 1, 2, 3, or 4 depending on the module) enables the CCF flag in the CCON SFR to generate an interrupt when a match or compare occurs in the associated module. PWM (CCAPMn.1) enables the pulse width modulation mode. The TOG bit (CCAPMn.2) when set causes the CEX output associated with the module to toggle when there is a match between the PCA counter and the module's capture/compare register. The match bit MAT (CCAPMn.3) when set will cause the CCFn bit in the CCON register to be set when there is a match between the PCA counter and the module's capture/compare register.

The next two bits CAPN (CCAPMn.4) and CAPP (CCAPMn.5) determine the edge that a capture input will be active on. The CAPN bit enables the negative edge, and the CAPP bit enables the positive edge. If both bits are set both edges will be enabled and a capture will occur for either transition. The last bit in the register ECOM (CCAPMn.6) when set enables the comparator function. Figure 13 shows the CCAPMn settings for the various PCA functions.

There are two additional registers associated with each of the PCA modules. They are CCAPnH and CCAPnL and these are the registers that store the 16-bit count when a capture occurs or a compare should occur. When a module is used in the PWM mode these registers are used to control the duty cycle of the output.

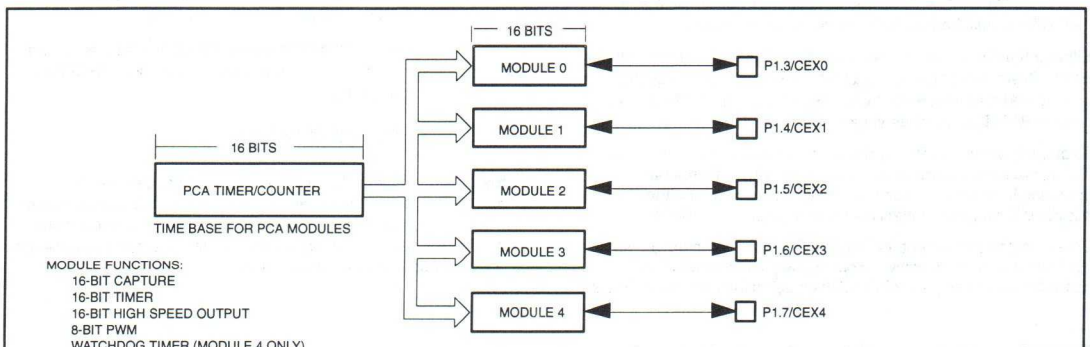



**Figure 7. Programmable Counter Array (PCA)**

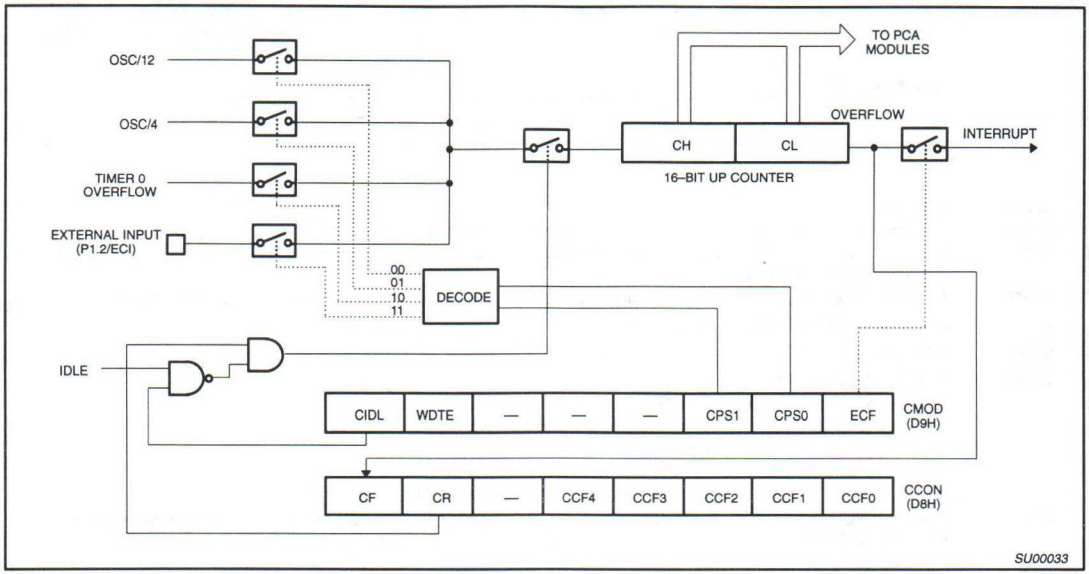



**Figure 8. PCA Timer/Counter**

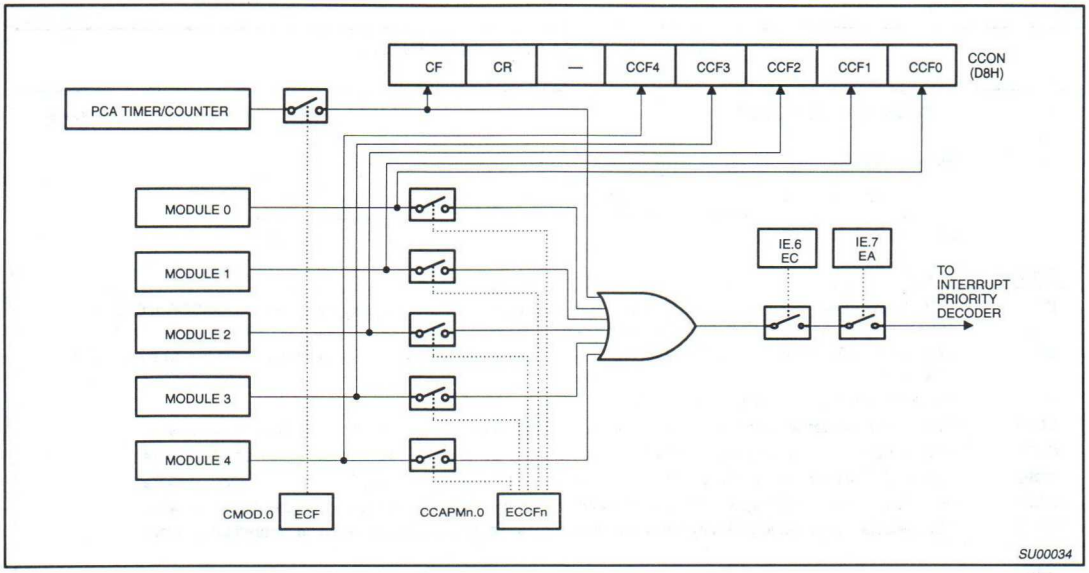



**Figure 9. PCA Interrupt System**

**CMOD** Address = OD9H — Reset Value = 00XX X000B

Bit Addressable

| | CIDL | WDTE | – | – | – | CPS1 | CPS0 | ECF |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Symbol | Function |
|---|---|
| **CIDL** | Counter Idle control: CIDL = 0 programs the PCA Counter to continue functioning during idle Mode. CIDL = 1 programs it to be gated off during idle. |
| **WDTE** | Watchdog Timer Enable: WDTE = 0 disables Watchdog Timer function on PCA Module 4. WDTE = 1 enables it. |
| – | Not implemented, reserved for future use.* |
| **CPS1** | PCA Count Pulse Select bit 1. |
| **CPS0** | PCA Count Pulse Select bit 0. |

| CPS1 | CPS0 | Selected PCA Input** | |
|---|---|---|---|
| 0 | 0 | 0 | Internal clock, $f_{OSC} \div 12$ |
| 0 | 1 | 1 | Internal clock, $f_{OSC} \div 4$ |
| 1 | 0 | 2 | Timer 0 overflow |
| 1 | 1 | 3 | External clock at ECI/P1.2 pin (max. rate = $f_{OSC} \div 8$) |

| | |
|---|---|
| **ECF** | PCA Enable Counter Overflow interrupt: ECF = 1 enables CF bit in CCON to generate an interrupt. ECF = 0 disables that function of CF. |

**NOTE:**
* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.
** $f_{OSC}$ = oscillator frequency

*SU00035*

**Figure 10. CMOD: PCA Counter Mode Register**

**CCON** Address = OD8H — Reset Value = 00X0 0000B

Bit Addressable

| | CF | CR | – | CCF4 | CCF3 | CCF2 | CCF1 | CCF0 |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Symbol | Function |
|---|---|
| **CF** | PCA Counter Overflow flag. Set by hardware when the counter rolls over. CF flags an interrupt if bit ECF in CMOD is set. CF may be set by either hardware or software but can only be cleared by software. |
| **CR** | PCA Counter Run control bit. Set by software to turn the PCA counter on. Must be cleared by software to turn the PCA counter off. |
| – | Not implemented, reserved for future use*. |
| **CCF4** | PCA Module 4 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF3** | PCA Module 3 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF2** | PCA Module 2 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF1** | PCA Module 1 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF0** | PCA Module 0 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |

**NOTE:**
* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

*SU00036*

**Figure 11. CCON: PCA Counter Control Register**

**CCAPMn** Address CCAPM0 0DAH
CCAPM1 0DBH
CCAPM2 0DCH
CCAPM3 0DDH
CCAPM4 0DEH

Reset Value = X000 0000B

Not Bit Addressable

| – | ECOMn | CAPPn | CAPNn | MATn | TOGn | PWMn | ECCFn |
|---|---|---|---|---|---|---|---|
| Bit: 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Symbol | Function |
|---|---|
| – | Not implemented, reserved for future use*. |
| **ECOMn** | Enable Comparator. ECOMn = 1 enables the comparator function. |
| **CAPPn** | Capture Positive, CAPPn = 1 enables positive edge capture. |
| **CAPNn** | Capture Negative, CAPNn = 1 enables negative edge capture. |
| **MATn** | Match. When MATn = 1, a match of the PCA counter with this module's compare/capture register causes the CCFn bit in CCON to be set, flagging an interrupt. |
| **TOGn** | Toggle. When TOGn = 1, a match of the PCA counter with this module's compare/capture register causes the CEXn pin to toggle. |
| **PWMn** | Pulse Width Modulation Mode. PWMn = 1 enables the CEXn pin to be used as a pulse width modulated output. |
| **ECCFn** | Enable CCF interrupt. Enables compare/capture flag CCFn in the CCON register to generate an interrupt. |

**NOTE:**
*User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

SU00037

**Figure 12. CCAPMn: PCA Modules Compare/Capture Registers**

| – | ECOMn | CAPPn | CAPNn | MATn | TOGn | PWMn | ECCFn | MODULE FUNCTION |
|---|---|---|---|---|---|---|---|---|
| X | 0 | 0 | 0 | 0 | 0 | 0 | 0 | No operation |
| X | X | 1 | 0 | 0 | 0 | 0 | X | 16-bit capture by a positive-edge trigger on CEXn |
| X | X | 0 | 1 | 0 | 0 | 0 | X | 16-bit capture by a negative trigger on CEXn |
| X | X | 1 | 1 | 0 | 0 | 0 | X | 16-bit capture by a transition on CEXn |
| X | 1 | 0 | 0 | 1 | 0 | 0 | X | 16-bit Software Timer |
| X | 1 | 0 | 0 | 1 | 1 | 0 | X | 16-bit High Speed Output |
| X | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 8-bit PWM |
| X | 1 | 0 | 0 | 1 | X | 0 | X | Watchdog Timer |

**Figure 13. PCA Module Modes (CCAPMn Register)**

### PCA Capture Mode

To use one of the PCA modules in the capture mode either one or both of the CCAPM bits CAPN and CAPP for that module must be set. The external CEX input for the module (on port 1) is sampled for a transition. When a valid transition occurs the PCA hardware loads the value of the PCA counter registers (CH and CL) into the module's capture registers (CCAPnL and CCAPnH). If the CCFn bit for the module in the CCON SFR and the ECCFn bit in the CCAPMn SFR are set then an interrupt will be generated. Refer to Figure 14.

### 16-bit Software Timer Mode

The PCA modules can be used as software timers by setting both the ECOM and MAT bits in the modules CCAPMn register. The PCA timer will be compared to the module's capture registers and when a match occurs an interrupt will occur if the CCFn (CCON SFR) and the ECCFn (CCAPMn SFR) bits for the module are both set (see Figure 15).

### High Speed Output Mode

In this mode the CEX output (on port 1) associated with the PCA module will toggle each time a match occurs between the PCA counter and the module's capture registers. To activate this mode the TOG, MAT, and ECOM bits in the module's CCAPMn SFR must be set (see Figure 16).

### Pulse Width Modulator Mode

All of the PCA modules can be used as PWM outputs. Figure 17 shows the PWM function. The frequency of the output depends on the source for the PCA timer. All of the modules will have the same frequency of output because they all share the PCA timer. The duty cycle of each module is independently variable using the module's capture register CCAPLn. When the value of the PCA CL SFR is less than the value in the module's CCAPLn SFR the output will be low, when it is equal to or greater than the output will be high. When CL overflows from FF to 00, CCAPLn is reloaded with the value in CCAPHn. the allows updating the PWM without glitches. The PWM and ECOM bits in the module's CCAPMn register must be set to enable the PWM mode.

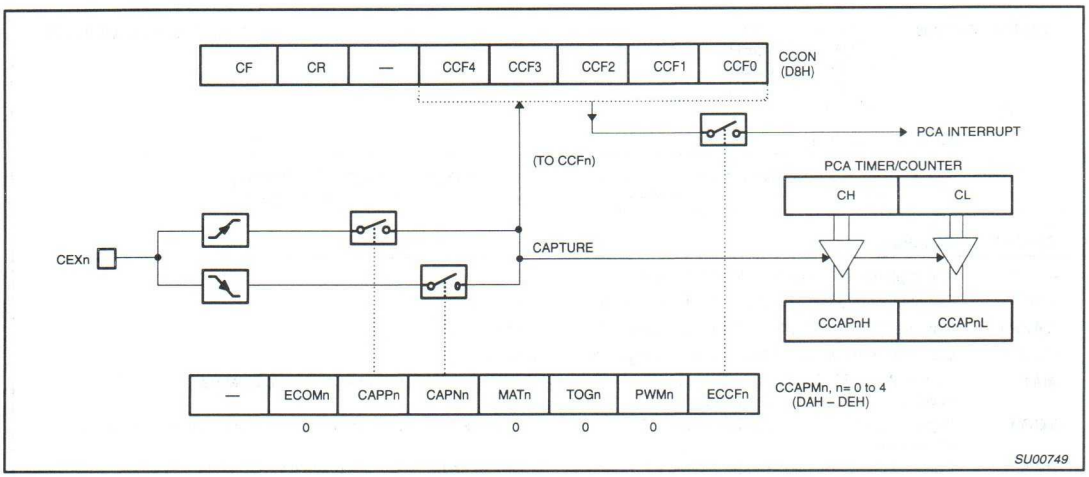



**Figure 14. PCA Capture Mode**

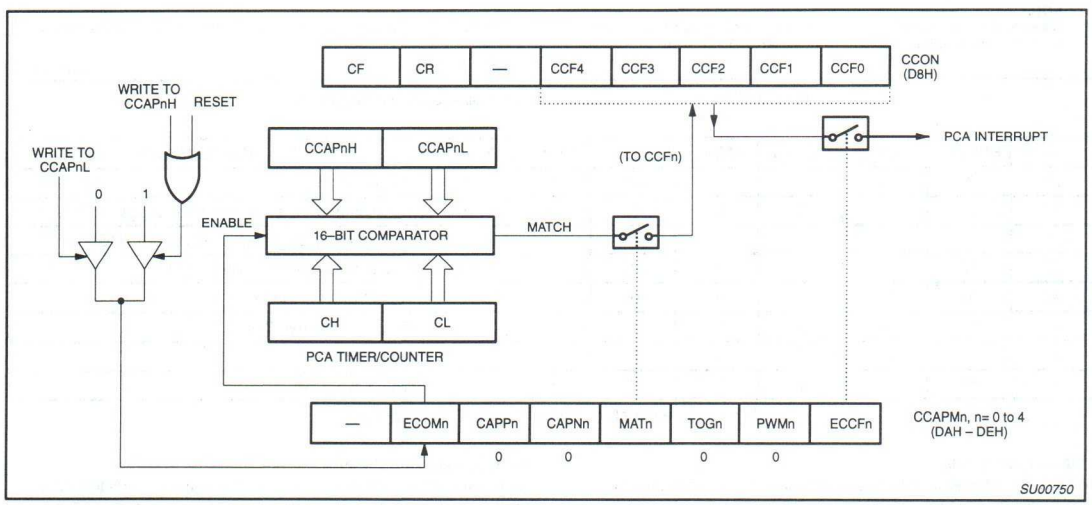



**Figure 15. PCA Compare Mode**

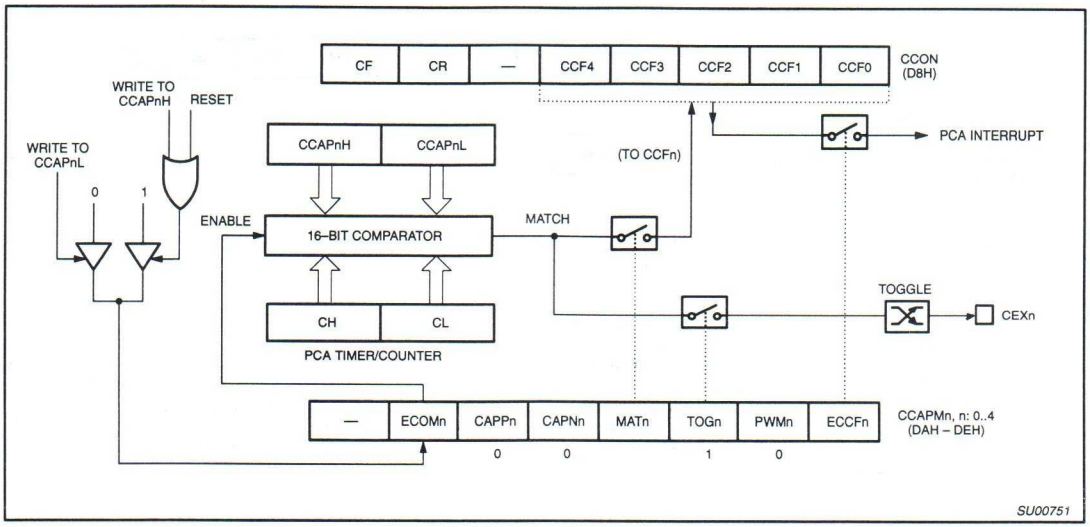



**Figure 16. PCA High Speed Output Mode**

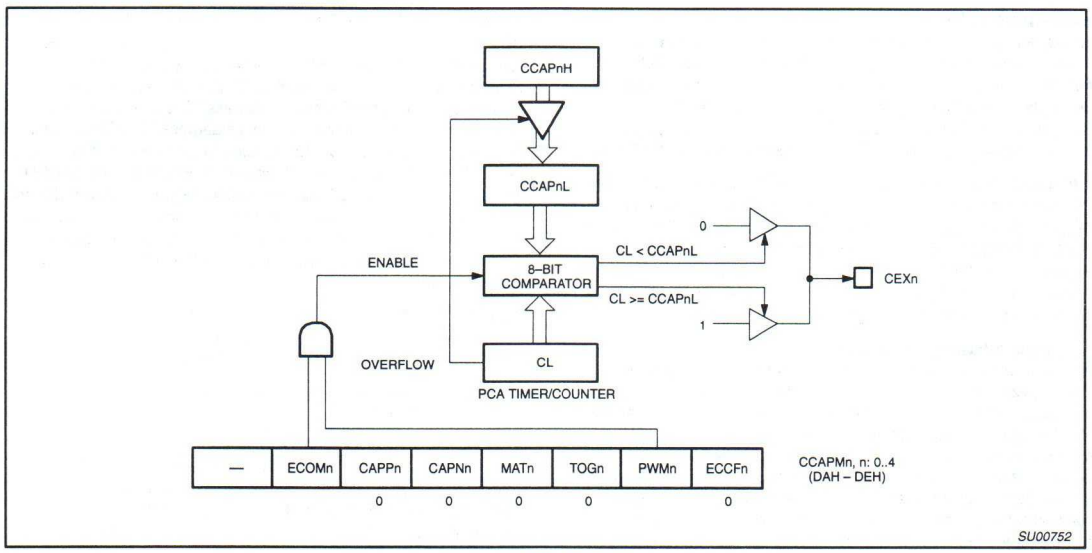



**Figure 17. PCA PWM Mode**

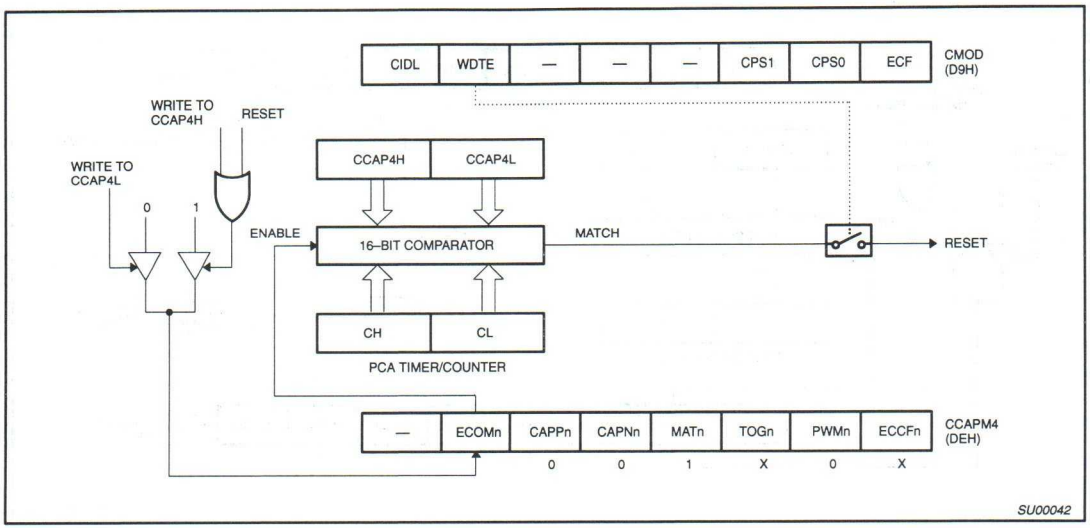



**Figure 18. PCA Watchdog Timer**

## Enhanced UART

The UART operates in all of the usual modes that are described in the first section of *Data Handbook IC20, 80C51-Based 8-Bit Microcontrollers*. In addition the UART can perform framing error detect by looking for missing stop bits, and automatic address recognition. The 8XC51FA/83C51FB UART also fully supports multiprocessor communication as does the standard 80C51 UART.

When used for framing error detect the UART looks for missing stop bits in the communication. A missing bit will set the FE bit in the SCON register. The FE bit shares the SCON.7 bit with SM0 and the function of SCON.7 is determined by PCON.6 (SMOD0) (see Figure 19). If SMOD0 is set then SCON.7 functions as FE. SCON.7 functions as SM0 when SMOD0 is cleared. When used as FE SCON.7 can only be cleared by software. Refer to Figure 20.

### Automatic Address Recognition

Automatic Address Recognition is a feature which allows the UART to recognize certain addresses in the serial bit stream by using hardware to make the comparisons. This feature saves a great deal of software overhead by eliminating the need for the software to examine every serial address which passes by the serial port. This feature is enabled by setting the SM2 bit in SCON. In the 9 bit UART modes, mode 2 and mode 3, the Receive Interrupt flag (RI) will be automatically set when the received byte contains either the "Given" address or the "Broadcast" address. The 9 bit mode requires that the 9th information bit is a 1 to indicate that the received information is an address and not data. Automatic address recognition is shown in Figure 21.

The 8 bit mode is called Mode 1. In this mode the RI flag will be set if SM2 is enabled and the information received has a valid stop bit following the 8 address bits and the information is either a Given or Broadcast address.

Mode 0 is the Shift Register mode and SM2 is ignored.

Using the Automatic Address Recognition feature allows a master to selectively communicate with one or more slaves by invoking the Given slave address or addresses. All of the slaves may be contacted by using the Broadcast address. Two special Function Registers are used to define the slave's address, SADDR, and the address mask, SADEN. SADEN is used to define which bits in the SADDR are to b used and which bits are "don't care". The SADEN mask can be logically ANDed with the SADDR to create the "lGiven" address which the master will use for addressing each of the slaves. Use of the Given address allows multiple slaves to be recognized while excluding others. The following examples will help to show the versatility of this scheme:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1101 |
| | Given | = | 1100 00X0 |
| Slave 1 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1110 |
| | Given | = | 1100 000X |

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires a 0 in bit 0 and it ignores bit 1. Slave 1 requires a 0 in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010 since slave 1 requires a 0 in bit 1. A unique address for slave 1 would be 1100 0001 since a 1 in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 1100 0000.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1001 |
| | Given | = | 1100 0XX0 |
| Slave 1 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1010 |
| | Given | = | 1110 0X0X |
| Slave 2 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1100 |
| | Given | = | 1110 00XX |

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 and 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary t make bit 2 = 1 to exclude slave 2.

The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are teated as don't-cares. In most cases, interpreting the don't-cares as ones, the broadcast address will be FF hexadecimal.

Upon reset SADDR (SFR address 0A9H) and SADEN (SFR address 0B9H) are leaded with 0s. This produces a given address of all "don't cares" as well as a Broadcast address of all "don't cares". this effectively disables the Automatic Addressing mode and allows the microcontroller to use standard 80C51 type UART drivers which do not make use of this feature.

**SCON** Address = 98H — Reset Value = 0000 0000B

Bit Addressable

| | SM0/FE | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

(SMOD0 = 0/1)*

| Symbol | Function |
|---|---|
| **FE** | Framing Error bit. This bit is set by the receiver when an invalid stop bit is detected. The FE bit is not cleared by valid frames but should be cleared by software. The SMOD0 bit must be set to enable access to the FE bit. |
| **SM0** | Serial Port Mode Bit 0, (SMOD0 must = 0 to access bit SM0) |
| **SM1** | Serial Port Mode Bit 1 |

| SM0 | SM1 | Mode | Description | Baud Rate** |
|---|---|---|---|---|
| 0 | 0 | 0 | shift register | $f_{OSC}/12$ |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{OSC}/64$ or $f_{OSC}/32$ |
| 1 | 1 | 3 | 9-bit UART | variable |

| | |
|---|---|
| **SM2** | Enables the Automatic Address Recognition feature in Modes 2 or 3. If SM2 = 1 then RI will not be set unless the received 9th data bit (RB8) is 1, indicating an address, and the received byte is a Given or Broadcast Address. In Mode 1, if SM2 = 1 then RI will not be activated unless a valid stop bit was received, and the received byte is a Given or Broadcast Address. In Mode 0, SM2 should be 0. |
| **REN** | Enables serial reception. Set by software to enable reception. Clear by software to disable reception. |
| **TB8** | The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired. |
| **RB8** | In modes 2 and 3, the 9th data bit that was received. In Mode 1, if SM2 = 0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used. |
| **TI** | Transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software. |
| **RI** | Receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software. |

**NOTE:**
*SMOD0 is located at PCON6.
**$f_{OSC}$ = oscillator frequency

SU00043

**Figure 19. SCON: Serial Port Control Register**

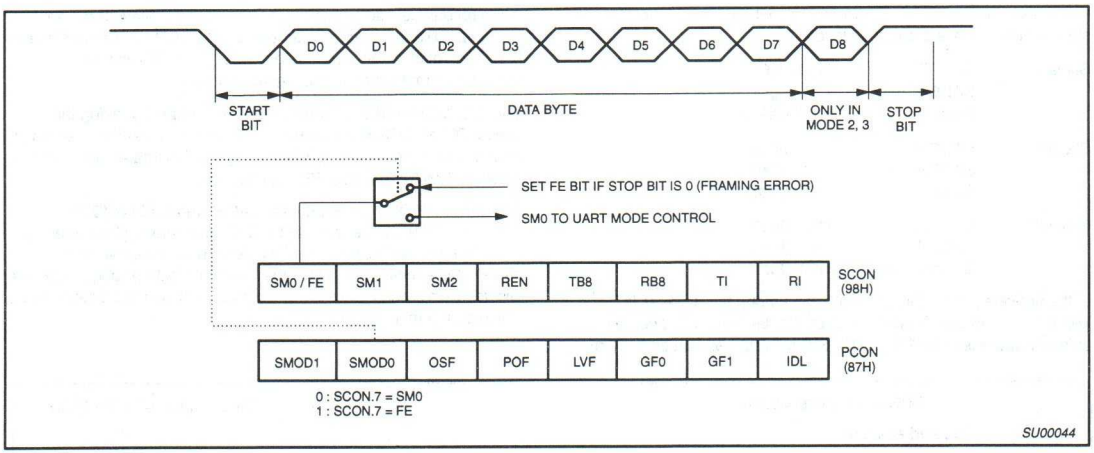



**Figure 20. UART Framing Error Detection**

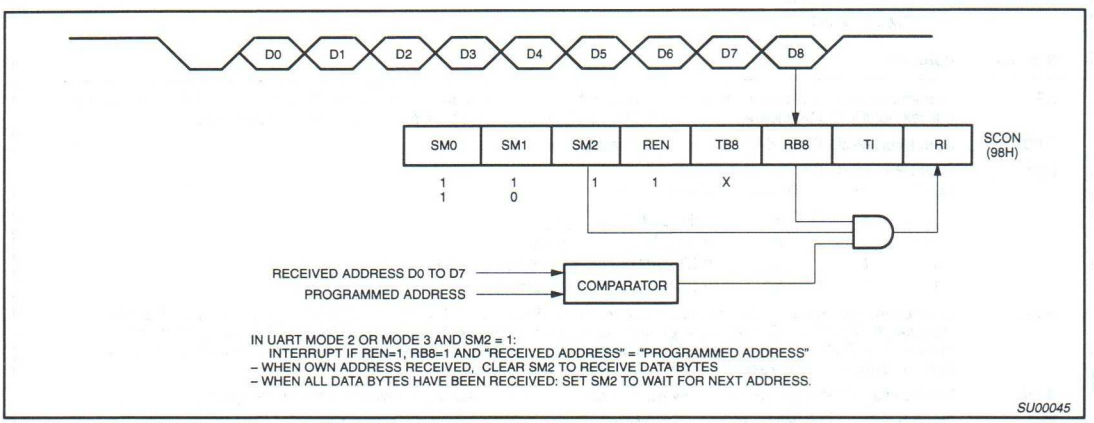



**Figure 21. UART Multiprocessor Communication, Automatic Address Recognition**

## Interrupt Priority Structure

The 8XC51FA/FB has a 7-source four-level interrupt structure. There are 3 SFRs associated with the interrupts on the 8XC51FA/FB. They are the IE and IP. (See Figures 22 and 23.) In addition, there is the IPH (Interrupt Priority High) register that makes the four-level interrupt structure possible. The IPH is located at SFR address B7H. The structure of the IPH register and a description of its bits is shown below:

**IPH (Interrupt Priority High) (B7H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | PPCH | PT2H | PSH | PT1H | PX1H | PT0H | PX0H |

IPH.0 PX0H External interrupt 0 priority high
IPH.1 PT0H Timer 0 interrupt priority high
IPH.2 PX1H External interrupt 1 priority high
IPH.3 PT1H Timer 1 interrupt priority high
IPH.4 PSH Serial Port interrupt high
IPH.5 PT2H Timer 2 interrupt priority high
IPH.6 PPCH PCA interrupt priority high
IPH.7 — Not implemented

The function of the IPH SFR is simple and when combined with the IP SFR determines the priority of each interrupt. The priority of each interrupt is determined as shown in the following table:

| PRIORITY BITS | | INTERRUPT PRIORITY LEVEL |
|---|---|---|
| IPH.x | IP.x | |
| 0 | 0 | Level 0 (lowest priority) |
| 0 | 1 | Level 1 |
| 1 | 0 | Level 2 |
| 1 | 1 | Level 3 (highest priority) |

The priority scheme for servicing the interrupts is the same as that for the 80C51, except there are four interrupt levels on the 8XC51FX rather than two as on the 80C51. An interrupt will be serviced as long as an interrupt of equal or higher priority is not already being serviced. If an interrupt of equal or higher level priority is being serviced, the new interrupt will wait until it is finished before being serviced. If a lower priority level interrupt is being serviced, it will be stopped and the new interrupt serviced. When the new interrupt is finished, the lower priority level interrupt that was stopped will be completed.

**Table 7. Interrupt Table**

| SOURCE | POLLING PRIORITY | REQUEST BITS | HARDWARE CLEAR? | VECTOR ADDRESS |
|---|---|---|---|---|
| X0 | 1 | IE0 | N (L)[1] Y (T)[2] | 03H |
| T0 | 2 | TP0 | Y | 0BH |
| X1 | 3 | IE1 | N (L) Y (T) | 13H |
| T1 | 4 | TF1 | Y | 1BH |
| SP | 5 | R1, TI | N | 23H |
| T2 | 6 | TF2, EXF2 | N | 2BH |
| PCA | 7 | CF, CCFn n = 0–4 | N | 33H |

**NOTES:**
1. L = Level activated
2. T = Transition activated

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **IE (0A8H)** | EA | EC | ET2 | ES | ET1 | EX1 | ET0 | EX0 |

Enable Bit = 1 enables the interrupt.
Enable Bit = 0 disables it.

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IE.7 | EA | Global disable bit. If EA = 0, all interrupts are disabled. If EA = 1, each interrupt can be individually enabled or disabled by setting or clearing its enable bit. |
| IE.6 | EC | PCA interrupt enable bit. |
| IE.5 | ET2 | Timer 2 interrupt enable bit. |
| IE.4 | ES | Serial Port interrupt enable bit. |
| IE.3 | ET1 | Timer 1 interrupt enable bit. |
| IE.2 | EX1 | External interrupt 1 enable bit. |
| IE.1 | ET0 | Timer 0 interrupt enable bit. |
| IE.0 | EX0 | External interrupt 0 enable bit. |

SU00571

**Figure 22. IE Registers**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **IP (0B8H)** | — | PPC | PT2 | PS | PT1 | PX1 | PT0 | PX0 |

Priority Bit = 1 assigns high priority
Priority Bit = 0 assigns low priority

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IP.7 | — | Not implemented, reserved for future use. |
| IP.6 | PPC | PCA interrupt priority bit. |
| IP.5 | PT2 | Timer 2 interrupt priority bit. |
| IP.4 | PS | Serial Port interrupt priority bit. |
| IP.3 | PT1 | Timer 1 interrupt priority bit. |
| IP.2 | PX1 | External interrupt 1 priority bit. |
| IP.1 | PT0 | Timer 0 interrupt priority bit. |
| IP.0 | PX0 | External interrupt 0 priority bit. |

SU00572

**Figure 23. IP Registers**

## Reduced EMI Mode

The AO bit (AUXR.0) in the AUXR register when set disables the ALE output.

## 8XC51FA/83C51FB/83C51FC Reduced EMI Mode

**AUXR (8EH)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | – | – | – | AO |

AO: Turns off ALE output.

## Dual DPTR

The dual DPTR structure (see Figure 24) is a way by which the 83C51FA, 83C51FB, and 83C51FC will specify the address of an external data memory location. (*NOTE: not available on 80C51FA [ROMless] at this time.) There are two 16-bit DPTR registers that address the external memory, and a single bit called DPS = AUSR1/bit0 that allows the program code to switch between them.

- New Register Name: AUXR1#
- SFR Address: A2H
- Reset Value: xxxxxxx0B

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | – | – | – | DPS |

Where:
DPS = AUXR1/bit0 = Switches between DPTR0 and DPTR1.

| Select Reg | DPS |
|---|---|
| DPTR0 | 0 |
| DPTR1 | 1 |

The DPS bit status should be saved by software when switching between DPTR0 and DPTR1.

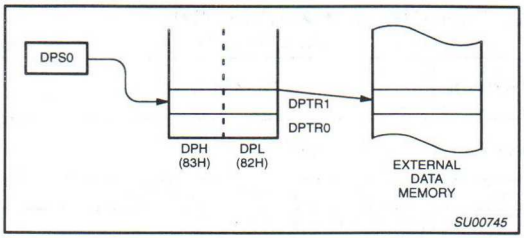



**Figure 24.**

### DPTR Instructions

The instructions that refer to DPTR refer to the data pointer that is currently selected using the AUXR1/bit 0 register. The six instructions that use the DPTR are as follows:

| | |
|---|---|
| INC DPTR | Increments the data pointer by 1 |
| MOV DPTR, #data16 | Loads the DPTR with a 16-bit constant |
| MOV A, @ A+DPTR | Move code byte relative to DPTR to ACC |
| MOVX A, @ DPTR | Move external RAM (16-bit address) to ACC |
| MOVX @ DPTR , A | Move ACC to external RAM (16-bit address) |
| JMP @ A + DPTR | Jump indirect relative to DPTR |

The data pointer can be accessed on a byte-by-byte basis be specifying the Low or High byte in an instruction which accesses the SFRs. See application note AN458 for more details.

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | 0 to +70 or –40 to +85 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Maximum $I_{OL}$ per I/O pin | 15 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.5 | W |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 2.7V to 5.5V ±10%, $V_{SS}$ = 0V; (–4 and –5 devices; 16MHz devices except S80C51FA [ROMless])

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $V_{IL}$ | Input low voltage | 4.0V < $V_{CC}$ < 5.5V | –0.5 | | 0.2$V_{CC}$–0.1 | V |
| | | 2.7V<$V_{CC}$< 4.0V | –0.5 | | 0.7 | V |
| $V_{IH}$ | Input high voltage (ports 0, 1, 2, 3, $\overline{EA}$) | | 0.2$V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | | 0.7$V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, [8] | $V_{CC}$ = 2.7V<br>$I_{OL}$ = 1.6mA[2] | | | 0.4 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$[8, 7] | $V_{CC}$ = 2.7V<br>$I_{OL}$ = 3.2mA[2] | | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3 [3] | $V_{CC}$ = 2.7V<br>$I_{OH}$ = –20µA | $V_{CC}$ – 0.7 | | | V |
| | | $V_{CC}$ = 4.5V<br>$I_{OH}$ = –30µA | $V_{CC}$ – 0.7 | | | V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode), ALE[9], $\overline{PSEN}$[3] | $V_{CC}$ = 2.7V<br>$I_{OH}$ = –3.2mA | $V_{CC}$ – 0.7 | | | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3 | $V_{IN}$ = 0.4V | –1 | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3[6] | $V_{IN}$ = 2.0V<br>See note 4 | | | –650 | µA |
| $I_{LI}$ | Input leakage current, port 0 | 0.45 < $V_{IN}$ < $V_{CC}$ – 0.3 | | | ±10 | µA |
| $I_{CC}$ | Power supply current (see Figure 32): | See note 5 | | | | |
| | Active mode @ 16MHz | | | | | |
| | 83C51FA/FB/FC | | | | 15 | mA |
| | 80C51FA | | | | 32 | mA |
| | Idle mode @ 16MHz | | | | | |
| | 83C51FA/FB/FC | | | | 4 | mA |
| | 80C51FA | | | | 5 | mA |
| | Power-down mode | $T_{amb}$ = 0°C to 70°C | | 3 | 50 | µA |
| | | $T_{amb}$ = –40°C to +85°C | | | 75 | µA |
| $R_{RST}$ | Internal reset pull-down resistor | | 40 | | 225 | kΩ |
| $C_{IO}$ | Pin capacitance[10] (except $\overline{EA}$) | | | | 15 | pF |

**NOTES:**

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $V_{CC}$–0.7 specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
5. See Figures 33 through 36 for $I_{CC}$ test conditions.
   Active mode: $I_{CC}$ = 0.9 × FREQ. + 1.1mA
   Idle mode: $I_{CC}$ = 0.18 × FREQ. +1.01mA; See Figure 32.
6. This value applies to $T_{amb}$ = 0°C to +70°C. For $T_{amb}$ = –40°C to +85°C, $I_{TL}$ = –750µA.
7. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
8. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 15mA (*NOTE: This is 85°C specification.)
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 71mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
9. ALE is tested to $V_{OH1}$, except when ALE is off then $V_{OH}$ is the voltage specification.
10. Pin capacitance is characterized but not tested. Pin capacitance is less than 25pF. Pin capacitance of ceramic package is less than 15pF (except $\overline{EA}$ is 25pF).

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, (–A, –B, –I, and –J devices; S80C51FA –4, –5); 5V ±10%; $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $V_{IL}$ | Input low voltage | 4.5V < $V_{CC}$ < 5.5V | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IH}$ | Input high voltage (ports 0, 1, 2, 3, $\overline{EA}$) | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3 [8] | $V_{CC}$ = 4.5V<br>$I_{OL}$ = 1.6mA[2] | | | 0.4 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$ [7, 8] | $V_{CC}$ = 4.5V<br>$I_{OL}$ = 3.2mA[2] | | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3 [3] | $V_{CC}$ = 4.5V<br>$I_{OH}$ = –30µA | $V_{CC}$ – 0.7 | | | V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode), ALE[9], $\overline{PSEN}$[3] | $V_{CC}$ = 4.5V<br>$I_{OH}$ = –3.2mA | $V_{CC}$ – 0.7 | | | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3 | $V_{IN}$ = 0.4V | –1 | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3[6] | $V_{IN}$ = 2.0V<br>See note 4 | | | –650 | µA |
| $I_{LI}$ | Input leakage current, port 0 | 0.45 < $V_{IN}$ < $V_{CC}$ – 0.3 | | | ±10 | µA |
| $I_{CC}$ | Power supply current (see Figure 32):<br>Active mode @ 16MHz[5]<br>Idle mode @ 16MHz[5]<br>Power-down mode | See note 5<br><br><br>$T_{amb}$ = 0°C to 70°C<br>$T_{amb}$ = –40°C to +85°C | | <br><br><br>3 | <br>15<br>4<br>50<br>75 | <br>µA<br>µA<br>µA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | 40 | | 225 | kΩ |
| $C_{IO}$ | Pin capacitance[10] (except $\overline{EA}$) | | | | 15 | pF |

**NOTES:**

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $V_{CC}$–0.7 specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
5. See Figures 33 through 36 for $I_{CC}$ test conditions.
   Active mode: $I_{CC}$ = 0.9 × FREQ. + 1.1mA
   Idle mode: $I_{CC}$ = 0.18 × FREQ. +1.0mA; See Figure 32.
6. This value applies to $T_{amb}$ = 0°C to +70°C. For $T_{amb}$ = –40°C to +85°C, $I_{TL}$ = –750µA.
7. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
8. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 15mA (*NOTE: This is 85°C specification.)
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 71mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
9. ALE is tested to $V_{OH1}$, except when ALE is off then $V_{OH}$ is the voltage specification.
10. Pin capacitance is characterized but not tested. Pin capacitance is less than 25pF. Pin capacitance of ceramic package is less than 15pF (except $\overline{EA}$ is 25pF).

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = +2.7V to +5.5V (except S80C51FA $V_{CC}$ = 5.0V ±10%), $V_{SS}$ = 0V[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 25 | Oscillator frequency<br>Speed versions : 4; 5 | | | 3.5 | 16 | MHz |
| $t_{LHLL}$ | 25 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 25 | Address valid to ALE low | 22 | | $t_{CLCL}-40$ | | ns |
| $t_{LLAX}$ | 25 | Address hold after ALE low | 32 | | $t_{CLCL}-30$ | | ns |
| $t_{LLIV}$ | 25 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 25 | ALE low to $\overline{PSEN}$ low | 32 | | $t_{CLCL}-30$ | | ns |
| $t_{PLPH}$ | 25 | $\overline{PSEN}$ pulse width | 142 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 25 | $\overline{PSEN}$ low to valid instruction in | | 82 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 25 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 25 | Input instruction float after $\overline{PSEN}$ | | 37 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 25 | Address to valid instruction in | | 207 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 25 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 26, 27 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 26, 27 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 26, 27 | $\overline{RD}$ low to valid data in | | 147 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 26, 27 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 26, 27 | Data float after $\overline{RD}$ | | 65 | | $2t_{CLCL}-60$ | ns |
| $t_{LLDV}$ | 26, 27 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 26, 27 | Address to valid data in | | 397 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 26, 27 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 137 | 239 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 26, 27 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 122 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 26, 27 | Data valid to $\overline{WR}$ transition | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{WHQX}$ | 26, 27 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{QVWH}$ | 27 | Data valid to $\overline{WR}$ high | 287 | | $7t_{CLCL}-150$ | | ns |
| $t_{RLAZ}$ | 26, 27 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 26, 27 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 29 | High time | 20 | | 20 | $t_{CLCL}-t_{CLCX}$ | ns |
| $t_{CLCX}$ | 29 | Low time | 20 | | 20 | $t_{CLCL}-t_{CHCX}$ | ns |
| $t_{CLCH}$ | 29 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 29 | Fall time | | 20 | | 20 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 28 | Serial port clock cycle time | 750 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 28 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 28 | Output data hold after clock rising edge | 8 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 28 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 28 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}-133$ | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 83C51FA/FB/FC and 80C51FA to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.
4. See application note AN457 for external memory interface.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 24MHz CLOCK | | VARIABLE CLOCK[4] | | 33MHz CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 25 | Oscillator frequency<br>Speed versions : A; B (24MHz)<br>: I; J (33MHz) | 3.5 | 24 | 3.5 | 33 | 3.5 | 33 | MHz |
| $t_{LHLL}$ | 25 | ALE pulse width | 43 | | $2t_{CLCL}$–40 | | 21 | | ns |
| $t_{AVLL}$ | 25 | Address valid to ALE low | 17 | | $t_{CLCL}$–25 | | 5 | | ns |
| $t_{LLAX}$ | 25 | Address hold after ALE low | 17 | | $t_{CLCL}$–25 | | | | ns |
| $t_{LLIV}$ | 25 | ALE low to valid instruction in | | 102 | | $4t_{CLCL}$–65 | | 55 | ns |
| $t_{LLPL}$ | 25 | ALE low to $\overline{PSEN}$ low | 17 | | $t_{CLCL}$–25 | | 5 | | ns |
| $t_{PLPH}$ | 25 | $\overline{PSEN}$ pulse width | 80 | | $3t_{CLCL}$–45 | | 45 | | ns |
| $t_{PLIV}$ | 25 | $\overline{PSEN}$ low to valid instruction in | | 65 | | $3t_{CLCL}$–60 | | 30 | ns |
| $t_{PXIX}$ | 25 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 25 | Input instruction float after $\overline{PSEN}$ | | 17 | | $t_{CLCL}$–25 | | 5 | ns |
| $t_{AVIV}$ | 25 | Address to valid instruction in | | 128 | | $5t_{CLCL}$–80 | | 70 | ns |
| $t_{PLAZ}$ | 25 | $\overline{PSEN}$ low to address float | | 10 | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | | | |
| $t_{RLRH}$ | 26, 27 | $\overline{RD}$ pulse width | 150 | | $6t_{CLCL}$–100 | | 82 | | ns |
| $t_{WLWH}$ | 26, 27 | $\overline{WR}$ pulse width | 150 | | $6t_{CLCL}$–100 | | 82 | | ns |
| $t_{RLDV}$ | 26, 27 | $\overline{RD}$ low to valid data in | | 118 | | $5t_{CLCL}$–90 | | 60 | ns |
| $t_{RHDX}$ | 26, 27 | Data hold after $\overline{RD}$ | 0 | | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 26, 27 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}$–28 | | 32 | ns |
| $t_{LLDV}$ | 26, 27 | ALE low to valid data in | | 183 | | $8t_{CLCL}$–150 | | 90 | ns |
| $t_{AVDV}$ | 26, 27 | Address to valid data in | | 210 | | $9t_{CLCL}$–165 | | 105 | ns |
| $t_{LLWL}$ | 26, 27 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 75 | 175 | $3t_{CLCL}$–50 | $3t_{CLCL}$+50 | 40 | 140 | ns |
| $t_{AVWL}$ | 26, 27 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 92 | | $4t_{CLCL}$–75 | | 45 | | ns |
| $t_{QVWX}$ | 26, 27 | Data valid to $\overline{WR}$ transition | 12 | | $t_{CLCL}$–30 | | 0 | | ns |
| $t_{WHQX}$ | 26, 27 | Data hold after $\overline{WR}$ | 17 | | $t_{CLCL}$–25 | | 5 | | ns |
| $t_{QVWH}$ | 27 | Data valid to $\overline{WR}$ high | 162 | | $7t_{CLCL}$–130 | | 80 | | ns |
| $t_{RLAZ}$ | 26, 27 | $\overline{RD}$ low to address float | | 0 | | 0 | | 0 | ns |
| $t_{WHLH}$ | 26, 27 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 17 | 67 | $t_{CLCL}$–25 | $t_{CLCL}$+25 | 5 | 55 | ns |
| **External Clock** | | | | | | | | | |
| $t_{CHCX}$ | 29 | High time | 17 | | 17 | $t_{CLCL}$–$t_{CLCX}$ | | | ns |
| $t_{CLCX}$ | 29 | Low time | 17 | | 17 | $t_{CLCL}$–$t_{CHCX}$ | | | ns |
| $t_{CLCH}$ | 29 | Rise time | | 5 | | 5 | | | ns |
| $t_{CHCL}$ | 29 | Fall time | | 5 | | 5 | | | ns |
| **Shift Register** | | | | | | | | | |
| $t_{XLXL}$ | 28 | Serial port clock cycle time | 505 | | $12t_{CLCL}$ | | 360 | | ns |
| $t_{QVXH}$ | 28 | Output data setup to clock rising edge | 283 | | $10t_{CLCL}$–133 | | 167 | | ns |
| $t_{XHQX}$ | 28 | Output data hold after clock rising edge | 3 | | $2t_{CLCL}$–80 | | | | ns |
| $t_{XHDX}$ | 28 | Input data hold after clock rising edge | 0 | | 0 | | 0 | | ns |
| $t_{XHDV}$ | 28 | Clock rising edge to input data valid | | 283 | | $10t_{CLCL}$–133 | | 167 | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 83C51FA/FB/FC and 80C51FA to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.
4. Variable clock is specified for oscillator frequencies greater than 16MHz to 33MHz. For frequencies equal or less than 16MHz, see 16MHz "AC Electrial Characteristics", page 3-52.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{\text{PSEN}}$
Q – Output data
R – $\overline{\text{RD}}$ signal
t – Time
V – Valid
W – $\overline{\text{WR}}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ =Time for ALE low to $\overline{\text{PSEN}}$ low.

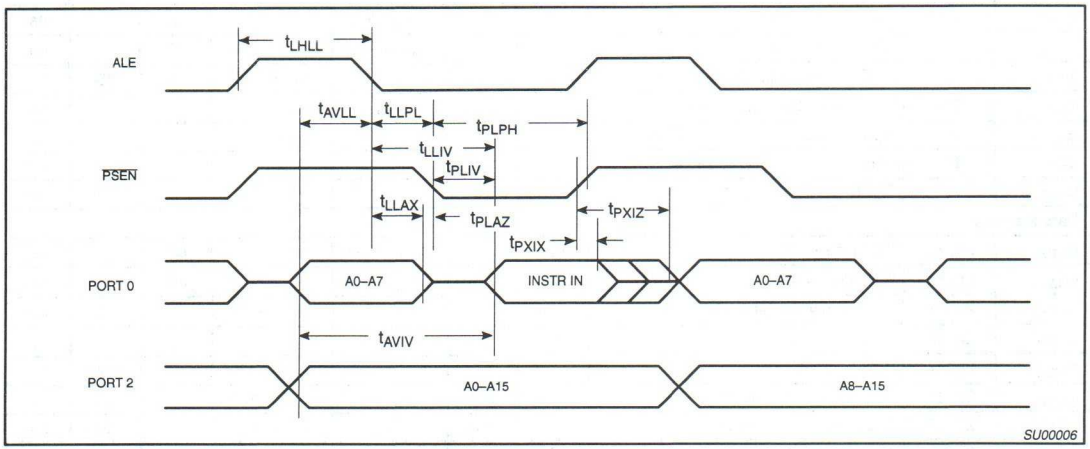



**Figure 25. External Program Memory Read Cycle**

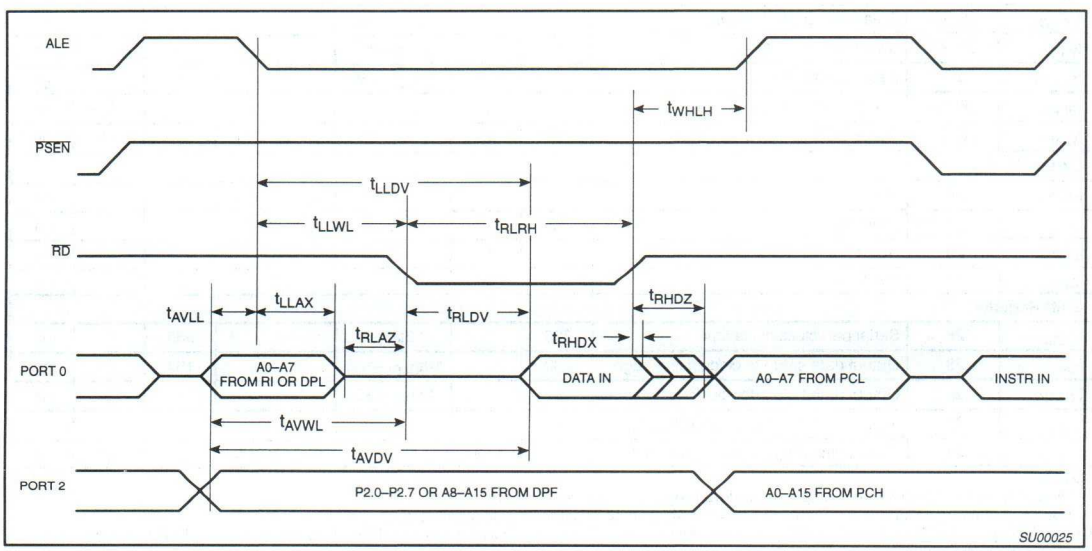



**Figure 26. External Data Memory Read Cycle**

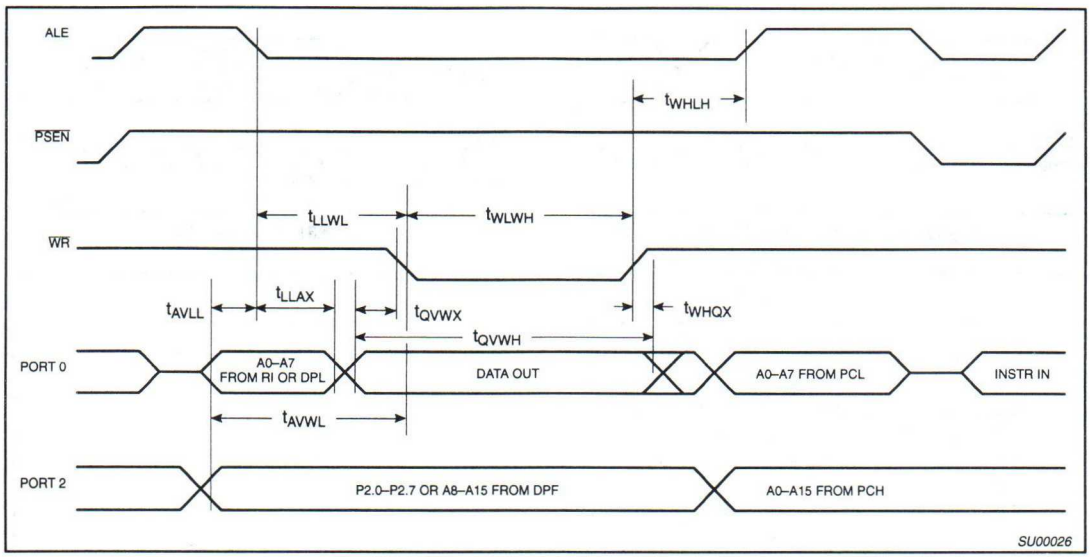



**Figure 27. External Data Memory Write Cycle**

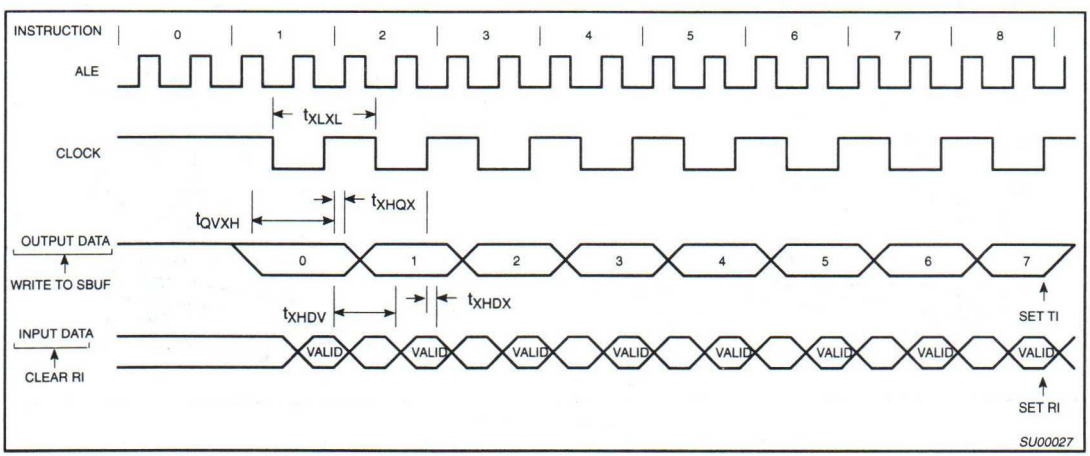



**Figure 28. Shift Register Mode Timing**

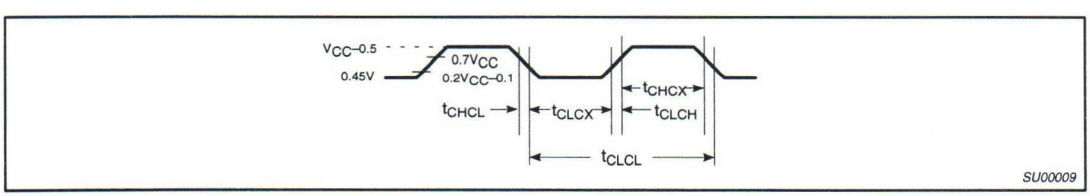
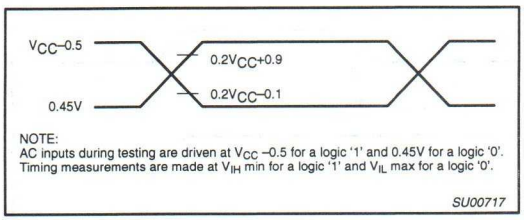


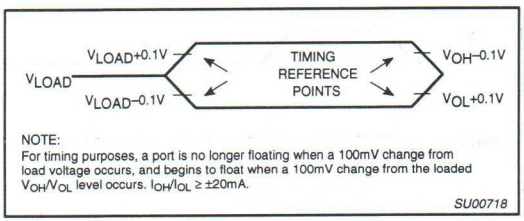

**Figure 29. External Clock Drive**




NOTE:
AC inputs during testing are driven at $V_{CC}$ –0.5 for a logic '1' and 0.45V for a logic '0'. Timing measurements are made at $V_{IH}$ min for a logic '1' and $V_{IL}$ max for a logic '0'.

SU00717

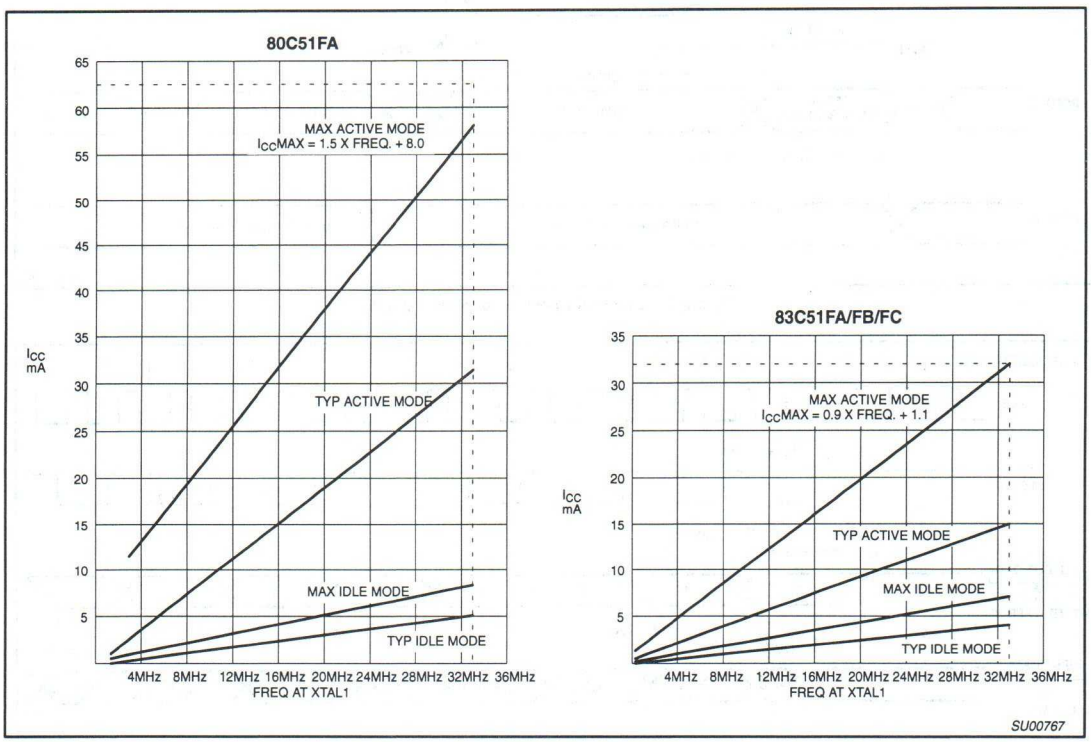
**Figure 30. AC Testing Input/Output**




NOTE:
For timing purposes, a port is no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded $V_{OH}/V_{OL}$ level occurs. $I_{OH}/I_{OL} \geq \pm 20$mA.

SU00718

**Figure 31. Float Waveform**




SU00767

**Figure 32. $I_{CC}$ vs. FREQ**
**Valid only within frequency specifications of the device under test**

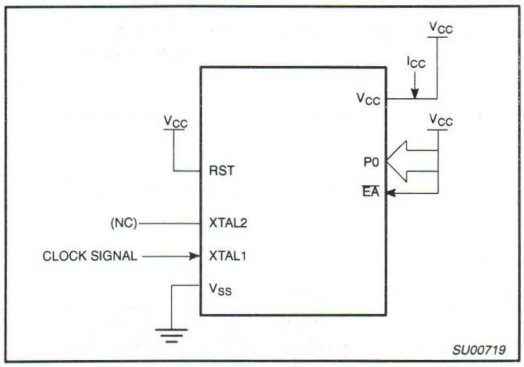



**Figure 33. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

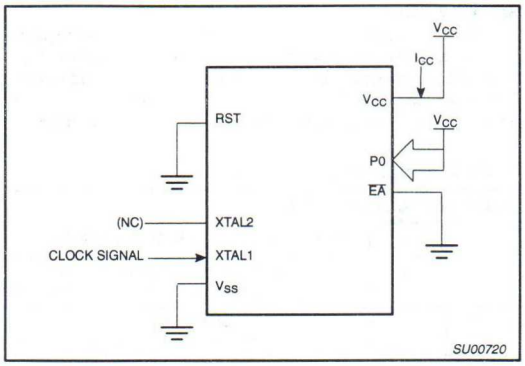



**Figure 34. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

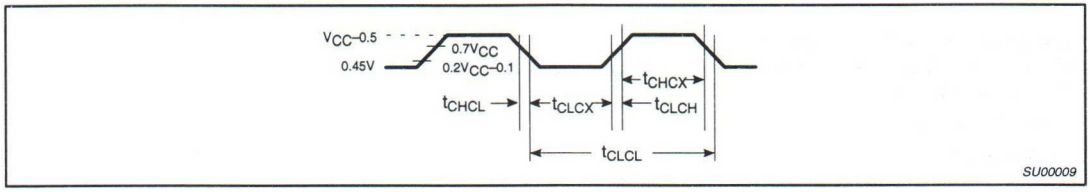



**Figure 35. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

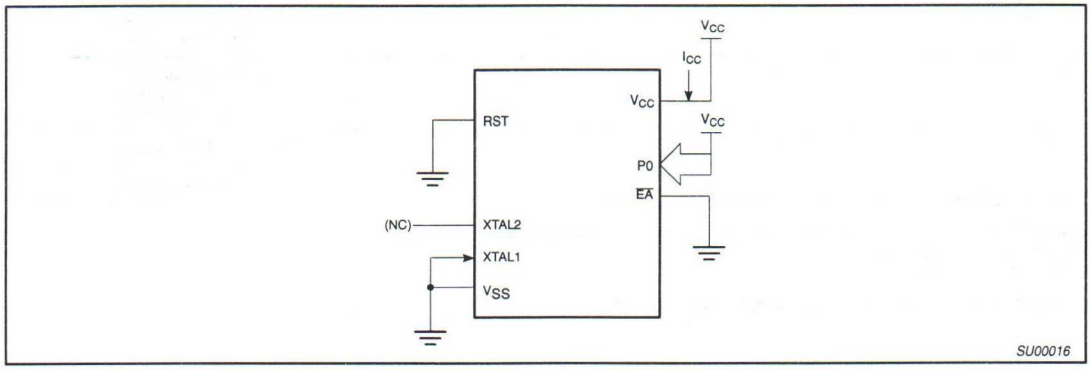



**Figure 36. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2V to 5.5V**

## Security Bits

With none of the security bits programmed the code in the program memory can be verified. If the encryption table is programmed, the code will be encrypted when verified. When only security bit 1 (see Table 8) is programmed, MOVC instructions executed from external program memory are disabled from fetching code bytes from the internal memory, EA is latched on Reset and all further programming of the EPROM is disabled. When security bits 1 and 2 are programmed, in addition to the above, verify mode is disabled.

## Encryption Array

64 bytes of encryption array are initially unprogrammed (all 1s).

**Table 8. Program Security Bits**

| PROGRAM LOCK BITS[1,2] | | | |
|---|---|---|---|
| | SB1 | SB2 | PROTECTION DESCRIPTION |
| 1 | U | U | No Program Security features enabled. (Code verify will still be encrypted by the Encryption Array if programmed.) |
| 2 | P | U | MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory, $\overline{EA}$ is sampled and latched on Reset, and further programming of the EPROM is disabled. |

**NOTES:**
1. P – programmed. U – unprogrammed.
2. Any other combination of the security bits is not defined.

## 83C51FA ROM CODE SUBMISSION

When submitting ROM code for the 83C51FA, the following must be specified:
1. 8k byte user ROM data
2. 64 byte ROM encryption key
3. ROM security bits.

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 1FFFH | DATA | 7:0 | User ROM Data |
| 2000H to 201FH | KEY | 7:0 | ROM Encryption Key<br>FFH = no encryption |
| 2020H | SEC | 0 | ROM Security Bit 1<br>0 = enable security<br>1 = disable security |
| 2020H | SEC | 1 | ROM Security Bit 2<br>0 = enable security<br>1 = disable security |

**Security Bit 1:** When programmed, this bit has two effects on masked ROM parts:
1. External MOVC is disabled, and
2. $\overline{EA}$ is latched on Reset.

**Security Bit 2:** When programmed, this bit inhibits Verify User ROM.

If the ROM Code file does not include the options, the following information must be included with the ROM code.

For each of the following, check the appropriate box, and send to Philips along with the code:

Security Bit #1: ☐ Enabled ☐ Disabled

Security Bit #2: ☐ Enabled ☐ Disabled

Encryption: ☐ No ☐ Yes If Yes, must send key file.

## 83C51FB ROM CODE SUBMISSION

When submitting ROM code for the 83C51FB, the following must be specified:

1. 16k byte user ROM data
2. 64 byte ROM encryption key
3. ROM security bits.

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 3FFFH | DATA | 7:0 | User ROM Data |
| 4000H to 401FH | KEY | 7:0 | ROM Encryption Key<br>FFH = no encryption |
| 4020H | SEC | 0 | ROM Security Bit 1<br>0 = enable security<br>1 = disable security |
| 4020H | SEC | 1 | ROM Security Bit 2<br>0 = enable security<br>1 = disable security |

**Security Bit 1:** When programmed, this bit has two effects on masked ROM parts:

1. External MOVC is disabled, and
2. $\overline{EA}$ is latched on Reset.

**Security Bit 2:** When programmed, this bit inhibits Verify User ROM.

If the ROM Code file does not include the options, the following information must be included with the ROM code.

For each of the following, check the appropriate box, and send to Philips along with the code:

Security Bit #1: ☐ Enabled ☐ Disabled

Security Bit #2: ☐ Enabled ☐ Disabled

Encryption: ☐ No ☐ Yes If Yes, must send key file.

## 83C51FC ROM CODE SUBMISSION

When submitting ROM code for the 83C51FC, the following must be specified:

1. 16k byte user ROM data
2. 64 byte ROM encryption key
3. ROM security bits.

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 7FFFH | DATA | 7:0 | User ROM Data |
| 8000H to 801FH | KEY | 7:0 | ROM Encryption Key<br>FFH = no encryption |
| 8020H | SEC | 0 | ROM Security Bit 1<br>0 = enable security<br>1 = disable security |
| 8020H | SEC | 1 | ROM Security Bit 2<br>0 = enable security<br>1 = disable security |

**Security Bit 1:** When programmed, this bit has two effects on masked ROM parts:

1. External MOVC is disabled, and
2. $\overline{EA}$ is latched on Reset.

**Security Bit 2:** When programmed, this bit inhibits Verify User ROM.

If the ROM Code file does not include the options, the following information must be included with the ROM code.

For each of the following, check the appropriate box, and send to Philips along with the code:

Security Bit #1: ☐ Enabled ☐ Disabled

Security Bit #2: ☐ Enabled ☐ Disabled

Encryption: ☐ No ☐ Yes If Yes, must send key file.

# CMOS single-chip 8-bit microcontrollers

# 87C51FA/87C51FB

## DESCRIPTION

The 87C51FA and 87C51FB Single-Chip 8-Bit Microcontrollers are manufactured in an advanced CMOS process and are derivatives of the 80C51 microcontroller family. The 87C51FA/FB has the same instruction set as the 80C51.

This device provides architectural enhancements that make it applicable in a variety of applications for general control systems. The 87C51FA contains 8k × 8 memory, and the 87C51FB contains 16k × 8 memory. They both contain a volatile 256 × 8 read/write data memory, four 8-bit I/O ports, three 16-bit timer/event counters, a Programmable Counter Array (PCA), a multi-source, two-priority-level, nested interrupt structure, an enhanced UART and on-chip oscillator and timing circuits. For systems that require extra capability, the 87C51FA/FB can be expanded using standard TTL compatible memories and logic.

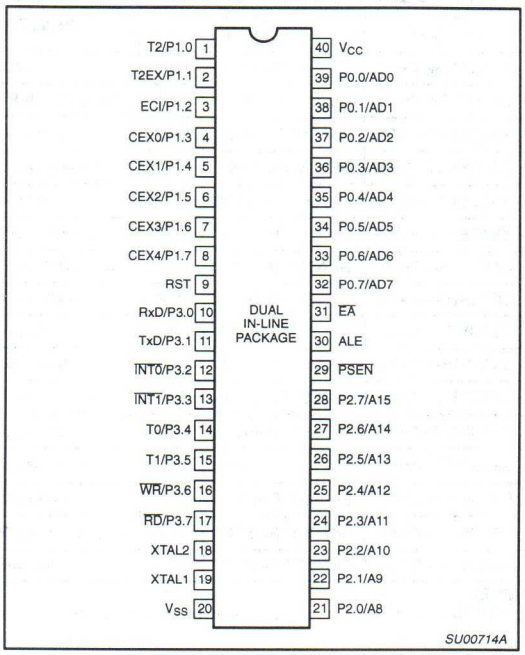
Its added features make it an even more powerful microcontroller for applications that require pulse width modulation, high-speed I/O and up/down counting capabilities such as motor control. It also has a more versatile serial channel that facilitates multiprocessor communications.

See 83C51FA/83C51FB/83C51FC/80C51FA datasheet for ROM and ROMless devices.

## FEATURES

- 80C51 central processing unit
- 87C51FA: 8k × 8 EPROM
  87C51FB: 16k × 8 EPROM
  - expandable externally to 64k bytes
  - Quick Pulse programming algorithm
  - Two level program security system
- 256 × 8 RAM, expandable externally to 64k bytes
- Three 16-bit timer/counters
  - T2 is an up/down counter
- Programmable Counter Array (PCA)
  - High speed output
  - Capture/compare
  - Pulse Width Modulator
  - Watchdog Timer
- Four 8-bit I/O ports
- Full-duplex enhanced UART
  - Framing error detection
  - Automatic address recognition
- Power control modes
  - Idle mode
  - Power-down mode
- Once (On Circuit Emulation) Mode
- Five package styles
- OTP package available

## PIN CONFIGURATIONS



## ORDERING INFORMATION

| 8k × 8 EPROM[1] | 16k × 8 EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQ. (MHz) | DRAWING NUMBER |
|---|---|---|---|---|---|
| S87C51FA–4N40 | S87C51FB–4N40 | OTP | 0 to +70, 40-Pin Plastic Dual In-line Package | 3.5 to 16 | SOT129-1 |
| S87C51FA–4F40 | S87C51FB–4F40 | UV | 0 to +70, 40-Pin Ceramic Dual In-line Package w/Window | 3.5 to 16 | 0590B |
| S87C51FA–4A44 | S87C51FB–4A44 | OTP | 0 to +70, 44-Pin Plastic Leaded Chip Carrier | 3.5 to 16 | SOT187-2 |
| S87C51FA–4K44 | S87C51FB–4K44 | UV | 0 to +70, 44-Pin Ceramic Leaded Chip Carrier w/Window | 3.5 to 16 | 1472A |
| S87C51FA–4B44 | S87C51FB–4B44 | OTP | 0 to +70, 44-Pin Plastic Quad Flat Pack | 3.5 to 16 | SOT307-2 |
| S87C51FA–5N40 | S87C51FB–5N40 | OTP | –40 to +85, 40-Pin Plastic Dual In-line Package | 3.5 to 16 | SOT129-1 |
| S87C51FA–5F40 | S87C51FB–5F40 | UV | –40 to +85, 40-Pin Ceramic Dual In-line Package w/Window | 3.5 to 16 | 0590B |
| S87C51FA–5A44 | S87C51FB–5A44 | OTP | –40 to +85, 44-Pin Plastic Leaded Chip Carrier | 3.5 to 16 | SOT187-2 |
| S87C51FA–5B44 | S87C51FB–5B44 | OTP | –40 to +85, 44-Pin Plastic Quad Flat Pack | 3.5 to 16 | SOT307-2 |
| S87C51FA–AN40 | S87C51FB–AN40 | OTP | 0 to +70, 40-Pin Plastic Dual In-line Package | 3.5 to 24 | SOT129-1 |
| S87C51FA–AF40 | S87C51FB–AF40 | UV | 0 to +70, 40-Pin Ceramic Dual In-line Package w/Window | 3.5 to 24 | 0590B |
| S87C51FA–AA44 | S87C51FB–AA44 | OTP | 0 to +70, 44-Pin Plastic Leaded Chip Carrier | 3.5 to 24 | SOT187-2 |
| S87C51FA–AK44 | S87C51FB–AK44 | UV | 0 to +70, 44-Pin Ceramic Leaded Chip Carrier w/Window | 3.5 to 24 | 1472A |
| S87C51FA–BN40 | S87C51FB–BN40 | OTP | –40 to +85, 40-Pin Plastic Dual In-line Package | 3.5 to 24 | SOT129-1 |
| S87C51FA–BF40 | S87C51FB–BF40 | UV | –40 to +85, 40-Pin Ceramic Dual In-line Package w/Window | 3.5 to 24 | 0590B |
| S87C51FA–BA44 | S87C51FB–BA44 | OTP | –40 to +85, 44-Pin Plastic Leaded Chip Carrier | 3.5 to 24 | SOT187-2 |

**NOTE:**
1. OTP = One Time Programmable EPROM. UV = Erasable EPROM.

## BLOCK DIAGRAM

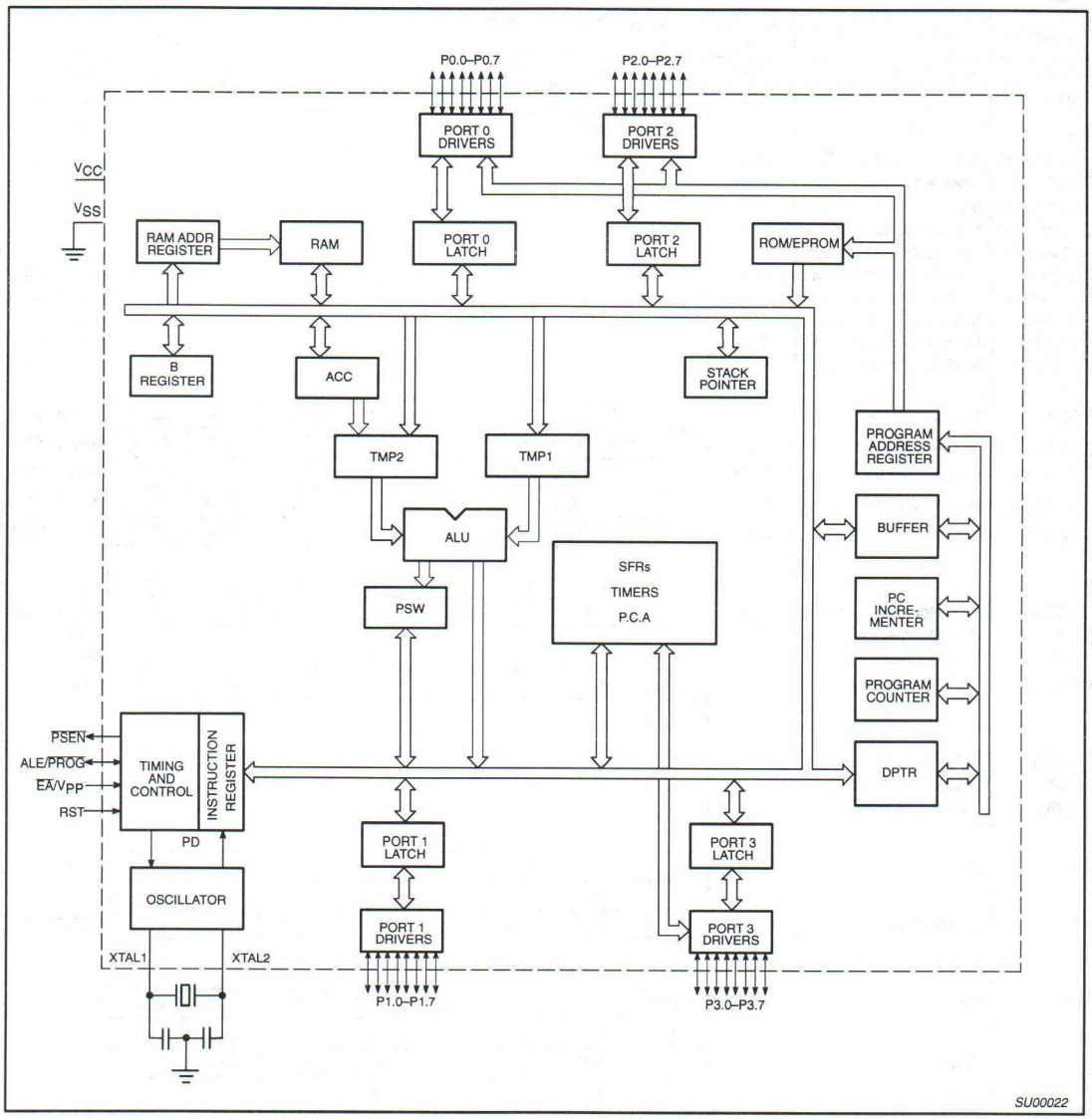


**Table 1. 87C51FA/87C51FB Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| AUXR# | Auxiliary | 8EH | – | – | – | – | – | – | – | AO | xxxxxxx0B |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| CCAP0H# | Module 0 Capture High | FAH | | | | | | | | | xxxxxxxxB |
| CCAP1H# | Module 1 Capture High | FBH | | | | | | | | | xxxxxxxxB |
| CCAP2H# | Module 2 Capture High | FCH | | | | | | | | | xxxxxxxxB |
| CCAP3H# | Module 3 Capture High | FDH | | | | | | | | | xxxxxxxxB |
| CCAP4H# | Module 4 Capture High | FEH | | | | | | | | | xxxxxxxxB |
| CCAP0L# | Module 0 Capture Low | EAH | | | | | | | | | xxxxxxxxB |
| CCAP1L# | Module 1 Capture Low | EBH | | | | | | | | | xxxxxxxxB |
| CCAP2L# | Module 2 Capture Low | ECH | | | | | | | | | xxxxxxxxB |
| CCAP3L# | Module 3 Capture Low | EDH | | | | | | | | | xxxxxxxxB |
| CCAP4L# | Module 4 Capture Low | EEH | | | | | | | | | xxxxxxxxB |
| CCAPM0# | Module 0 Mode | DAH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM1# | Module 1 Mode | DBH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM2# | Module 2 Mode | DCH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM3# | Module 3 Mode | DDH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM4# | Module 4 Mode | DEH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| CCON*# | PCA Counter Control | D8H | CF | CR | – | CCF4 | CCF3 | CCF2 | CCF1 | CCF0 | 00x00000B |
| CH# | PCA Counter High | F9H | | | | | | | | | 00H |
| CL# | PCA Counter Low | E9H | | | | | | | | | 00H |
| CMOD# | PCA Counter Mode | D9H | CIDL | WDTE | – | – | – | CPS1 | CPS0 | ECF | 00xxx000B |
| DPTR: | Data Pointer (2 bytes) | | | | | | | | | | |
| DPH | Data Pointer High | 83H | | | | | | | | | 00H |
| DPL | Data Pointer Low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE* | Interrupt Enable | A8H | EA | EC | ET2 | ES | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP* | Interrupt Priority | B8H | – | PPC | PT2 | PS | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | CEX4 | CEX3 | CEX2 | CEX1 | CEX0 | ECI | T2EX | T2 | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | AD15 | AD14 | AD13 | AD12 | AD11 | AD10 | AD9 | AD8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TxD | RxD | FFH |
| PCON# | Power Control | 87H | SMOD1 | SMOD0 | – | POF[1] | GF1 | GF0 | PD | IDL | 00xxxx00B |

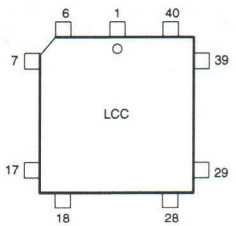
* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
1. Reset value depends on reset source.

**Table 1. 87C51FA/87C51FB Special Function Registers** (Continued)

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program Status Word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| RACAP2H# | Timer 2 Capture High | CBH | | | | | | | | | 00H |
| RACAP2L# | Timer 2 Capture Low | CAH | | | | | | | | | 00H |
| SADDR# | Slave Address | A9H | | | | | | | | | 00H |
| SADEN# | Slave Address Mask | B9H | | | | | | | | | 00H |
| SBUF | Serial Data Buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial Control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack Pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer Control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| T2CON* | Timer 2 Control | C8H | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/T2 | CP/RL2 | 00H |
| T2MOD# | Timer 2 Mode Control | C9H | – | – | – | – | – | – | T2OE[2] | DCEN | xxxxxx00B |
| TH0 | Timer High 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer High 1 | 8DH | | | | | | | | | 00H |
| TH2# | Timer High 2 | CDH | | | | | | | | | 00H |
| TL0 | Timer Low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer Low 1 | 8BH | | | | | | | | | 00H |
| TL2# | Timer Low 2 | CCH | | | | | | | | | 00H |
| TMOD | Timer Mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
2. T2OE – see Programmable Clock-Out.

## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

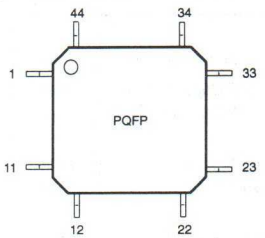



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | NC* | 16 | P3.4/T0 | 31 | P2.7/A15 |
| 2 | P1.0/T2 | 17 | P3.5/T1 | 32 | $\overline{PSEN}$ |
| 3 | P1.1/T2EX | 18 | P3.6/$\overline{WR}$ | 33 | ALE |
| 4 | P1.2/ECI | 19 | P3.7/$\overline{RD}$ | 34 | NC* |
| 5 | P1.3/CEX0 | 20 | XTAL2 | 35 | $\overline{EA}$ |
| 6 | P1.4/CEX1 | 21 | XTAL1 | 36 | P0.7/AD7 |
| 7 | P1.5/CEX2 | 22 | $V_{SS}$ | 37 | P0.6/AD6 |
| 8 | P1.6/CEX3 | 23 | NC* | 38 | P0.5/AD5 |
| 9 | P1.7/CEX4 | 24 | P2.0/A8 | 39 | P0.4/AD4 |
| 10 | RST | 25 | P2.1/A9 | 40 | P0.3/AD3 |
| 11 | P3.0/RxD | 26 | P2.2/A10 | 41 | P0.2/AD2 |
| 12 | NC* | 27 | P2.3/A11 | 42 | P0.1/AD1 |
| 13 | P3.1/TxD | 28 | P2.4/A12 | 43 | P0.0/AD0 |
| 14 | P3.2/$\overline{INT0}$ | 29 | P2.5/A13 | 44 | $V_{CC}$ |
| 15 | P3.3/$\overline{INT1}$ | 30 | P2.6/A14 | | |

* DO NOT CONNECT

SU00715B

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS




| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | P1.5/CEX2 | 16 | $V_{SS}$ | 31 | P0.6/AD6 |
| 2 | P1.6/CEX3 | 17 | NC* | 32 | P0.5/AD5 |
| 3 | P1.7/CEX4 | 18 | P2.0/A8 | 33 | P0.4/AD4 |
| 4 | RST | 19 | P2.1/A9 | 34 | P0.3/AD3 |
| 5 | P3.0/RxD | 20 | P2.2/A10 | 35 | P0.2/AD2 |
| 6 | NC* | 21 | P2.3/A11 | 36 | P0.1/AD1 |
| 7 | P3.1/TxD | 22 | P2.4/A12 | 37 | P0.0/AD0 |
| 8 | P3.2/$\overline{INT0}$ | 23 | P2.5/A13 | 38 | $V_{CC}$ |
| 9 | P3.3/$\overline{INT1}$ | 24 | P2.6/A14 | 39 | NC* |
| 10 | P3.4/T0 | 25 | P2.7/A15 | 40 | P1.0/T2 |
| 11 | P3.5/T1 | 26 | $\overline{PSEN}$ | 41 | P1.1/T2EX |
| 12 | P3.6/$\overline{WR}$ | 27 | ALE | 42 | P1.2/ECI |
| 13 | P3.7/$\overline{RD}$ | 28 | NC* | 43 | P1.3/CEX0 |
| 14 | XTAL2 | 29 | $\overline{EA}$ | 44 | P1.4/CEX1 |
| 15 | XTAL1 | 30 | P0.7/AD7 | | |

* DO NOT CONNECT

SU00716B

## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER DIP | PIN NUMBER LCC | PIN NUMBER QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification and receives code bytes during EPROM programming. External pull-ups are required during program verification. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also receives the low-order address byte during program memory verification. Alternate functions include: |
| | 1 | 2 | 40 | I | **T2 (P1.0):** Timer/Counter 2 external count input/Clockout (See Programmable Clock-Out.) |
| | 2 | 3 | 41 | I | **T2EX (P1.1):** Timer/Counter 2 Reload/Capture/Direction Control |
| | 3 | 4 | 42 | I | **ECI (P1.2):** External Clock Input to the PCA |
| | 4 | 5 | 43 | I/O | **CEX0 (P1.3):** Capture/Compare External I/O for PCA module 0 |
| | 5 | 6 | 44 | I/O | **CEX1 (P1.4):** Capture/Compare External I/O for PCA module 1 |
| | 6 | 7 | 1 | I/O | **CEX2 (P1.5):** Capture/Compare External I/O for PCA module 2 |
| | 7 | 8 | 2 | I/O | **CEX3 (P1.6):** Capture/Compare External I/O for PCA module 3 |
| | 8 | 9 | 3 | I/O | **CEX4 (P1.7):** Capture/Compare External I/O for PCA module 4 |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. Some Port 2 pins receive the high order address bits during EPROM programming and verification. |

## PIN DESCRIPTIONS (Continued)

| MNEMONIC | PIN NUMBER DIP | PIN NUMBER LCC | PIN NUMBER QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{CC}$. |
| ALE/$\overline{PROG}$ | 30 | 33 | 27 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the 87C51FA/FB is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}/V_{PP}$ | 31 | 35 | 29 | I | **External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H to 1FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 1FFFH. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. If security bit 1 is programmed, $\overline{EA}$ will be internally latched on Reset. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTE:**
To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher than $V_{CC}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

### TIMER 2

This is a 16-bit up or down counter, which can be operated as either a timer or event counter. It can be operated in one of three different modes (autoreload, capture or as the baud rate generator for the UART).

In the autoreload mode the Timer can be set to count up or down by setting or clearing the bit DCEN in the T2CON Special Function Register. The SFR's RCAP2H and RCAP2L are used to reload the Timer upon overflow or a 1-to-0 transition on the T2EX input (P1.1).

In the Capture mode Timer 2 can either set TF2 and generate an interrupt or capture its value. To capture Timer 2 in response to a 1-to-0 transition on the T2EX input, the EXEN2 bit in the T2CON must be set. Timer 2 is then captured in SFR's RCAP2H and RCAP2L.

As the baud rate generator, Timer 2 is selected by setting TCLK and/or RCLK in T2CON. As the baud rate generator Timer 2 is incremented at $^1/_2$ the oscillator frequency.

### POWER OFF FLAG

The Power Off Flag (POF) is set by on-chip circuitry when the $V_{CC}$ level on the 87C51FA/FB rises from 0 to 5V. The POF bit can be set or cleared by software allowing a user to determine if the reset is the result of a power-on or a warm start after powerdown. The $V_{CC}$ level must remain above 3V for the POF to remain unaffected by the $V_{CC}$ level.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

### Reset

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up.

### Idle Mode

In the idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

### Power-Down Mode

To save even more power, a Power Down mode can be invoked by software. In this mode, the oscillator is stopped and the instruction that invoked Power Down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the Power Down mode is terminated.

On the 87C51FA/FB either a hardware reset or external interrupt can use an exit from Power Down. Reset redefines all the SFRs but does not change the on-chip RAM. An external interrupt allows both the SFRs and the on-chip RAM to retain their values.

To properly terminate Power Down the reset or external interrupt should not be executed before $V_{CC}$ is restored to its normal operating level and must be held active long enough for the oscillator to restart and stabilize (normally less than 10ms).

With an external interrupt, INT0 and INT1 must be enabled and configured as level-sensitive. Holding the pin low restarts the oscillator but bringing the pin back high completes the exit. Once the interrupt is serviced, the next instruction to be executed after RETI will be the one following the instruction that put the device into Power Down.

### Design Consideration

- When the idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal rest algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

### ONCE™ Mode

The ONCE ("On-Circuit Emulation") Mode facilitates testing and debugging of systems using the 87C51FA/FB without the 87C51FA/FB having to be removed from the circuit. The ONCE Mode is invoked by:

1. Pull ALE low while the device is in reset and $\overline{PSEN}$ is high;
2. Hold ALE low as RST is deactivated.

While the device is in ONCE Mode, the Port 0 pins go into a float state, and the other port pins and ALE and $\overline{PSEN}$ are weakly pulled high. The oscillator circuit remains active. While the 87C51FA/FB is in this mode, an emulator or test CPU can be used to drive the circuit. Normal operation is restored when a normal reset is applied.

### Programmable Clock-Out

The 87C51FA/FB has a new feature. A 50% duty cycle clock can be programmed to come out on P1.0. This pin, besides being a regular I/O pin, has two alternate functions. It can be programmed (1) to input the external clock for Timer/Counter 2 or (2) to output a 50% duty cycle clock ranging from 61Hz to 4MHz at a 16MHz operating frequency.

To configure the Timer/Counter 2 as a clock generator, bit C/T2 (in T2CON) must be cleared and bit T2OE in T2MOD must be set. Bit TR2 (T2CON.2) also must be set to start the timer.

The Clock-Out frequency depends on the oscillator frequency and the reload value of Timer 2 capture registers (RCAP2H, RCAP2L) as shown in this equation:

$$\frac{\text{Oscillator Frequency}}{4 \times (65536 - \text{RCAP2H, RCAP2L})}$$

In the Clock-Out mode, Timer 2 roll-overs will not generate an interrupt. This is similar to when it is used as a baud-rate generator. It is possible to use Timer 2 as a baud-rate generator and a clock generator simultaneously. Note, however, that the baud-rate and the Clock-Out frequency will be the same.

**Table 2. External Pin Status During Idle and Power-Down Mode**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

## Programmable Counter Array (PCA)

The Programmable Counter Array is a special Timer that has five 16-bit capture/compare modules associated with it. Each of the modules can be programmed to operate in one of four modes: rising and/or falling edge capture, software timer, high-speed output, or pulse width modulator. Each module has a pin associated with it in port 1. Module 0 is connected to P1.3(CEX0), module 1 to P1.4(CEX1), etc. The basic PCA configuration is shown in Figure 1.

The PCA timer is a common time base for all five modules and can be programmed to run at: 1/12 the oscillator frequency, 1/4 the oscillator frequency, the Timer 0 overflow, or the input on the ECI pin (P1.2). The timer count source is determined from the CPS1 and CPS0 bits in the CMOD SFR as follows (see Figure 4):

| CPS1 | CPS0 | PCA Timer Count Source |
|---|---|---|
| 0 | 0 | 1/12 oscillator frequency |
| 0 | 1 | 1/4 oscillator frequency |
| 1 | 0 | Timer 0 overflow |
| 1 | 1 | External Input at ECI pin |

In the CMOD SFR are three additional bits associated with the PCA. They are CIDL which allows the PCA to stop during idle mode, WDTE which enables or disables the watchdog function on module 4, and ECF which when set causes an interrupt and the PCA overflow flag, CF (in the CCON SFR) to be set when the PCA timer overflows. These functions are shown in Figure 2.

The watchdog timer function is implemented in module 4 as implemented in other parts that have a PCA that are available on the market. However, if a watchdog timer is required in the target application, it is recommended to use the hardware watchdog timer that is implemented on the 87C51FA/FB separately from the PCA (see Figure 12).

The CCON SFR contains the run control bit for the PCA and the flags for the PCA timer (CF) and each module (refer to Figure 5). To run the PCA the CR bit (CCON.6) must be set by software. The PCA is shut off by clearing this bit. The CF bit (CCON.7) is set when the PCA counter overflows and an interrupt will be generated if the ECF bit in the CMOD register is set, The CF bit can only be cleared by software. Bits 0 through 4 of the CCON register are the flags for the modules (bit 0 for module 0, bit 1 for module 1, etc.) and are set by hardware when either a match or a capture occurs. These flags also can only be cleared by software. The PCA interrupt system shown in Figure 3.

Each module in the PCA has a special function register associated with it. These registers are: CCAPM0 for module 0, CCAPM1 for module 1, etc. (see Figure 6). The registers contain the bits that control the mode in which each module will operate. The ECCF bit (CCAPMn.0 where n=0, 1, 2, 3, or 4 depending on the module) enables the CCF flag in the CCON SFR to generate an interrupt when a match or compare occurs in the associated module. PWM (CCAPMn.1) enables the pulse width modulation mode. The TOG bit (CCAPMn.2) when set causes the CEX output associated with the module to toggle when there is a match between the PCA counter and the module's capture/compare register. The match bit MAT (CCAPMn.3) when set will cause the CCFn bit in the CCON register to be set when there is a match between the PCA counter and the module's capture/compare register.

The next two bits CAPN (CCAPMn.4) and CAPP (CCAPMn.5) determine the edge that a capture input will be active on. The CAPN bit enables the negative edge, and the CAPP bit enables the positive edge. If both bits are set both edges will be enabled and a capture will occur for either transition. The last bit in the register ECOM (CCAPMn.6) when set enables the comparator function. Figure 7 shows the CCAPMn settings for the various PCA functions.

There are two additional registers associated with each of the PCA modules. They are CCAPnH and CCAPnL and these are the registers that store the 16-bit count when a capture occurs or a compare should occur. When a module is used in the PWM mode these registers are used to control the duty cycle of the output.

### PCA Capture Mode

To use one of the PCA modules in the capture mode either one or both of the CCAPM bits CAPN and CAPP for that module must be set. The external CEX input for the module (on port 1) is sampled for a transition. When a valid transition occurs the PCA hardware loads the value of the PCA counter registers (CH and CL) into the module's capture registers (CCAPnL and CCAPnH). If the CCFn bit for the module in the CCON SFR and the ECCFn bit in the CCAPMn SFR are set then an interrupt will be generated. Refer to Figure 8.

### 16-bit Software Timer Mode

The PCA modules can be used as software timers by setting both the ECOM and MAT bits in the modules CCAPMn register. The PCA timer will be compared to the module's capture registers and when a match occurs an interrupt will occur if the CCFn (CCON SFR) and the ECCFn (CCAPMn SFR) bits for the module are both set (see Figure 9).

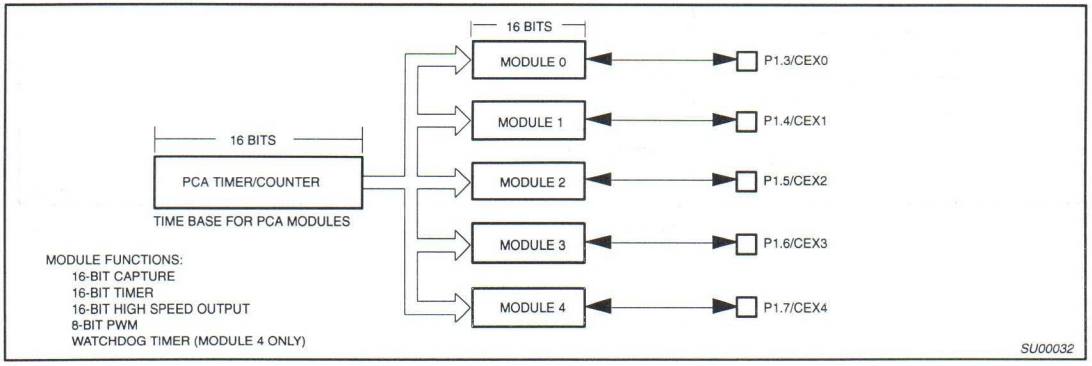



**Figure 1. Programmable Counter Array (PCA)**

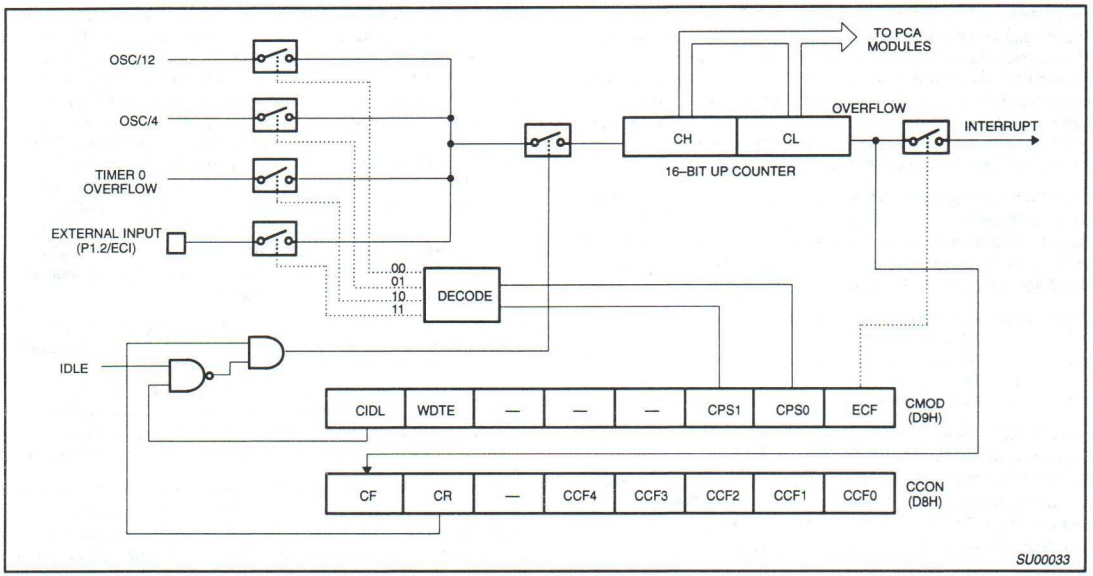



**Figure 2. PCA Timer/Counter**

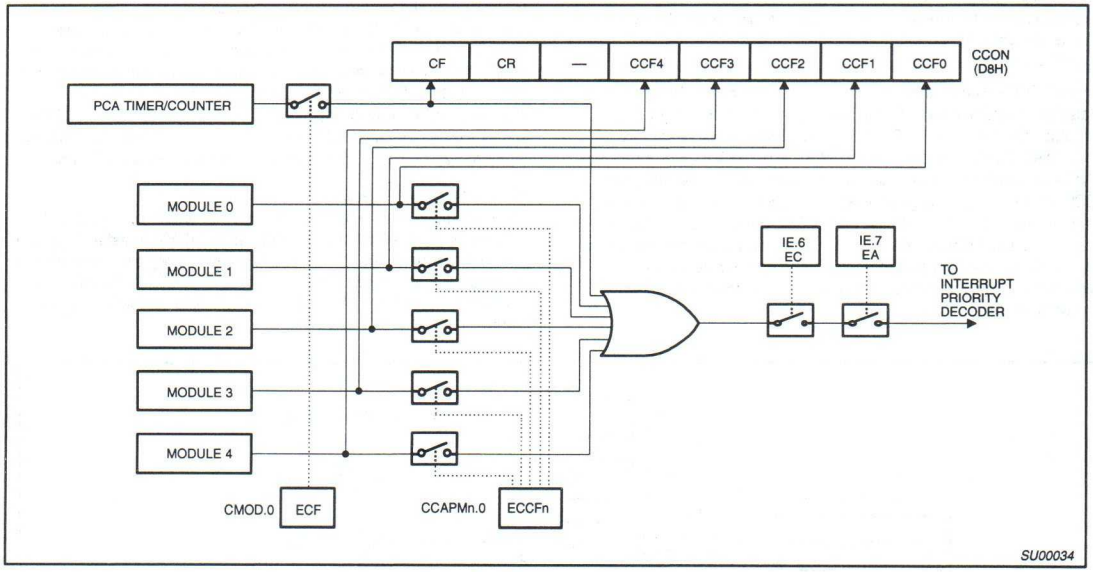



**Figure 3. PCA Interrupt System**

**CMOD** Address = OD9H Reset Value = 00XX X000B

Bit Addressable

| CIDL | WDTE | – | – | – | CPS1 | CPS0 | ECF |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Bit:

| Symbol | Function |
|---|---|
| **CIDL** | Counter Idle control: CIDL = 0 programs the PCA Counter to continue functioning during idle Mode. CIDL = 1 programs it to be gated off during idle. |
| **WDTE** | Watchdog Timer Enable: WDTE = 0 disables Watchdog Timer function on PCA Module 4. WDTE = 1 enables it. |
| – | Not implemented, reserved for future use.* |
| **CPS1** | PCA Count Pulse Select bit 1. |
| **CPS0** | PCA Count Pulse Select bit 0. |

| CPS1 | CPS0 | Selected PCA Input** | |
|---|---|---|---|
| 0 | 0 | 0 | Internal clock, $f_{OSC} \div 12$ |
| 0 | 1 | 1 | Internal clock, $f_{OSC} \div 4$ |
| 1 | 0 | 2 | Timer 0 overflow |
| 1 | 1 | 3 | External clock at ECI/P1.2 pin (max. rate = $f_{OSC} \div 8$) |

| Symbol | Function |
|---|---|
| **ECF** | PCA Enable Counter Overflow interrupt: ECF = 1 enables CF bit in CCON to generate an interrupt. ECF = 0 disables that function of CF. |

**NOTE:**
* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.
** $f_{OSC}$ = oscillator frequency

*SU00035*

**Figure 4. CMOD: PCA Counter Mode Register**

**CCON** Address = OD8H Reset Value = 00X0 0000B

Bit Addressable

| CF | CR | – | CCF4 | CCF3 | CCF2 | CCF1 | CCF0 |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Bit:

| Symbol | Function |
|---|---|
| **CF** | PCA Counter Overflow flag. Set by hardware when the counter rolls over. CF flags an interrupt if bit ECF in CMOD is set. CF may be set by either hardware or software but can only be cleared by software. |
| **CR** | PCA Counter Run control bit. Set by software to turn the PCA counter on. Must be cleared by software to turn the PCA counter off. |
| – | Not implemented, reserved for future use*. |
| **CCF4** | PCA Module 4 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF3** | PCA Module 3 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF2** | PCA Module 2 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF1** | PCA Module 1 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF0** | PCA Module 0 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |

**NOTE:**
* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

*SU00036*

**Figure 5. CCON: PCA Counter Control Register**

**CCAPMn** Address CCAPM0 0DAH
CCAPM1 0DBH
CCAPM2 0DCH
CCAPM3 0DDH
CCAPM4 0DEH

Reset Value = X000 0000B

Not Bit Addressable

| | – | ECOMn | CAPPn | CAPNn | MATn | TOGn | PWMn | ECCFn |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Symbol | Function |
|---|---|
| – | Not implemented, reserved for future use*. |
| **ECOMn** | Enable Comparator. ECOMn = 1 enables the comparator function. |
| **CAPPn** | Capture Positive, CAPPn = 1 enables positive edge capture. |
| **CAPNn** | Capture Negative, CAPNn = 1 enables negative edge capture. |
| **MATn** | Match. When MATn = 1, a match of the PCA counter with this module's compare/capture register causes the CCFn bit in CCON to be set, flagging an interrupt. |
| **TOGn** | Toggle. When TOGn = 1, a match of the PCA counter with this module's compare/capture register causes the CEXn pin to toggle. |
| **PWMn** | Pulse Width Modulation Mode. PWMn = 1 enables the CEXn pin to be used as a pulse width modulated output. |
| **ECCFn** | Enable CCF interrupt. Enables compare/capture flag CCFn in the CCON register to generate an interrupt. |

**NOTE:**
*User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

*SU00037*

**Figure 6. CCAPMn: PCA Modules Compare/Capture Registers**

| – | ECOMn | CAPPn | CAPNn | MATn | TOGn | PWMn | ECCFn | MODULE FUNCTION |
|---|---|---|---|---|---|---|---|---|
| X | 0 | 0 | 0 | 0 | 0 | 0 | 0 | No operation |
| X | X | 1 | 0 | 0 | 0 | 0 | X | 16-bit capture by a positive-edge trigger on CEXn |
| X | X | 0 | 1 | 0 | 0 | 0 | X | 16-bit capture by a negative trigger on CEXn |
| X | X | 1 | 1 | 0 | 0 | 0 | X | 16-bit capture by a transition on CEXn |
| X | 1 | 0 | 0 | 1 | 0 | 0 | X | 16-bit Software Timer |
| X | 1 | 0 | 0 | 1 | 1 | 0 | X | 16-bit High Speed Output |
| X | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 8-bit PWM |
| X | 1 | 0 | 0 | 1 | X | 0 | X | Watchdog Timer |

**Figure 7. PCA Module Modes (CCAPMn Register)**

**High Speed Output Mode**
In this mode the CEX output (on port 1) associated with the PCA module will toggle each time a match occurs between the PCA counter and the module's capture registers. To activate this mode the TOG, MAT, and ECOM bits in the module's CCAPMn SFR must be set (see Figure 10).

**Pulse Width Modulator Mode**
All of the PCA modules can be used as PWM outputs. Figure 11 shows the PWM function. The frequency of the output depends on the source for the PCA timer. All of the modules will have the same frequency of output because they all share the PCA timer. The duty cycle of each module is independently variable using the module's capture register CCAPLn. When the value of the PCA CL SFR is less than the value in the module's CCAPLn SFR the output will be low, when it is equal to or greater than the output will be high. When CL overflows from FF to 00, CCAPLn is reloaded with the value in CCAPHn then allows updating the PWM without glitches. The PWM and ECOM bits in the module's CCAPMn register must be set to enable the PWM mode.

## Enhanced UART

The UART operates in all of the usual modes that are described in the first section of this book for the 80C51. In addition the UART can perform framing error detect by looking for missing stop bits, and automatic address recognition. The 87C51FA/FB UART also fully supports multiprocessor communication as does the standard 80C51 UART.

When used for framing error detect the UART looks for missing stop bits in the communication. A missing bit will set the FE bit in the SCON register. The FE bit shares the SCON.7 bit with SM0 and the function of SCON.7 is determined by PCON.6 (SMOD0) (see Figure 13). If SMOD0 is set then SCON.7 functions as FE. SCON.7 functions as SM0 when SMOD0 is cleared. When used as FE SCON.7 can only be cleared by software. Refer to Figure 14.

**Automatic Address Recognition**
Automatic Address Recognition is a feature which allows the UART to recognize certain addresses in the serial bit stream by using hardware to make the comparisons. This feature saves a great deal

of software overhead by eliminating the need for the software to examine every serial address which passes by the serial port. This feature is enabled by setting the SM2 bit in SCON. In the 9 bit UART modes, mode 2 and mode 3, the Receive Interrupt flag (RI) will be automatically set when the received byte contains either the "Given" address or the "Broadcast" address. The 9 bit mode requires that the 9th information bit is a 1 to indicate that the received information is an address and not data. Automatic address recognition is shown in Figure 15.

The 8 bit mode is called Mode 1. In this mode the RI flag will be set if SM2 is enabled and the information received has a valid stop bit following the 8 address bits and the information is either a Given or Broadcast address.

Mode 0 is the Shift Register mode and SM2 is ignored.

Using the Automatic Address Recognition feature allows a master to selectively communicate with one or more slaves by invoking the Given slave address or addresses. All of the slaves may be contacted by using the Broadcast address. Two special Function Registers are used to define the slave's address, SADDR, and the address mask, SADEN. SADEN is used to define which bits in the SADDR are to b used and which bits are "don't care". The SADEN mask can be logically ANDed with the SADDR to create the "lGiven" address which the master will use for addressing each of the slaves. Use of the Given address allows multiple slaves to be recognized while excluding others. The following examples will help to show the versatility of this scheme:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1101 |
| | Given | = | 1100 00X0 |
| Slave 1 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1110 |
| | Given | = | 1100 000X |

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires a 0 in bit 0 and it ignores bit 1. Slave 1 requires a 0 in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010 since slave 1 requires a 0 in bit 1. A unique address for slave 1 would be 1100 0001 since a 1 in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 1100 0000.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1001 |
| | Given | = | 1100 0XX0 |
| Slave 1 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1010 |
| | Given | = | 1110 0X0X |
| Slave 2 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1100 |
| | Given | = | 1110 00XX |

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 and 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary to make bit 2 = 1 to exclude slave 2.

The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are teated as don't-cares. In most cases, interpreting the don't-cares as ones, the broadcast address will be FF hexadecimal.

Upon reset SADDR (SFR address 0A9H) and SADEN (SFR address 0B9H) are loaded with 0s. This produces a given address of all "don't cares" as well as a Broadcast address of all "don't cares". This effectively disables the Automatic Addressing mode and allows the microcontroller to use standard 80C51 type UART drivers which do not make use of this feature.

## Reduced EMI Mode

The AO bit (AUXR.0) in the AUXR register when set disables the ALE output.

## 87C51FA/FB Reduced EMI Mode

**AUXR (0X8E)**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | — | AO |

AO: Turns off ALE output.

**Interrupt Enable (IE) Register**

| | | |
|---|---|---|
| EA | IE.7 | enable all interrupts |
| EC | IE.6 | enable PCA interrupt |
| ET2 | IE.5 | enable Timer 2 interrupt |
| ES | IE.4 | enable Serial I/O interrupt |
| ET1 | IE.3 | enable Timer 1 interrupt |
| EX1 | IE.2 | enable External interrupt 1 |
| ET0 | IE.1 | enable Timer 0 interrupt |
| EX0 | IE.0 | enable External interrupt 0 |

**Interrupt Priority (IP) Register**

| | | |
|---|---|---|
| | IP.7 | reserved |
| PPC | IP.6 | PCA interrupt priority |
| PT2 | IP.5 | Timer 2 interrupt priority |
| PS | IP.4 | Serial I/O interrupt priority |
| PT1 | IP.3 | Timer 1 interrupt priority |
| PX1 | IP.2 | External interrupt 1 priority |
| PT0 | IP.1 | Timer 0 interrupt priority |
| PX0 | IP.0 | External interrupt 0 priority |

| Priority | Source | Flag | Vector |
|---|---|---|---|
| 1 | INT0 | IE0 | 03H highest priority |
| 2 | Timer 0 | TF0 | 0BH |
| 3 | INT1 | IE1 | 13H |
| 4 | Timer 1 | TF1 | 1BH |
| 5 | PCA | CF,CCFn | 33H |
| 6 | Serial I/O | RI,TI | 23H |
| 7 | Timer 2 | TF2/EXF2 | 2BH lowest priority |

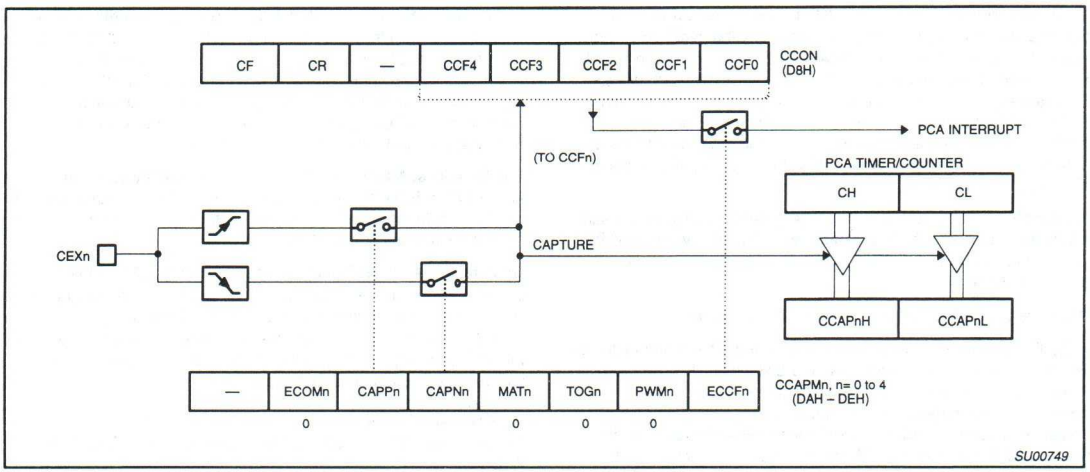



**Figure 8. PCA Capture Mode**

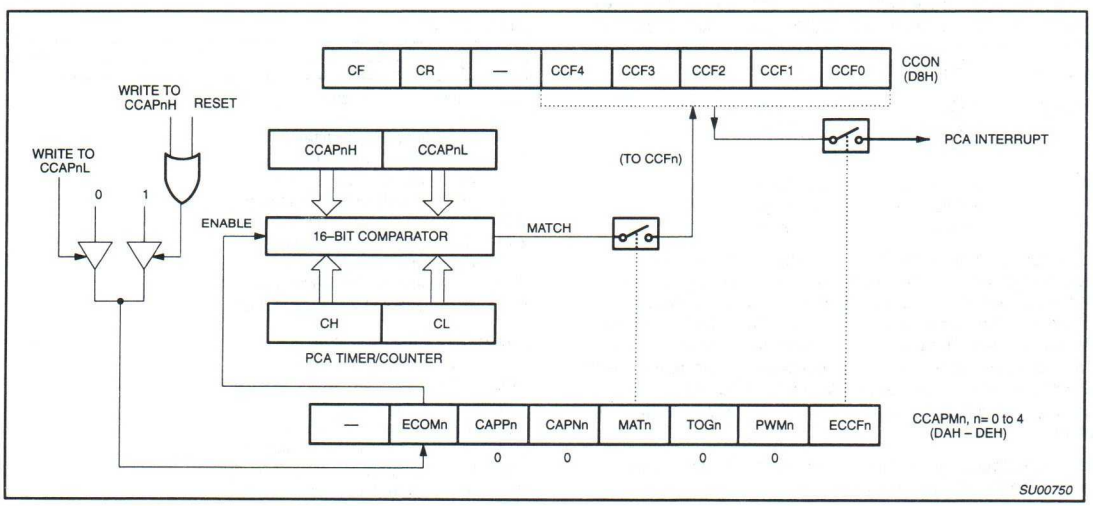



**Figure 9. PCA Compare Mode**

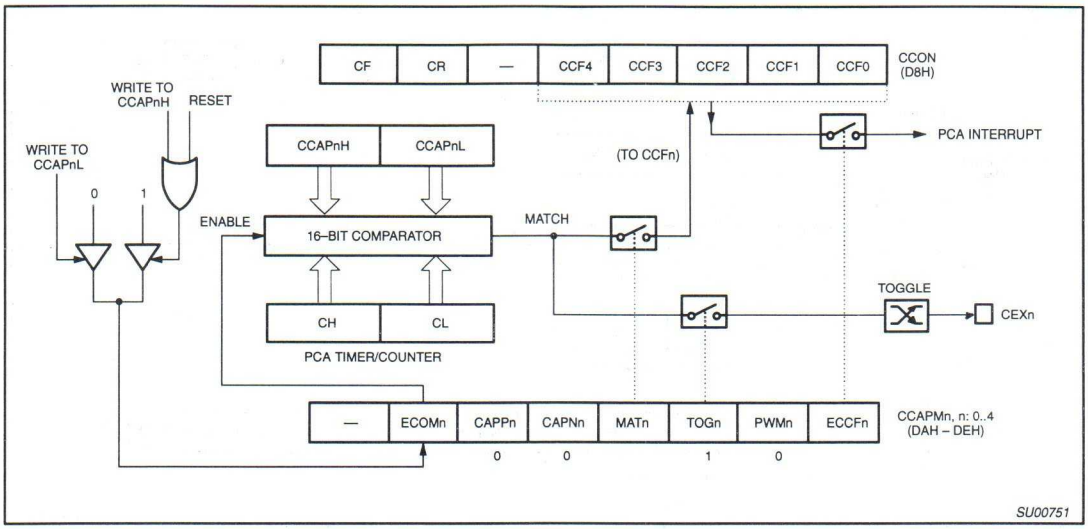



**Figure 10. PCA High Speed Output Mode**

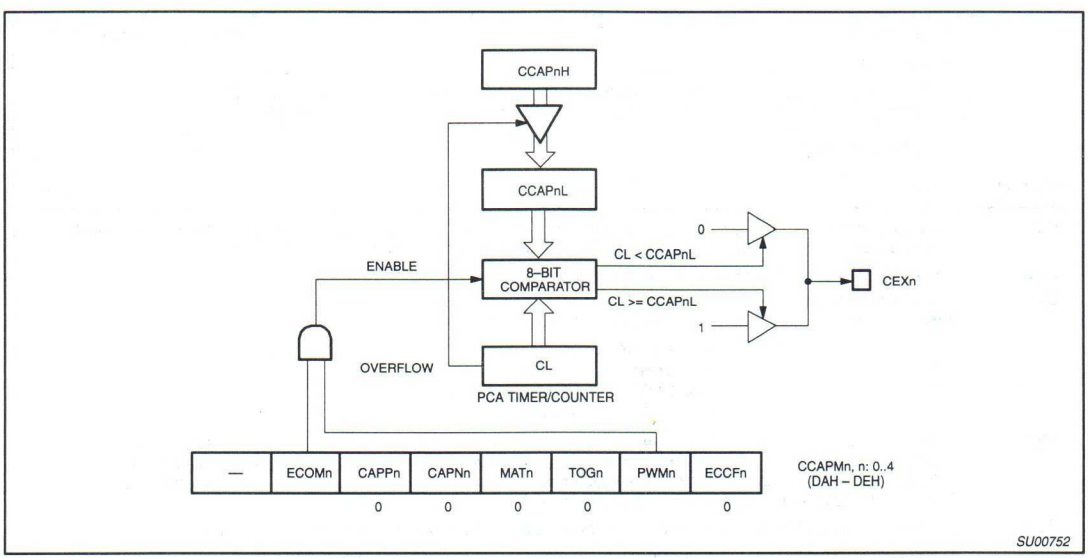



**Figure 11. PCA PWM Mode**

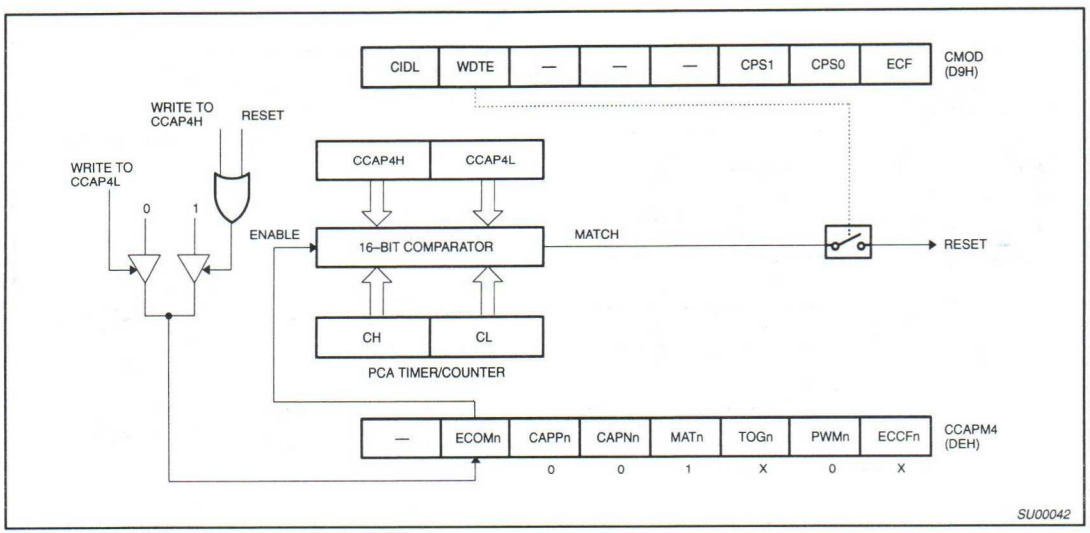



**Figure 12. PCA Watchdog Timer**

**SCON** Address = 98H — Reset Value = 0000 0000B

Bit Addressable

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | **SM0/FE** | **SM1** | **SM2** | **REN** | **TB8** | **RB8** | **TI** | **RI** |

(SMOD0 = 0/1)*

| Symbol | Function |
|---|---|
| **FE** | Framing Error bit. This bit is set by the receiver when an invalid stop bit is detected. The FE bit is not cleared by valid frames but should be cleared by software. The SMOD0 bit must be set to enable access to the FE bit. |
| **SM0** | Serial Port Mode Bit 0, (SMOD0 must = 0 to access bit SM0) |
| **SM1** | Serial Port Mode Bit 1 |

| **SM0** | **SM1** | **Mode** | **Description** | **Baud Rate**** |
|---|---|---|---|---|
| 0 | 0 | 0 | shift register | $f_{OSC}/12$ |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{OSC}/64$ or $f_{OSC}/32$ |
| 1 | 1 | 3 | 9-bit UART | variable |

| | |
|---|---|
| **SM2** | Enables the Automatic Address Recognition feature in Modes 2 or 3. If SM2 = 1 then RI will not be set unless the received 9th data bit (RB8) is 1, indicating an address, and the received byte is a Given or Broadcast Address. In Mode 1, if SM2 = 1 then RI will not be activated unless a valid stop bit was received, and the received byte is a Given or Broadcast Address. In Mode 0, SM2 should be 0. |
| **REN** | Enables serial reception. Set by software to enable reception. Clear by software to disable reception. |
| **TB8** | The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired. |
| **RB8** | In modes 2 and 3, the 9th data bit that was received. In Mode 1, if SM2 = 0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used. |
| **TI** | Transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software. |
| **RI** | Receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software. |

**NOTE:**
*SMOD0 is located at PCON6.
**$f_{OSC}$ = oscillator frequency

SU00043

**Figure 13. SCON: Serial Port Control Register**

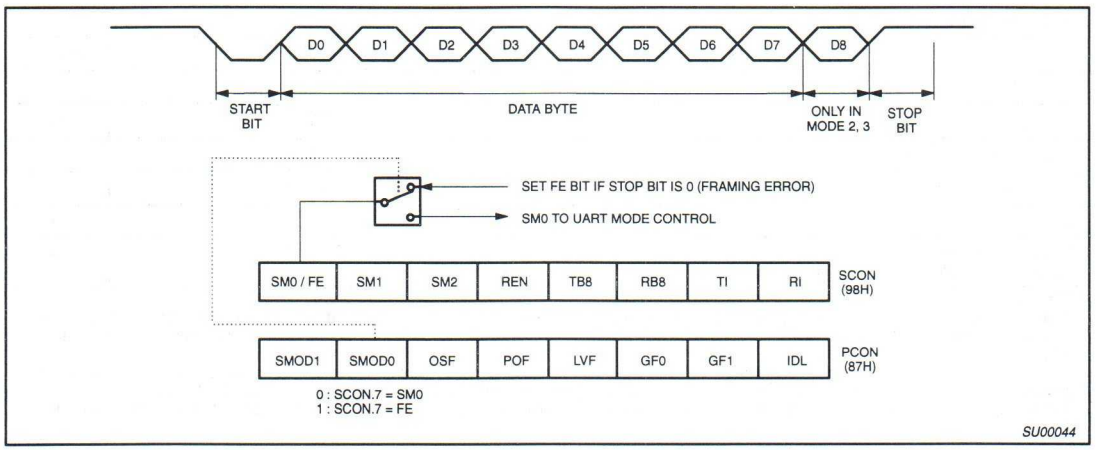



**Figure 14. UART Framing Error Detection**

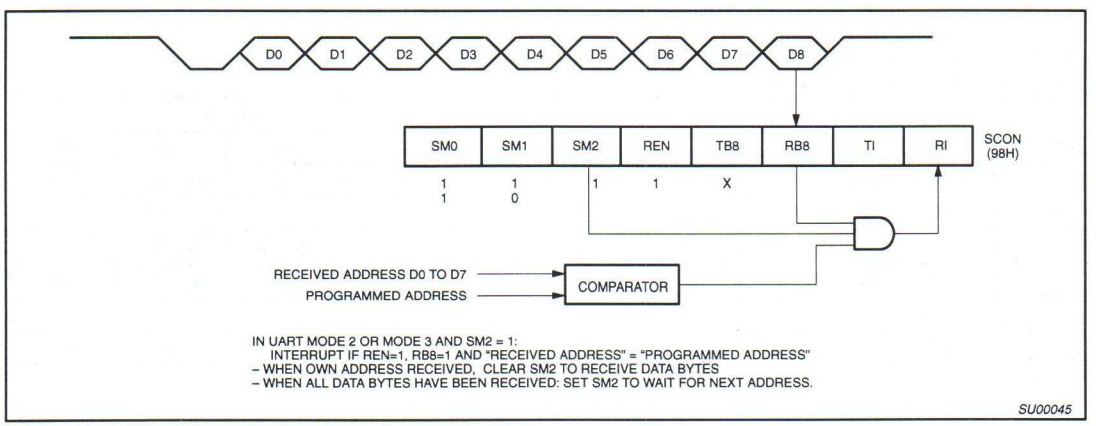



**Figure 15. UART Multiprocessor Communication, Automatic Address Recognition**

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | 0 to +70 or –40 to +85 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Maximum $I_{OL}$ per I/O pin | 15 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.5 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

**Electrical Deviations from Commercial Specifications for Extended Temperature Range**

DC and AC parameters not included here are the same as in the commercial temperature range table.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$ | | –0.5 | $0.2V_{CC}$–0.15 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | | 0 | $0.2V_{CC}$–0.35 | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST | | $0.2V_{CC}$+1 | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage to XTAL1, RST | | $0.7V_{CC}$+0.1 | $V_{CC}$+0.5 | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3 | $V_{IN}$ = 0.45V | | –75 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3 | $V_{IN}$ = 2.0V | | –750 | µA |

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$[7] | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$[7] | | 0 | | $0.2V_{CC}$–0.3 | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST[7] | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST[7] | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3[9] | $I_{OL}$ = 1.6mA[2] | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$[9] | $I_{OL}$ = 3.2mA[2] | | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3, ALE, $\overline{PSEN}$[3] | $I_{OH}$ = –30µA | $V_{CC}$ – 0.7 | | | V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode), ALE[10], $\overline{PSEN}$[3] | $I_{OH}$ = –3.2mA | $V_{CC}$ – 0.7 | | | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3[7] | $V_{IN}$ = 0.45V | | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3[7] | See note 4 | | | –650 | µA |
| $I_{LI}$ | Input leakage current, port 0 | 0.45 $V_{IN}$ < $V_{CC}$ – 0.3 | | | ±10 | µA |
| $I_{CC}$ | Power supply current:<br>Active mode @ 16MHz[5]<br>Idle mode @ 16MHz<br>Power-down mode<br>$T_{amb}$ = 0 to +70°C<br>$T_{amb}$ = –40 to +85°C | See note 6 | | <br>15<br>3<br><br>10 | <br>32<br>5<br><br>50<br>75 | <br>mA<br>mA<br><br>µA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | 50 | | 225 | kΩ |
| $C_{IO}$ | Pin capacitance[11] (except $\overline{EA}$) | | | | 15 | pF |

**NOTES:**

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
5. $I_{CCMAX}$ at other frequencies is given by: Active mode: $I_{CCMAX}$ = 1.50 × FREQ + 8; Idle mode: $I_{CCMAX}$ = 0.14 × FREQ +2.31, where FREQ is the external oscillator frequency in MHz. $I_{CCMAX}$ is given in mA. See Figure 23.
6. See Figures 24 through 27 for $I_{CC}$ test conditions.
7. These values apply only to $T_{amb}$ = 0°C to +70°C. For $T_{amb}$ = –40°C to +85°C, see table on previous page.
8. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
9. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 15mA (*NOTE: This is 85°C specification.)
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 71mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
10. ALE is tested to $V_{OH1}$, except when ALE is off then $V_{OH}$ is the voltage specification.
11. Pin capacitance is less than 25pF. Pin capacitance of ceramic package is less than 15pF (except $\overline{EA}$, it is 25pF).

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 16 | Oscillator frequency –4, –5 | | | 3.5 | 16 | MHz |
| | | –A, –B | | | 3.5 | 24 | MHz |
| $t_{LHLL}$ | 16 | ALE pulse width | 85 | | $2t_{CLCL}$–40 | | ns |
| $t_{AVLL}$ | 16 | Address valid to ALE low | 22 | | $t_{CLCL}$–40 | | ns |
| $t_{LLAX}$ | 16 | Address hold after ALE low | 32 | | $t_{CLCL}$–30 | | ns |
| $t_{LLIV}$ | 16 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}$–100 | ns |
| $t_{LLPL}$ | 16 | ALE low to $\overline{PSEN}$ low | 32 | | $t_{CLCL}$–30 | | ns |
| $t_{PLPH}$ | 16 | $\overline{PSEN}$ pulse width | 142 | | $3t_{CLCL}$–45 | | ns |
| $t_{PLIV}$[4] | 16 | $\overline{PSEN}$ low to valid instruction in | | 82 | | $3t_{CLCL}$–105 | ns |
| $t_{PXIX}$ | 16 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 16 | Input instruction float after $\overline{PSEN}$ | | 37 | | $t_{CLCL}$–25 | ns |
| $t_{AVIV}$ | 16 | Address to valid instruction in | | 207 | | $5t_{CLCL}$–105 | ns |
| $t_{PLAZ}$ | 16 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 17, 18 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}$–100 | | ns |
| $t_{WLWH}$ | 17, 18 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}$–100 | | ns |
| $t_{RLDV}$ | 17, 18 | $\overline{RD}$ low to valid data in | | 147 | | $5t_{CLCL}$–165 | ns |
| $t_{RHDX}$ | 17, 18 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 17, 18 | Data float after $\overline{RD}$ | | 65 | | $2t_{CLCL}$–60 | ns |
| $t_{LLDV}$ | 17, 18 | ALE low to valid data in | | 350 | | $8t_{CLCL}$–150 | ns |
| $t_{AVDV}$ | 17, 18 | Address to valid data in | | 397 | | $9t_{CLCL}$–165 | ns |
| $t_{LLWL}$ | 17, 18 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 137 | 237 | $3t_{CLCL}$–50 | $3t_{CLCL}$+50 | ns |
| $t_{AVWL}$ | 17, 18 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 122 | | $4t_{CLCL}$–130 | | ns |
| $t_{QVWX}$ | 17, 18 | Data valid to $\overline{WR}$ transition | 13 | | $t_{CLCL}$–50 | | ns |
| $t_{WHQX}$ | 17, 18 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}$–50 | | ns |
| $t_{QVWH}$ | 18 | Data valid to $\overline{WR}$ high | 287 | | $7t_{CLCL}$–150 | | ns |
| $t_{RLAZ}$ | 17, 18 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 17, 18 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}$–40 | $t_{CLCL}$+40 | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 20 | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | 20 | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | 20 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 20 | Fall time | | 20 | | 20 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 19 | Serial port clock cycle time | 750 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 19 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}$–133 | | ns |
| $t_{XHQX}$ | 19 | Output data hold after clock rising edge | 8 | | $2t_{CLCL}$–117 | | ns |
| $t_{XHDX}$ | 19 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 19 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}$–133 | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 87C51FA/FB to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.
4. See Application Note AN457.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 24MHz CLOCK | | VARIABLE CLOCK[4] | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 16 | Oscillator frequency<br>Speed versions : A, B (24MHz) | 3.5 | 24 | 3.5 | 24 | MHz |
| $t_{LHLL}$ | 16 | ALE pulse width | 43 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 16 | Address valid to ALE low | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{LLAX}$ | 16 | Address hold after ALE low | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{LLIV}$ | 16 | ALE low to valid instruction in | | 102 | | $4t_{CLCL}-65$ | ns |
| $t_{LLPL}$ | 16 | ALE low to $\overline{PSEN}$ low | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{PLPH}$ | 16 | $\overline{PSEN}$ pulse width | 80 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 16 | $\overline{PSEN}$ low to valid instruction in | | 65 | | $3t_{CLCL}-60$ | ns |
| $t_{PXIX}$ | 16 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 16 | Input instruction float after $\overline{PSEN}$ | | 17 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 16 | Address to valid instruction in | | 128 | | $5t_{CLCL}-80$ | ns |
| $t_{PLAZ}$ | 16 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 17, 18 | $\overline{RD}$ pulse width | 150 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 17, 18 | $\overline{WR}$ pulse width | 150 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 17, 18 | $\overline{RD}$ low to valid data in | | 118 | | $5t_{CLCL}-90$ | ns |
| $t_{RHDX}$ | 17, 18 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 17, 18 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-28$ | ns |
| $t_{LLDV}$ | 17, 18 | ALE low to valid data in | | 183 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 17, 18 | Address to valid data in | | 210 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 17, 18 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 75 | 175 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 17, 18 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 92 | | $4t_{CLCL}-75$ | | ns |
| $t_{QVWX}$ | 17, 18 | Data valid to $\overline{WR}$ transition | 12 | | $t_{CLCL}-30$ | | ns |
| $t_{WHQX}$ | 17, 18 | Data hold after $\overline{WR}$ | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{QVWH}$ | 18 | Data valid to $\overline{WR}$ high | 162 | | $7t_{CLCL}-130$ | | ns |
| $t_{RLAZ}$ | 17, 18 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 17, 18 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 17 | 67 | $t_{CLCL}-25$ | $t_{CLCL}+25$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 20 | High time | 17 | | 17 | $t_{CLCL}-t_{CLCX}$ | ns |
| $t_{CLCX}$ | 20 | Low time | 17 | | 17 | $t_{CLCL}-t_{CHCX}$ | ns |
| $t_{CLCH}$ | 20 | Rise time | | 5 | | 5 | ns |
| $t_{CHCL}$ | 20 | Fall time | | 5 | | 5 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 19 | Serial port clock cycle time | 505 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 19 | Output data setup to clock rising edge | 283 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 19 | Output data hold after clock rising edge | 3 | | $2t_{CLCL}-80$ | | ns |
| $t_{XHDX}$ | 19 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 19 | Clock rising edge to input data valid | | 283 | | $10t_{CLCL}-133$ | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 87C51FA/FB to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.
4. Variable clock is specified for oscillator frequencies greater than 16MHz to 33MHz. For frequencies equal or less than 16MHz, see 16MHz "AC Electrial Characteristics", page 3-80.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{PSEN}$ low.

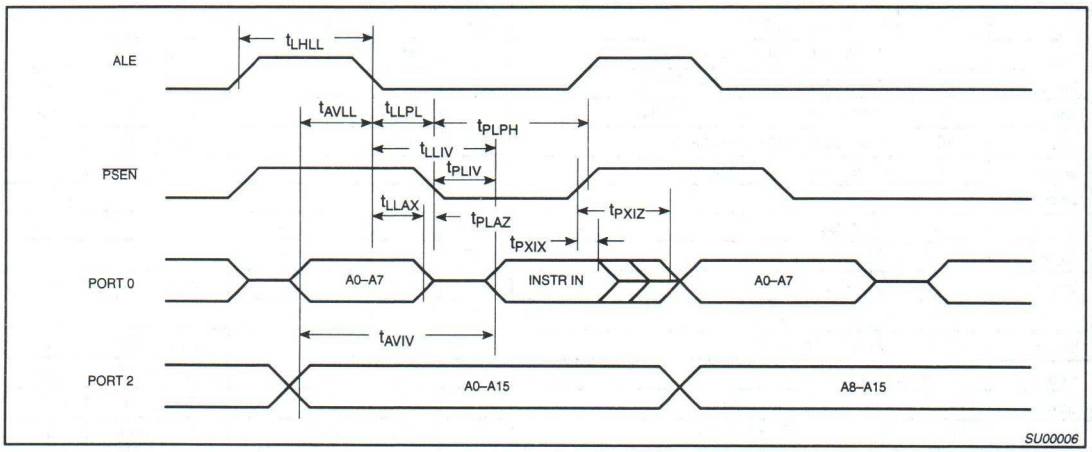



**Figure 16. External Program Memory Read Cycle**

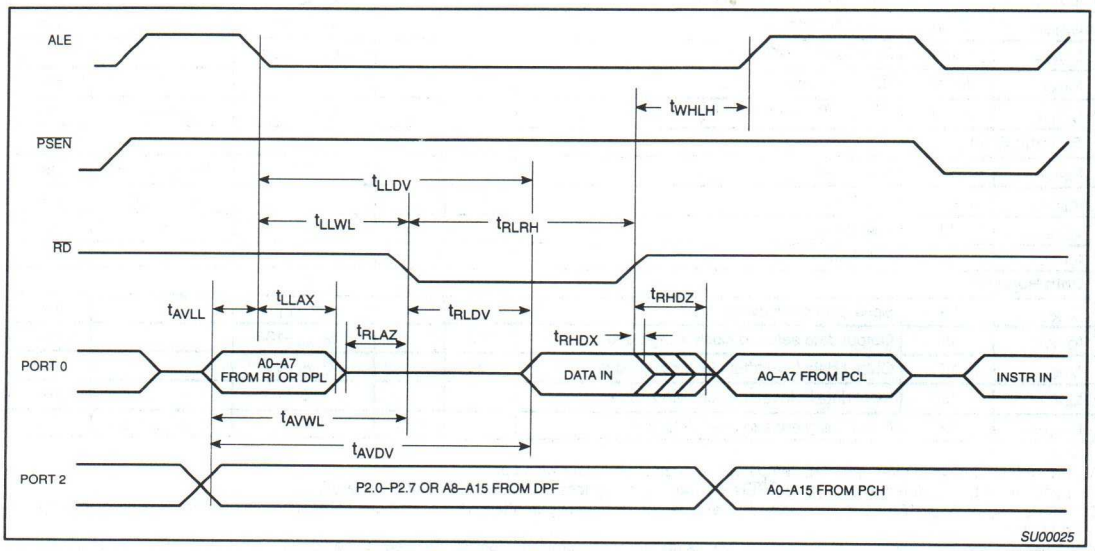



**Figure 17. External Data Memory Read Cycle**

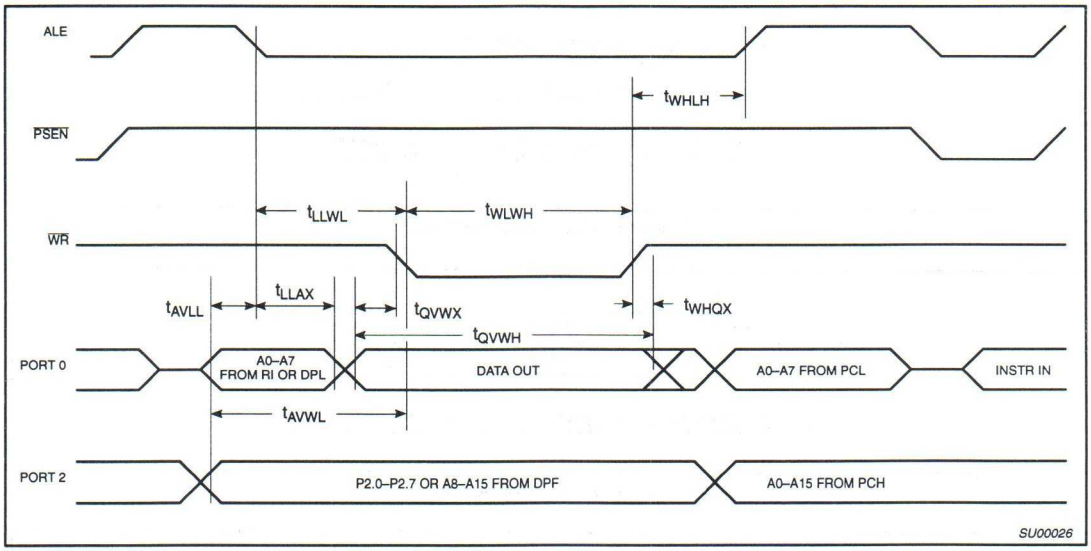



**Figure 18. External Data Memory Write Cycle**

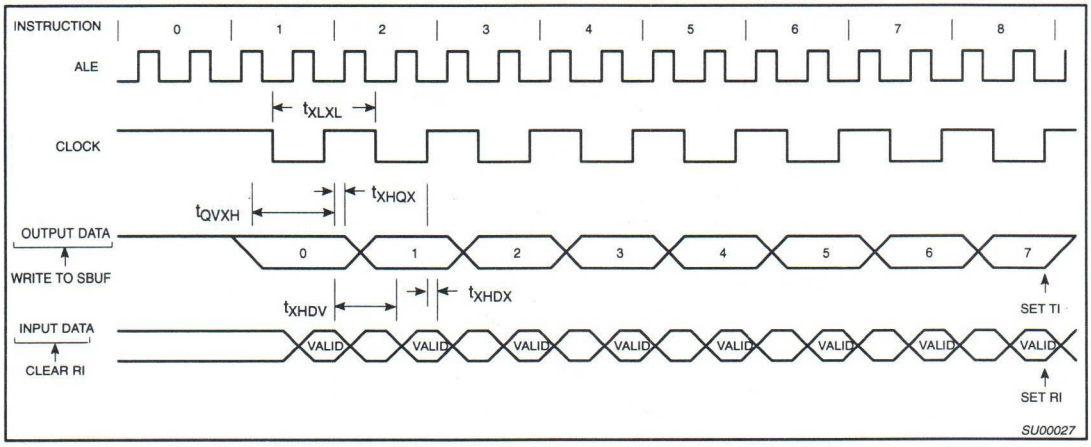



**Figure 19. Shift Register Mode Timing**

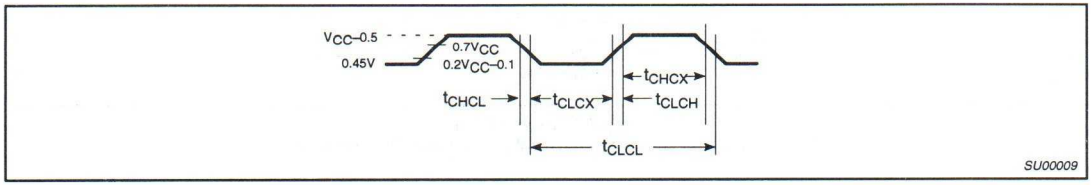
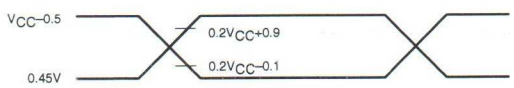


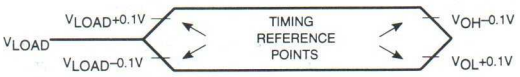

**Figure 20. External Clock Drive**




NOTE:
AC inputs during testing are driven at $V_{CC}$ –0.5 for a logic '1' and 0.45V for a logic '0'.
Timing measurements are made at $V_{IH}$ min for a logic '1' and $V_{IL}$ max for a logic '0'.

SU00010

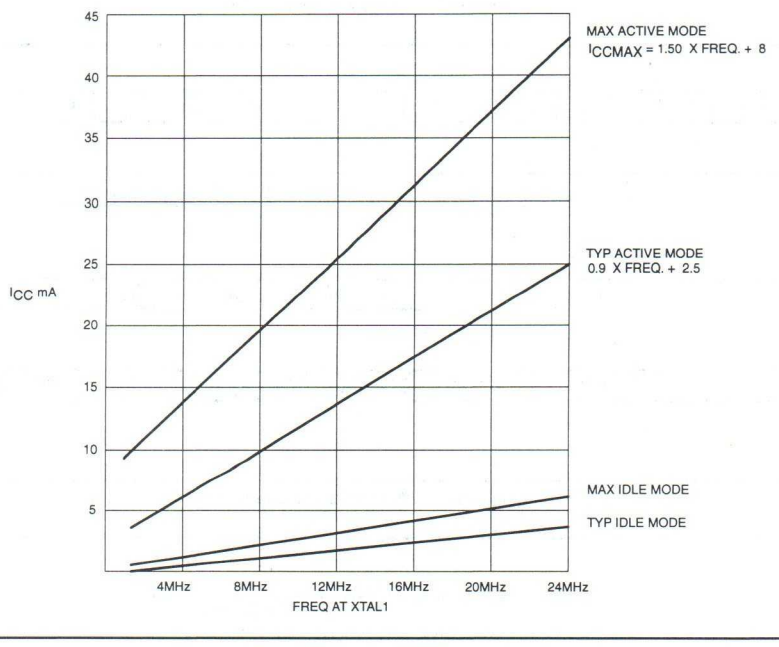
**Figure 21. AC Testing Input/Output**




NOTE:
For timing purposes, a port is no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded $V_{OH}/V_{OL}$ level occurs. $I_{OH}/I_{OL} \geq \pm 20$mA.

SU00011

**Figure 22. Float Waveform**




SU00046

**Figure 23. $I_{CC}$ vs. FREQ**
**Valid only within frequency specifications of the device under test**

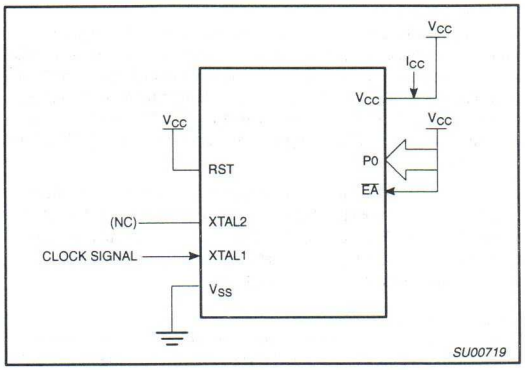



Figure 24. $I_{CC}$ Test Condition, Active Mode
All other pins are disconnected

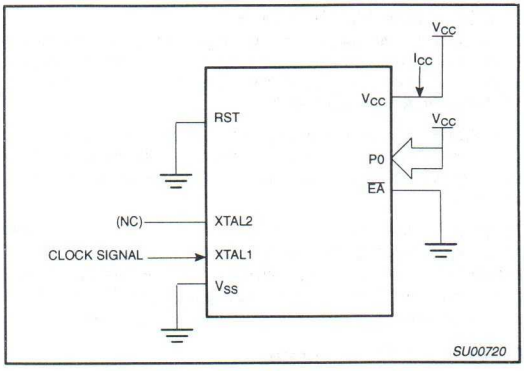



Figure 25. $I_{CC}$ Test Condition, Idle Mode
All other pins are disconnected

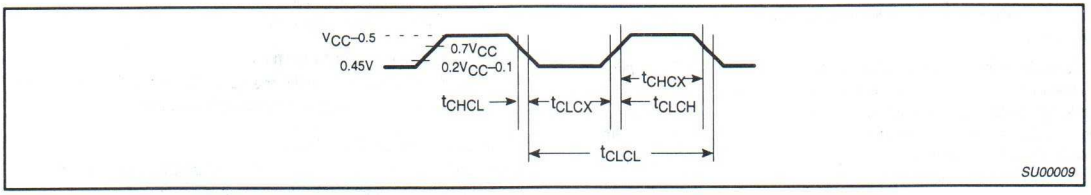



Figure 26. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes
$t_{CLCH} = t_{CHCL} = 5ns$

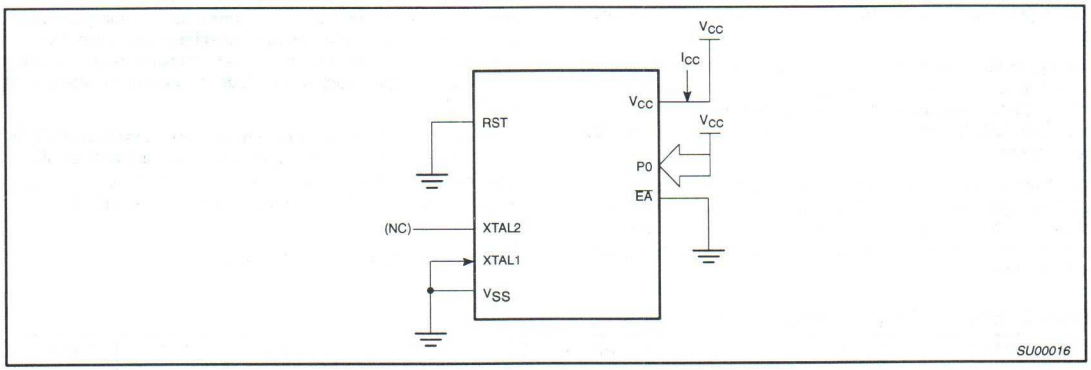



Figure 27. $I_{CC}$ Test Condition, Power Down Mode
All other pins are disconnected. $V_{CC}$ = 2V to 5.5V

## EPROM CHARACTERISTICS

The 87C51FA/FB is programmed by using a modified Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C51FA/FB contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C51FA/FB manufactured by Philips.

Table 3 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the security bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 28 and 29. Figure 30 shows the circuit configuration for normal program memory verification.

### Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 28. Note that the 87C51FA/FB is running with a 3.5MHz to 12MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1 and 2, as shown in Figure 28. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 3 specified in Table 3 are held at the 'Program Code Data' levels indicated in Table 3. The ALE/$\overline{PROG}$ is pulsed low 15 to 25 times as shown in Figure 29.

To program the encryption table, repeat the 15 to 25 pulse programming sequence for addresses 0 through 1FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the security bits, repeat the 15 to 25 pulse programming sequence using the 'Pgm Security Bit' levels. After one security bit is programmed, further programming of the code memory and encryption table is disabled. However, the other security bit can still be programmed.

Note that the $\overline{EA}$/$V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

#### Program Verification

If security bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 1 and 2 as shown in Figure 30. The other pins are held at the 'Verify Code Data' levels indicated in Table 3. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

#### Reading the Signature Bytes

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:

(030H) = 15H indicates manufactured by Philips
(031H) = B1H indicates 87C51FA
B2H indicates 87C51FB

### Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 3, and which satisfies the timing specifications, is suitable.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-s/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

### Table 3. EPROM Programming Modes

| MODE | RST | $\overline{PSEN}$ | ALE/$\overline{PROG}$ | $\overline{EA}$/$V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm security bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm security bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V±10% during programming and verification.

* ALE/$\overline{PROG}$ receives 15 to 25 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 50µs to 100µs and high for a minimum of 10µs.

™Trademark phrase of Intel Corporation.

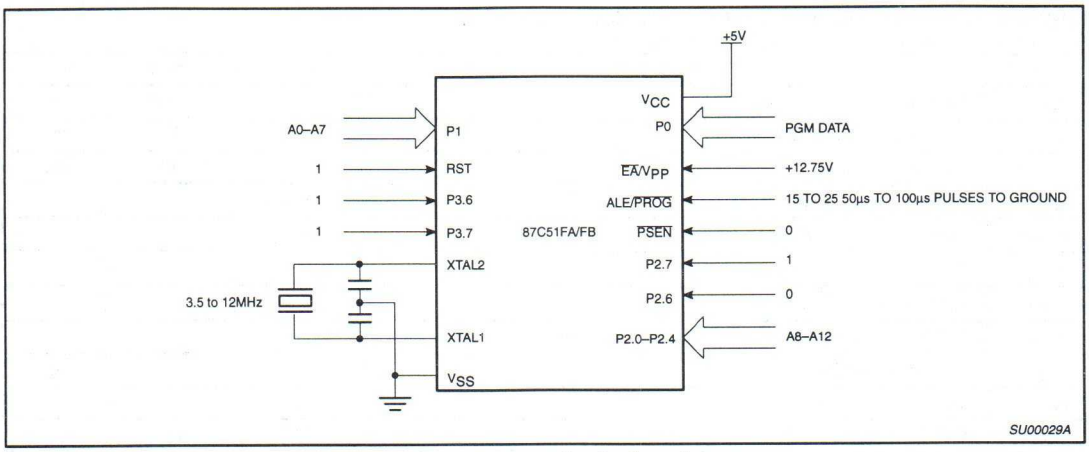



**Figure 28. Programming Configuration**

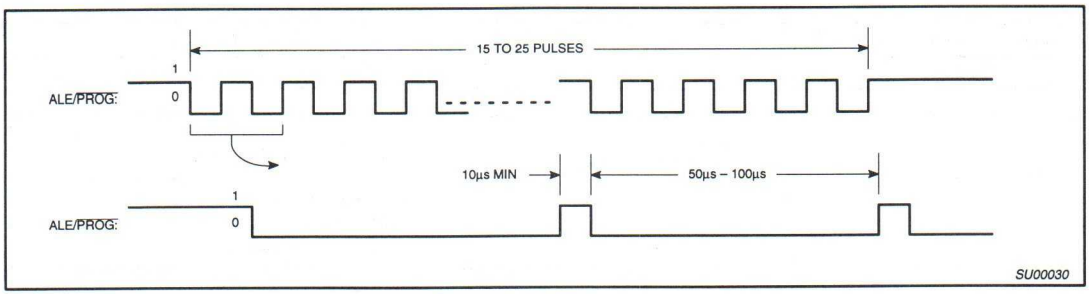



**Figure 29. PROG Waveform**

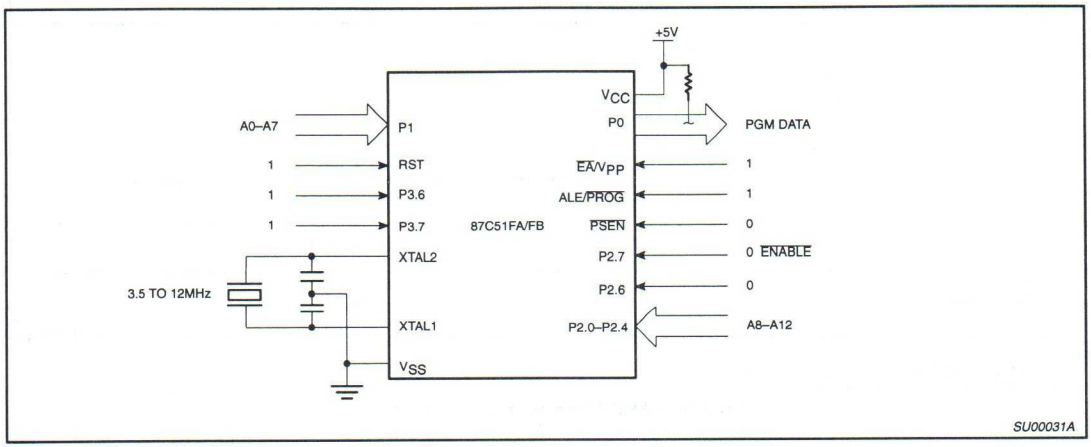
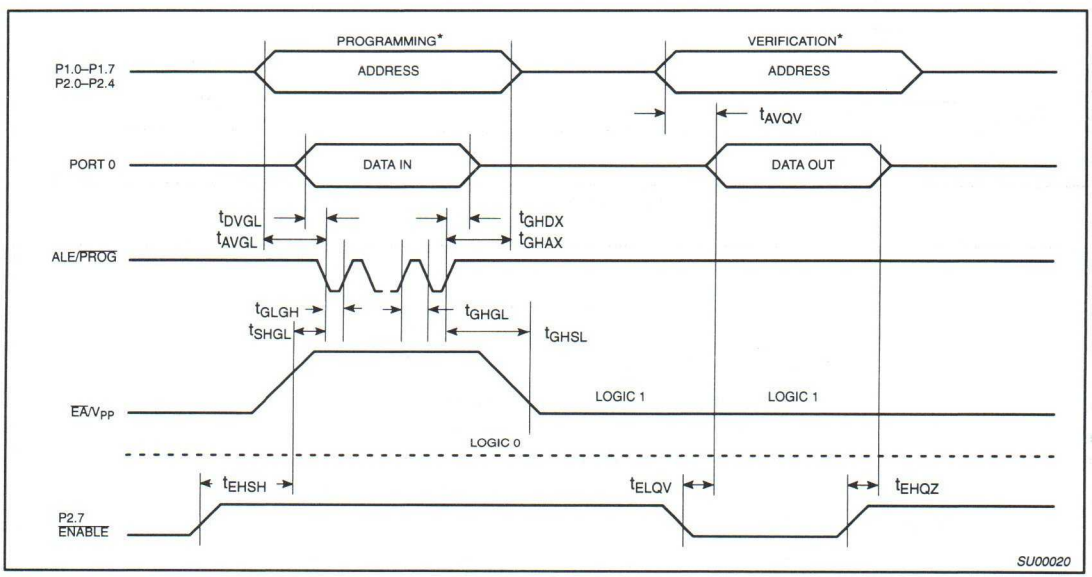



**Figure 30. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (See Figure 31)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 50 | 100 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




**NOTE:**

* FOR PROGRAMMING VERIFICATION SEE FIGURE 28.
FOR VERIFICATION CONDITIONS SEE FIGURE 30.

**Figure 31. EPROM Programming and Verification**

# CMOS single-chip 8-bit microcontroller 87C51FC

## DESCRIPTION

The 87C51FC Single-Chip 8-Bit Microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 8XC51FC has the same instruction set as the 80C51.

This device provides architectural enhancements that make it applicable in a variety of applications for general control systems. The 87C51FC contains 32k × 8 EPROM memory, a volatile 256 × 8 read/write data memory, four 8-bit I/O ports, three 16-bit timer/event counters, a Programmable Counter Array (PCA), a multi-source, four-priority-level, nested interrupt structure, an enhanced UART and on-chip oscillator and timing circuits. For systems that require extra capability, the 8XC51FC can be expanded using standard TTL compatible memories and logic.

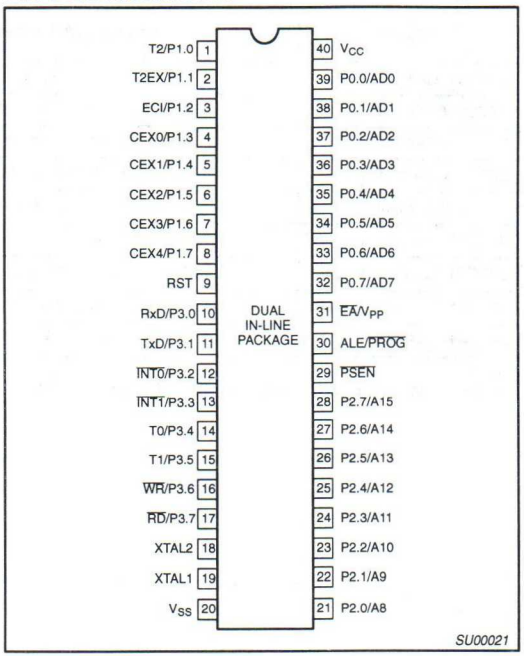
Its added features make it an even more powerful microcontroller for applications that require pulse width modulation, high-speed I/O and up/down counting capabilities such as motor control. It also has a more versatile serial channel that facilitates multiprocessor communications.

See 83C51FA/83C51FB/83C51FC/80C51FA datasheet for ROM device specification.

## FEATURES

- 80C51 central processing unit
- 32k × 8 EPROM expandable externally to 64k bytes (87C51FC)
  - Improved Quick Pulse programming algorithm
  - Three level program security system
  - 64 byte encryption array
- 256 × 8 RAM, expandable externally to 64k bytes
- Three 16-bit timer/counters
  - T2 is an up/down counter
- Programmable Counter Array (PCA)
  - High speed output
  - Capture/compare
  - Pulse Width Modulator
  - Watchdog Timer
- Four 8-bit I/O ports
- Full-duplex enhanced UART
  - Framing error detection
  - Automatic address recognition
- Power control modes
  - Idle mode
  - Power-down mode
- Once (On Circuit Emulation) Mode
- Five package styles
- OTP package available
- Programmable clock out
- 7 interrupt sources
- 4 level priority

## PIN CONFIGURATIONS



## ORDERING INFORMATION

| EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQUENCY | DRAWING NUMBER |
|---|---|---|---|---|
| S87C51FC–4N40 | OTP | 0 to +70, 40-Pin Plastic Dual In-line Package | 3.5 to 16MHz | SOT129-1 |
| S87C51FC–4F40 | UV | 0 to +70, 40-Pin Ceramic Dual In-line Package w/Window | 3.5 to 16MHz | 0590B |
| S87C51FC–4A44 | OTP | 0 to +70, 44-Pin Plastic Leaded Chip Carrier | 3.5 to 16MHz | SOT187-2 |
| S87C51FC–4K44 | UV | 0 to +70, 44-Pin Ceramic Leaded Chip Carrier w/Window | 3.5 to 16MHz | 1472A |
| S87C51FC–4B44 | OTP | 0 to +70, 44-Pin Plastic Quad Flat Pack | 3.5 to 16MHz | SOT307-2 |
| S87C51FC–5N40 | OTP | –40 to +85, 40-Pin Plastic Dual In-line Package | 3.5 to 16MHz | SOT129-1 |
| S87C51FC–5F40 | UV | –40 to +85, 40-Pin Ceramic Dual In-line Package w/Window | 3.5 to 16MHz | 0590B |
| S87C51FC–5A44 | OTP | –40 to +85, 44-Pin Plastic Leaded Chip Carrier | 3.5 to 16MHz | SOT187-2 |
| S87C51FC–5B44 | OTP | –40 to +85, 44-Pin Plastic Quad Flat Pack | 3.5 to 16MHz | SOT307-2 |

**NOTE:**
1. OTP = One Time Programmable EPROM. UV = Erasable EPROM.

## BLOCK DIAGRAM

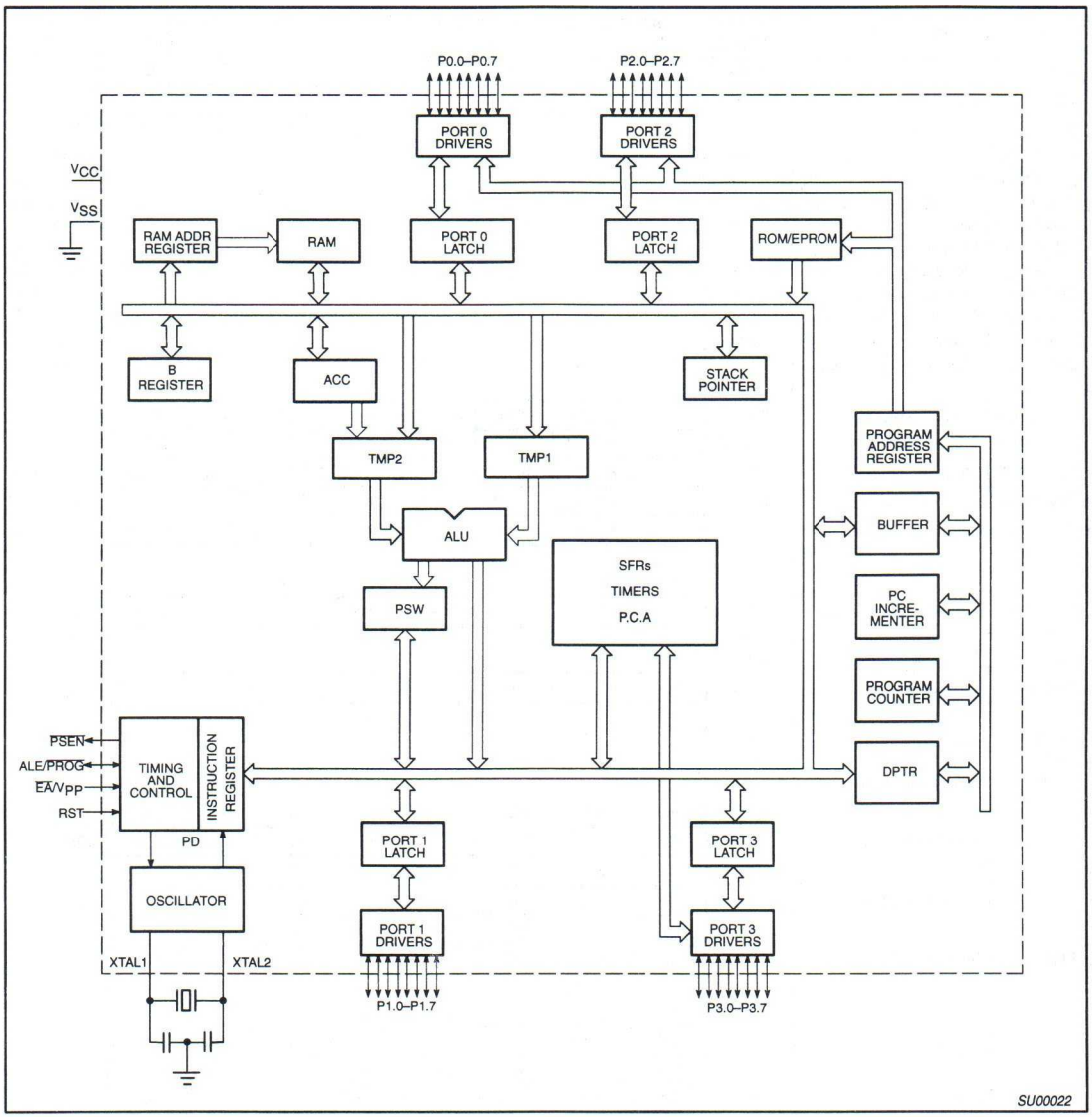


**Table 1. 8XC51FC Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| AUXR# | Auxiliary | 8EH | – | – | – | – | – | – | – | AO | xxxxxxx0B |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| CCAP0H# | Module 0 Capture High | FAH | | | | | | | | | xxxxxxxxB |
| CCAP1H# | Module 1 Capture High | FBH | | | | | | | | | xxxxxxxxB |
| CCAP2H# | Module 2 Capture High | FCH | | | | | | | | | xxxxxxxxB |
| CCAP3H# | Module 3 Capture High | FDH | | | | | | | | | xxxxxxxxB |
| CCAP4H# | Module 4 Capture High | FEH | | | | | | | | | xxxxxxxxB |
| CCAP0L# | Module 0 Capture Low | EAH | | | | | | | | | xxxxxxxxB |
| CCAP1L# | Module 1 Capture Low | EBH | | | | | | | | | xxxxxxxxB |
| CCAP2L# | Module 2 Capture Low | ECH | | | | | | | | | xxxxxxxxB |
| CCAP3L# | Module 3 Capture Low | EDH | | | | | | | | | xxxxxxxxB |
| CCAP4L# | Module 4 Capture Low | EEH | | | | | | | | | xxxxxxxxB |
| CCAPM0# | Module 0 Mode | DAH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM1# | Module 1 Mode | DBH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM2# | Module 2 Mode | DCH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM3# | Module 3 Mode | DDH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM4# | Module 4 Mode | DEH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| CCON*# | PCA Counter Control | D8H | CF | CR | – | CCF4 | CCF3 | CCF2 | CCF1 | CCF0 | 00x00000B |
| CH# | PCA Counter High | F9H | | | | | | | | | 00H |
| CL# | PCA Counter Low | E9H | | | | | | | | | 00H |
| CMOD# | PCA Counter Mode | D9H | CIDL | WDTE | – | – | – | CPS1 | CPS0 | ECF | 00xxx000B |
| DPTR: | Data Pointer (2 bytes) | | | | | | | | | | |
| DPH | Data Pointer High | 83H | | | | | | | | | 00H |
| DPL | Data Pointer Low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE* | Interrupt Enable | A8H | EA | EC | ET2 | ES | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP* | Interrupt Priority | B8H | – | PPC | PT2 | PS | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| IPH# | Interrupt Priority High | B7H | – | PPCH | PT2H | PSH | PT1H | PX1H | PT0H | PX0H | x0000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | CEX4 | CEX3 | CEX2 | CEX1 | CEX0 | ECI | T2EX | T2 | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | AD15 | AD14 | AD13 | AD12 | AD11 | AD10 | AD9 | AD8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TxD | RxD | FFH |
| PCON# | Power Control | 87H | SMOD1 | SMOD0 | – | POF[2] | GF1 | GF0 | PD | IDL | 00xx0000B |

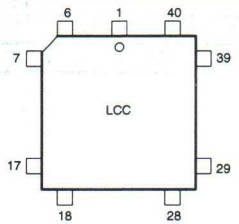
* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

**Table 1. 8XC51FC Special Function Registers** (Continued)

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program Status Word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| RACAP2H# | Timer 2 Capture High | CBH | | | | | | | | | 00H |
| RACAP2L# | Timer 2 Capture Low | CAH | | | | | | | | | 00H |
| SADDR# | Slave Address | A9H | | | | | | | | | 00H |
| SADEN# | Slave Address Mask | B9H | | | | | | | | | 00H |
| SBUF | Serial Data Buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial Control | 98H | SM0/FE | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack Pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer Control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| T2CON* | Timer 2 Control | C8H | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/$\overline{T2}$ | CP/$\overline{RL2}$ | 00H |
| T2MOD# | Timer 2 Mode Control | C9H | – | – | – | – | – | – | T2OE[3] | DCEN | xxxxxx00B |
| TH0 | Timer High 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer High 1 | 8DH | | | | | | | | | 00H |
| TH2# | Timer High 2 | CDH | | | | | | | | | 00H |
| TL0 | Timer Low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer Low 1 | 8BH | | | | | | | | | 00H |
| TL2# | Timer Low 2 | CCH | | | | | | | | | 00H |
| TMOD | Timer Mode | 89H | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
1. Reset value depends on reset source.
2. Bit will not be affected by Reset.
3. T2OE—see Programmable Clock-Out.

## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

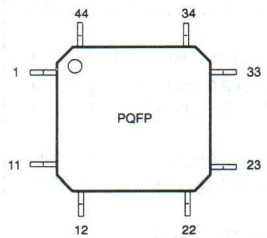



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | NC* | 16 | P3.4/T0 | 31 | P2.7/A15 |
| 2 | P1.0/T2 | 17 | P3.5/T1 | 32 | $\overline{PSEN}$ |
| 3 | P1.1/T2EX | 18 | P3.6/$\overline{WR}$ | 33 | ALE/$\overline{PROG}$ |
| 4 | P1.2/ECI | 19 | P3.7/$\overline{RD}$ | 34 | NC* |
| 5 | P1.3/CEX0 | 20 | XTAL2 | 35 | $\overline{EA}$/$V_{PP}$ |
| 6 | P1.4/CEX1 | 21 | XTAL1 | 36 | P0.7/AD7 |
| 7 | P1.5/CEX2 | 22 | $V_{SS}$ | 37 | P0.6/AD6 |
| 8 | P1.6/CEX3 | 23 | NC* | 38 | P0.5/AD5 |
| 9 | P1.7/CEX4 | 24 | P2.0/A8 | 39 | P0.4/AD4 |
| 10 | RST | 25 | P2.1/A9 | 40 | P0.3/AD3 |
| 11 | P3.0/RxD | 26 | P2.2/A10 | 41 | P0.2/AD2 |
| 12 | NC* | 27 | P2.3/A11 | 42 | P0.1/AD1 |
| 13 | P3.1/TxD | 28 | P2.4/A12 | 43 | P0.0/AD0 |
| 14 | P3.2/$\overline{INT0}$ | 29 | P2.5/A13 | 44 | $V_{CC}$ |
| 15 | P3.3/$\overline{INT1}$ | 30 | P2.6/A14 | | |

* DO NOT CONNECT

SU00023

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS




| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | P1.5/CEX2 | 16 | $V_{SS}$ | 31 | P0.6/AD6 |
| 2 | P1.6/CEX3 | 17 | NC* | 32 | P0.5/AD5 |
| 3 | P1.7/CEX4 | 18 | P2.0/A8 | 33 | P0.4/AD4 |
| 4 | RST | 19 | P2.1/A9 | 34 | P0.3/AD3 |
| 5 | P3.0/RxD | 20 | P2.2/A10 | 35 | P0.2/AD2 |
| 6 | NC* | 21 | P2.3/A11 | 36 | P0.1/AD1 |
| 7 | P3.1/TxD | 22 | P2.4/A12 | 37 | P0.0/AD0 |
| 8 | P3.2/$\overline{INT0}$ | 23 | P2.5/A13 | 38 | $V_{CC}$ |
| 9 | P3.3/$\overline{INT1}$ | 24 | P2.6/A14 | 39 | NC* |
| 10 | P3.4/T0 | 25 | P2.7/A15 | 40 | P1.0/T2 |
| 11 | P3.5/T1 | 26 | $\overline{PSEN}$ | 41 | P1.1/T2EX |
| 12 | P3.6/$\overline{WR}$ | 27 | ALE/$\overline{PROG}$ | 42 | P1.2/ECI |
| 13 | P3.7/$\overline{RD}$ | 28 | NC* | 43 | P1.3/CEX0 |
| 14 | XTAL2 | 29 | $\overline{EA}$/$V_{PP}$ | 44 | P1.4/CEX1 |
| 15 | XTAL1 | 30 | P0.7/AD7 | | |

* DO NOT CONNECT

SU00024

## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER DIP | LCC | QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification and receives code bytes during EPROM programming. External pull-ups are required during program verification. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also receives the low-order address byte during program memory verification. Alternate functions include: |
| | 1 | 2 | 40 | I | **T2 (P1.0):** Timer/Counter 2 external count input. (See Programmable Clock-Out.) |
| | 2 | 3 | 41 | I | **T2EX (P1.1):** Timer/Counter 2 Reload/Capture/Direction Control |
| | 3 | 4 | 42 | I | **ECI (P1.2):** External Clock Input to the PCA |
| | 4 | 5 | 43 | I/O | **CEX0 (P1.3):** Capture/Compare External I/O for PCA module 0 |
| | 5 | 6 | 44 | I/O | **CEX1 (P1.4):** Capture/Compare External I/O for PCA module 1 |
| | 6 | 7 | 1 | I/O | **CEX2 (P1.5):** Capture/Compare External I/O for PCA module 2 |
| | 7 | 8 | 2 | I/O | **CEX3 (P1.6):** Capture/Compare External I/O for PCA module 3 |
| | 8 | 9 | 3 | I/O | **CEX4 (P1.7):** Capture/Compare External I/O for PCA module 4 |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. Some Port 2 pins receive the high order address bits during EPROM programming and verification. |

## PIN DESCRIPTIONS (Continued)

| MNEMONIC | PIN NUMBER DIP | PIN NUMBER LCC | PIN NUMBER QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{CC}$. |
| ALE/$\overline{PROG}$ | 30 | 33 | 27 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. ALE can be disabled by setting SFR auxiliary.0. With this bit set, ALE will be active only during a MOVX instruction. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the 8XC51FC is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}$/$V_{PP}$ | 31 | 35 | 29 | I | **External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H and 7FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 7FFFH. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. If security bit 1 is programmed, $\overline{EA}$ will be internally latched on Reset. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTE:**
To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher than $V_{CC}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

### TIMER 2

This is a 16-bit up or down counter, which can be operated as either a timer or event counter. It can be operated in one of three different modes (autoreload, capture or as the baud rate generator for the UART).

In the autoreload mode the Timer can be set to count up or down by setting or clearing the bit DCEN in the T2CON Special Function Register. The SFR's RCAP2H and RCAP2L are used to reload the Timer upon overflow or a 1-to-0 transition on the T2EX input (P1.1).

In the Capture mode Timer 2 can either set TF2 and generate an interrupt or capture its value. To capture Timer 2 in response to a 1-to-0 transition on the T2EX input, the EXEN2 bit in the T2CON must be set. Timer 2 is then captured in SFR's RCAP2H and RCAP2L.

As the baud rate generator, Timer 2 is selected by setting TCLK and/or RCLK in T2CON. As the baud rate generator Timer 2 is incremented at $^1/_2$ the oscillator frequency.

### POWER OFF FLAG

The Power Off Flag (POF) is set by on-chip circuitry when the $V_{CC}$ level on the 8XC51FC rises from 0 to 5V. The POF bit can be set or cleared by software allowing a user to determine if the reset is the result of a power-on or a warm start after powerdown. The $V_{CC}$ level must remain above 3V for the POF to remain unaffected by the $V_{CC}$ level.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

### Reset

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up. Ports 1, 2, and 3 will asynchronously be driven to their reset condition when a voltage above $V_{IH1}$ is applied to RESET.

### Idle Mode

In the idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

### Power-Down Mode

To save even more power, a Power Down mode can be invoked by software. In this mode, the oscillator is stopped and the instruction that invoked Power Down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the Power Down mode is terminated.

On the 8XC51FC either a hardware reset or external interrupt can use an exit from Power Down. Reset redefines all the SFRs but does not change the on-chip RAM. An external interrupt allows both the SFRs and the on-chip RAM to retain their values.

To properly terminate Power Down the reset or external interrupt should not be executed before $V_{CC}$ is restored to its normal operating level and must be held active long enough for the oscillator to restart and stabilize (normally less than 10ms).

With an external interrupt, INT0 and INT1 must be enabled and configured as level-sensitive. Holding the pin low restarts the oscillator but bringing the pin back high completes the exit. Once the interrupt is serviced, the next instruction to be executed after RETI will be the one following the instruction that put the device into Power Down.

### Design Consideration

- When the idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal rest algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.
- The windowed parts must be covered with an opaque label to assure proper chip operation.

### ONCE™ Mode

The ONCE ("On-Circuit Emulation") Mode facilitates testing and debugging of systems using the 8XC51FC without the 8XC51FC having to be removed from the circuit. The ONCE Mode is invoked by:

1. Pull ALE low while the device is in reset and $\overline{PSEN}$ is high;
2. Hold ALE low as RST is deactivated.

While the device is in ONCE Mode, the Port 0 pins go into a float state, and the other port pins and ALE and $\overline{PSEN}$ are weakly pulled high. The oscillator circuit remains active. While the 8XC51FC is in this mode, an emulator or test CPU can be used to drive the circuit. Normal operation is restored when a normal reset is applied.

### Programmable Clock-Out

The 8XC51FC has a new feature. A 50% duty cycle clock can be programmed to come out on P1.0. This pin, besides being a regular I/O pin, has two alternate functions. It can be programmed (1) to input the external clock for Timer/Counter 2 or (2) to output a 50% duty cycle clock ranging from 61Hz to 4MHz at a 16MHz operating frequency.

To configure the Timer/Counter 2 as a clock generator, bit C/T2 (in T2CON) must be cleared and bit T20E in T2MOD must be set. Bit TR2 (T2CON.2) also must be set to start the timer.

The Clock-Out frequency depends on the oscillator frequency and the reload value of Timer 2 capture registers (RCAP2H, RCAP2L) as shown in this equation:

$$\frac{\text{OscillatorFrequency}}{4 \times (65536 - \text{RCAP2H, RCAP2L})}$$

In the Clock-Out mode Timer 2 roll-overs will not generate an interrupt. This is similar to when it is used as a baud-rate generator. It is possible to use Timer 2 as a baud-rate generator and a clock generator simultaneously. Note, however, that the baud-rate and the Clock-Out frequency will be the same.

**Table 2. External Pin Status During Idle and Power-Down Mode**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

## Programmable Counter Array (PCA)

The Programmable Counter Array is a special Timer that has five 16-bit capture/compare modules associated with it. Each of the modules can be programmed to operate in one of four modes: rising and/or falling edge capture, software timer, high-speed output, or pulse width modulator. Each module has a pin associated with it in port 1. Module 0 is connected to P1.3(CEX0), module 1 to P1.4(CEX1), etc. The basic PCA configuration is shown in Figure 1.

The PCA timer is a common time base for all five modules and can be programmed to run at: 1/12 the oscillator frequency, 1/4 the oscillator frequency, the Timer 0 overflow, or the input on the ECI pin (P1.2). The timer count source is determined from the CPS1 and CPS0 bits in the CMOD SFR as follows (see Figure 4):

| CPS1 | CPS0 | PCA Timer Count Source |
|---|---|---|
| 0 | 0 | 1/12 oscillator frequency |
| 0 | 1 | 1/4 oscillator frequency |
| 1 | 0 | Timer 0 overflow |
| 1 | 1 | External Input at ECI pin |

In the CMOD SFR are three additional bits associated with the PCA. They are CIDL which allows the PCA to stop during idle mode, WDTE which enables or disables the watchdog function on module 4, and ECF which when set causes an interrupt and the PCA overflow flag CF (in the CCON SFR) to be set when the PCA timer overflows. These functions are shown in Figure 2.

The watchdog timer function is implemented in module 4 (see Figure 14).

The CCON SFR contains the run control bit for the PCA and the flags for the PCA timer (CF) and each module (refer to Figure 5). To run the PCA the CR bit (CCON.6) must be set by software. The PCA is shut off by clearing this bit. The CF bit (CCON.7) is set when the PCA counter overflows and an interrupt will be generated if the ECF bit in the CMOD register is set, The CF bit can only be cleared by software. Bits 0 through 4 of the CCON register are the flags for the modules (bit 0 for module 0, bit 1 for module 1, etc.) and are set by hardware when either a match or a capture occurs. These flags also can only be cleared by software. The PCA interrupt system shown in Figure 3.

Each module in the PCA has a special function register associated with it. These registers are: CCAPM0 for module 0, CCAPM1 for module 1, etc. (see Figure 6). The registers contain the bits that control the mode that each module will operate in. The ECCF bit (CCAPMn.0 where n=0, 1, 2, 3, or 4 depending on the module) enables the CCF flag in the CCON SFR to generate an interrupt when a match or compare occurs in the associated module. PWM (CCAPMn.1) enables the pulse width modulation mode. The TOG bit (CCAPMn.2) when set causes the CEX output associated with the module to toggle when there is a match between the PCA counter and the module's capture/compare register. The match bit MAT (CCAPMn.3) when set will cause the CCFn bit in the CCON register to be set when there is a match between the PCA counter and the module's capture/compare register.

The next two bits CAPN (CCAPMn.4) and CAPP (CCAPMn.5) determine the edge that a capture input will be active on. The CAPN bit enables the negative edge, and the CAPP bit enables the positive edge. If both bits are set both edges will be enabled and a capture will occur for either transition. The last bit in the register ECOM (CCAPMn.6) when set enables the comparator function. Figure 7 shows the CCAPMn settings for the various PCA functions.

There are two additional registers associated with each of the PCA modules. They are CCAPnH and CCAPnL and these are the registers that store the 16-bit count when a capture occurs or a compare should occur. When a module is used in the PWM mode these registers are used to control the duty cycle of the output.

### PCA Capture Mode

To use one of the PCA modules in the capture mode either one or both of the CCAPM bits CAPN and CAPP for that module must be set. The external CEX input for the module (on port 1) is sampled for a transition. When a valid transition occurs the PCA hardware loads the value of the PCA counter registers (CH and CL) into the module's capture registers (CCAPnL and CCAPnH). If the CCFn bit for the module in the CCON SFR and the ECCFn bit in the CCAPMn SFR are set then an interrupt will be generated. Refer to Figure 10.

### 16-bit Software Timer Mode

The PCA modules can be used as software timers by setting both the ECOM and MAT bits in the modules CCAPMn register. The PCA timer will be compared to the module's capture registers and when a match occurs an interrupt will occur if the CCFn (CCON SFR) and the ECCFn (CCAPMn SFR) bits for the module are both set (see Figure 11).

### High Speed Output Mode

In this mode the CEX output (on port 1) associated with the PCA module will toggle each time a match occurs between the PCA counter and the module's capture registers. To activate this mode the TOG, MAT, and ECOM bits in the module's CCAPMn SFR must be set (see Figure 12).

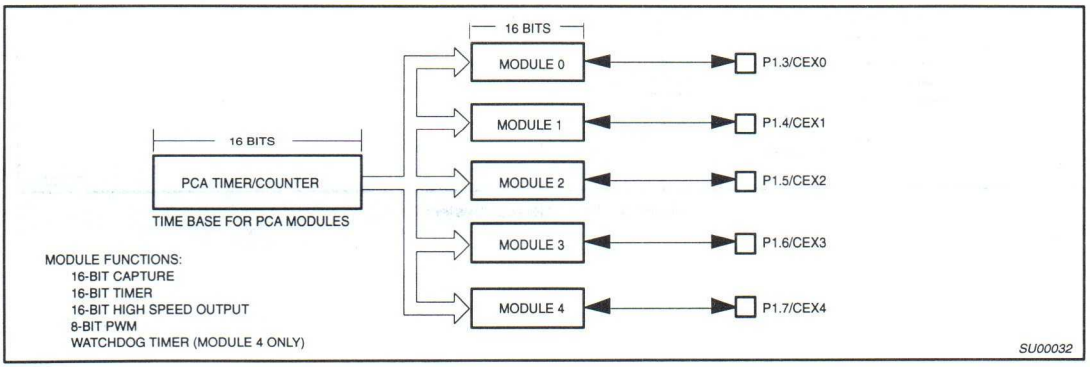



**Figure 1. Programmable Counter Array (PCA)**

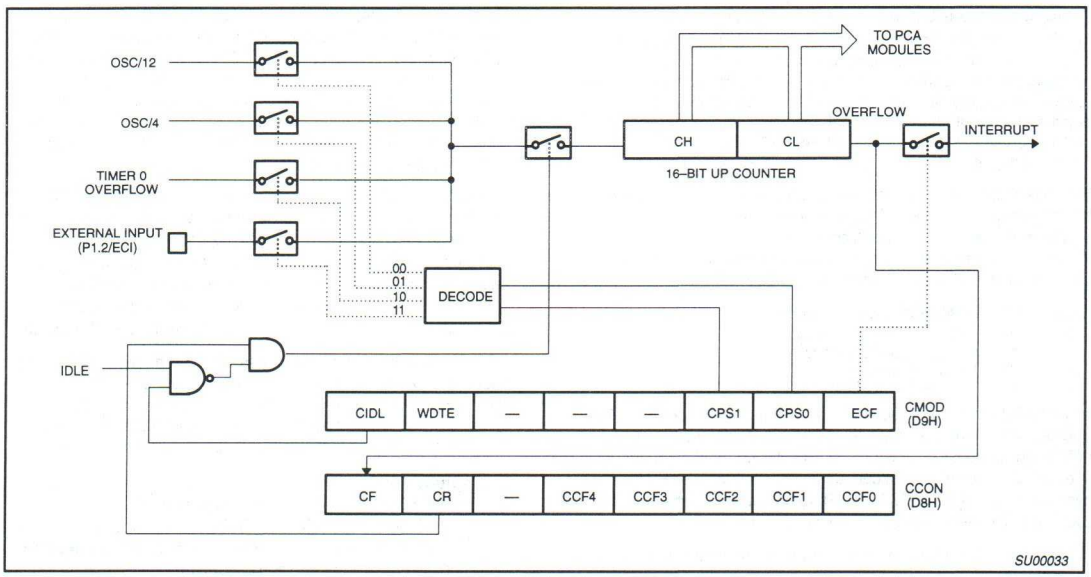



**Figure 2. PCA Timer/Counter**

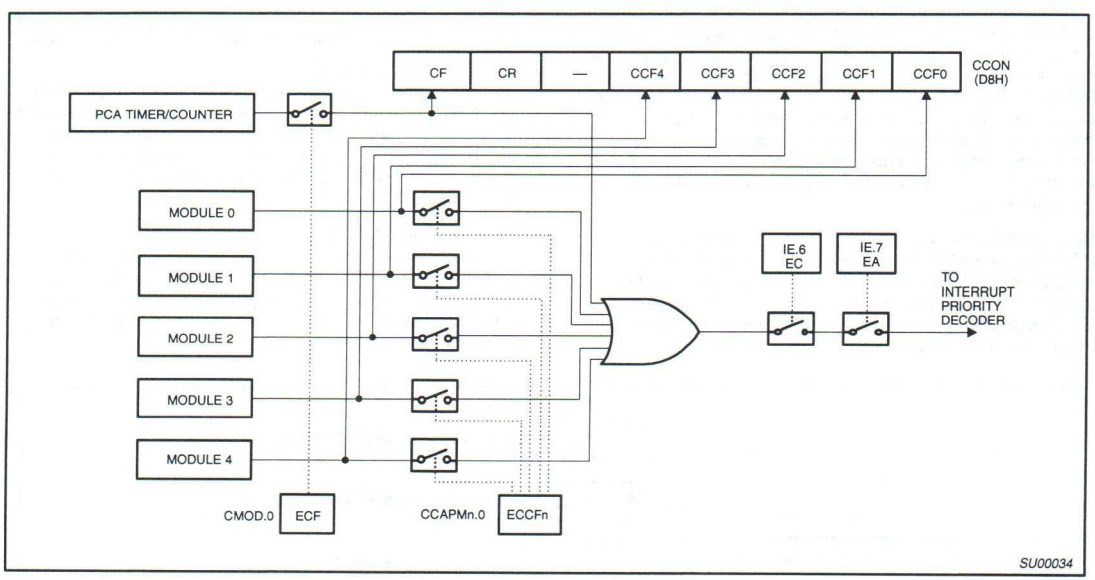



**Figure 3. PCA Interrupt System**

**CMOD** Address = OD9H Reset Value = 00XX X000B

Bit Addressable

| CIDL | WDTE | – | – | – | CPS1 | CPS0 | ECF |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Bit:

| Symbol | Function |
|---|---|
| **CIDL** | Counter Idle control: CIDL = 0 programs the PCA Counter to continue functioning during idle Mode. CIDL = 1 programs it to be gated off during idle. |
| **WDTE** | Watchdog Timer Enable: WDTE = 0 disables Watchdog Timer function on PCA Module 4. WDTE = 1 enables it. |
| – | Not implemented, reserved for future use.* |
| **CPS1** | PCA Count Pulse Select bit 1. |
| **CPS0** | PCA Count Pulse Select bit 0. |

| CPS1 | CPS0 | Selected PCA Input** | |
|---|---|---|---|
| 0 | 0 | 0 | Internal clock, $f_{OSC} \div 12$ |
| 0 | 1 | 1 | Internal clock, $f_{OSC} \div 4$ |
| 1 | 0 | 2 | Timer 0 overflow |
| 1 | 1 | 3 | External clock at ECI/P1.2 pin (max. rate = $f_{OSC} \div 8$) |

| | |
|---|---|
| **ECF** | PCA Enable Counter Overflow interrupt: ECF = 1 enables CF bit in CCON to generate an interrupt. ECF = 0 disables that function of CF. |

**NOTE:**
* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.
** $f_{OSC}$ = oscillator frequency

*SU00035*

**Figure 4. CMOD: PCA Counter Mode Register**

**CCON** Address = OD8H Reset Value = 00X0 0000B

Bit Addressable

| CF | CR | – | CCF4 | CCF3 | CCF2 | CCF1 | CCF0 |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Bit:

| Symbol | Function |
|---|---|
| **CF** | PCA Counter Overflow flag. Set by hardware when the counter rolls over. CF flags an interrupt if bit ECF in CMOD is set. CF may be set by either hardware or software but can only be cleared by software. |
| **CR** | PCA Counter Run control bit. Set by software to turn the PCA counter on. Must be cleared by software to turn the PCA counter off. |
| – | Not implemented, reserved for future use*. |
| **CCF4** | PCA Module 4 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF3** | PCA Module 3 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF2** | PCA Module 2 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF1** | PCA Module 1 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF0** | PCA Module 0 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |

**NOTE:**
* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

*SU00036*

**Figure 5. CCON: PCA Counter Control Register**

**CCAPMn** Address CCAPM0 0DAH
CCAPM1 0DBH
CCAPM2 0DCH
CCAPM3 0DDH
CCAPM4 0DEH

Reset Value = X000 0000B

Not Bit Addressable

| | – | ECOMn | CAPPn | CAPNn | MATn | TOGn | PWMn | ECCFn |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Symbol | Function |
|---|---|
| – | Not implemented, reserved for future use*. |
| **ECOMn** | Enable Comparator. ECOMn = 1 enables the comparator function. |
| **CAPPn** | Capture Positive, CAPPn = 1 enables positive edge capture. |
| **CAPNn** | Capture Negative, CAPNn = 1 enables negative edge capture. |
| **MATn** | Match. When MATn = 1, a match of the PCA counter with this module's compare/capture register causes the CCFn bit in CCON to be set, flagging an interrupt. |
| **TOGn** | Toggle. When TOGn = 1, a match of the PCA counter with this module's compare/capture register causes the CEXn pin to toggle. |
| **PWMn** | Pulse Width Modulation Mode. PWMn = 1 enables the CEXn pin to be used as a pulse width modulated output. |
| **ECCFn** | Enable CCF interrupt. Enables compare/capture flag CCFn in the CCON register to generate an interrupt. |

**NOTE:**
*User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

*SU00037*

**Figure 6. CCAPMn: PCA Modules Compare/Capture Registers**

| – | ECOMn | CAPPn | CAPNn | MATn | TOGn | PWMn | ECCFn | MODULE FUNCTION |
|---|---|---|---|---|---|---|---|---|
| X | 0 | 0 | 0 | 0 | 0 | 0 | 0 | No operation |
| X | X | 1 | 0 | 0 | 0 | 0 | X | 16-bit capture by a positive-edge trigger on CEXn |
| X | X | 0 | 1 | 0 | 0 | 0 | X | 16-bit capture by a negative trigger on CEXn |
| X | X | 1 | 1 | 0 | 0 | 0 | X | 16-bit capture by a transition on CEXn |
| X | 1 | 0 | 0 | 1 | 0 | 0 | X | 16-bit Software Timer |
| X | 1 | 0 | 0 | 1 | 1 | 0 | X | 16-bit High Speed Output |
| X | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 8-bit PWM |
| X | 1 | 0 | 0 | 1 | X | 0 | X | Watchdog Timer |

**Figure 7. PCA Module Modes (CCAPMn Register)**

**Pulse Width Modulator Mode**

All of the PCA modules can be used as PWM outputs. Figure 13 shows the PWM function. The frequency of the output depends on the source for the PCA timer. All of the modules will have the same frequency of output because they all share the PCA timer. The duty cycle of each module is independently variable using the module's capture register CCAPLn. When the value of the PCA CL SFR is less than the value in the module's CCAPLn SFR the output will be low, when it is equal to or greater than the output will be high. When CL overflows from FF to 00, CCAPLn is reloaded with the value in CCAPHn. the allows updating the PWM without glitches. The PWM and ECOM bits in the module's CCAPMn register must be set to enable the PWM mode.

## Enhanced UART

The UART operates in all of the usual modes that are described in the first section of this book for the 80C51. In addition the UART can perform framing error detect by looking for missing stop bits, and automatic address recognition. The 8XC51FC UART also fully supports multiprocessor communication as does the standard 80C51 UART.

When used for framing error detect the UART looks for missing stop bits in the communication. A missing bit will set the FE bit in the SCON register. The FE bit shares the SCON.7 bit with SM0 and the function of SCON.7 is determined by PCON.6 (SMOD0) (see Figure 15). If SMOD0 is set then SCON.7 functions as FE. SCON.7 functions as SM0 when SMOD0 is cleared. When used as FE SCON.7 can only be cleared by software. Refer to Figure 16.

### Automatic Address Recognition

Automatic Address Recognition is a feature which allows the UART to recognize certain addresses in the serial bit stream by using hardware to make the comparisons. This feature saves a great deal of software overhead by eliminating the need for the software to examine every serial address which passes by the serial port. This feature is enabled by setting the SM2 bit in SCON. In the 9 bit UART modes, mode 2 and mode 3, the Receive Interrupt flag (RI) will be automatically set when the received byte contains either the "Given" address or the "Broadcast" address. The 9 bit mode requires that the 9th information bit is a 1 to indicate that the received information is an address and not data. Automatic address recognition is shown in Figure 17.

The 8 bit mode is called Mode 1. In this mode the RI flag will be set if SM2 is enabled and the information received has a valid stop bit following the 8 address bits and the information is either a Given or Broadcast address.

Mode 0 is the Shift Register mode and SM2 is ignored.

Using the Automatic Address Recognition feature allows a master to selectively communicate with one or more slaves by invoking the Given slave address or addresses. All of the slaves may be contacted by using the Broadcast address. Two special Function Registers are used to define the slave's address, SADDR, and the address mask, SADEN. SADEN is used to define which bits in the SADDR are to b used and which bits are "don't care". The SADEN mask can be logically ANDed with the SADDR to create the "lGiven" address which the master will use for addressing each of the slaves. Use of the Given address allows multiple slaves to be recognized while excluding others. The following examples will help to show the versatility of this scheme:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1101 |
| | Given | = | 1100 00X0 |
| Slave 1 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1110 |
| | Given | = | 1100 000X |

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires a 0 in bit 0 and it ignores bit 1. Slave 1 requires a 0 in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010 since slave 1 requires a 0 in bit 1. A unique address for slave 1 would be 1100 0001 since a 1 in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 1100 0000.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1001 |
| | Given | = | 1100 0XX0 |
| Slave 1 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1010 |
| | Given | = | 1110 0X0X |
| Slave 2 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1100 |
| | Given | = | 1110 00XX |

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 and 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary t make bit 2 = 1 to exclude slave 2.

The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are teated as don't-cares. In most cases, interpreting the don't-cares as ones, the broadcast address will be FF hexadecimal.

Upon reset SADDR (SFR address 0A9H) and SADEN (SFR address 0B9H) are loaded with 0s. This produces a given address of all "don't cares" as well as a Broadcast address of all "don't cares". this effectively disables the Automatic Addressing mode and allows the microcontroller to use standard 80C51 type UART drivers which do not make use of this feature.

## Reduced EMI Mode

The AO bit (AUXR.O) in the AUXR register when set disables the ALE output.

## 8XC51FC Reduced EMI Mode

**AUXR (8EH)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | – | – | – | AO |

AO: Turns off ALE output.

## Interrupt Priority Structure

The 8XC51FC has a 7-source four-level interrupt structure. There are 3 SFRs associated with the interrupts on the 8XC51FC. They are the IE and IP. (See Figures 8 and 9.) In addition, there is the IPH (Interrupt Priority High) register that makes the four-level interrupt structure possible. The IPH is located at SFR address B7H. The structure of the IPH register and a description of its bits is shown below:

**IPH (Interrupt Priority High) (B7H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | PPCH | PT2H | PSH | PT1H | PX1H | PT0H | PX0H |

| | | |
|---|---|---|
| IPH.0 | PX0H | External interrupt 0 priority high |
| IPH.1 | PT0H | Timer 0 interrupt priority high |
| IPH.2 | PX1H | External interrupt 1 priority high |
| IPH.3 | PT1H | Timer 1 interrupt priority high |
| IPH.4 | PSH | Serial Port interrupt high |
| IPH.5 | PT2H | Timer 2 interrupt priority high |
| IPH.6 | PPCH | PCA interrupt priority high |
| IPH.7 | — | Not implemented |

The function of the IPH SFR is simple and when combined with the IP SFR determines the priority of each interrupt. The priority of each interrupt is determined as shown in the following table:

| PRIORITY BITS | | INTERRUPT PRIORITY LEVEL |
|---|---|---|
| IPH.x | IP.x | |
| 0 | 0 | Level 0 (lowest priority) |
| 0 | 1 | Level 1 |
| 1 | 0 | Level 2 |
| 1 | 1 | Level 3 (highest priority) |

The priority scheme for servicing the interrupts is the same as that for the 80C51, except there are four interrupt levels on the 8XC51FC rather than two as on the 80C51. An interrupt will be serviced as long as an interrupt of equal or higher priority is not already being serviced. If an interrupt of equal or higher level priority is being serviced, the new interrupt will wait until it is finished before being serviced. If a lower priority level interrupt is being serviced, it will be stopped and the new interrupt serviced. When the new interrupt is finished, the lower priority level interrupt that was stopped will be completed.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| IE (0A8H) | EA | EC | ET2 | ES | ET1 | EX1 | ET0 | EX0 |

Enable Bit = 1 enables the interrupt.
Enable Bit = 0 disables it.

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IE.7 | EA | Global disable bit. If EA = 0, all interrupts are disabled. If EA = 1, each interrupt can be individually enabled or disabled by setting or clearing its enable bit. |
| IE.6 | EC | PCA interrupt enable bit. |
| IE.5 | ET2 | Timer 2 interrupt enable bit. |
| IE.4 | ES | Serial Port interrupt enable bit. |
| IE.3 | ET1 | Timer 1 interrupt enable bit. |
| IE.2 | EX1 | External interrupt 1 enable bit. |
| IE.1 | ET0 | Timer 0 interrupt enable bit. |
| IE.0 | EX0 | External interrupt 0 enable bit. |

SU00571

**Figure 8. IE Registers**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| IP (0B8H) | — | PPC | PT2 | PS | PT1 | PX1 | PT0 | PX0 |

Priority Bit = 1 assigns high priority
Priority Bit = 0 assigns low priority

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IP.7 | — | Not implemented, reserved for future use. |
| IP.6 | PPC | PCA interrupt priority bit. |
| IP.5 | PT2 | Timer 2 interrupt priority bit. |
| IP.4 | PS | Serial Port interrupt priority bit. |
| IP.3 | PT1 | Timer 1 interrupt priority bit. |
| IP.2 | PX1 | External interrupt 1 priority bit. |
| IP.1 | PT0 | Timer 0 interrupt priority bit. |
| IP.0 | PX0 | External interrupt 0 priority bit. |

SU00572

**Figure 9. IP Registers**

**Table 3. Interrupt Table**

| SOURCE | POLLING PRIORITY | REQUEST BITS | HARDWARE CLEAR? | VECTOR ADDRESS |
|---|---|---|---|---|
| X0 | 1 | IE0 | N (L)[1] Y (T)[2] | 03H |
| T0 | 2 | TP0 | Y | 0B |
| X1 | 3 | IE1 | N (L) Y (T) | 13 |
| T1 | 4 | TF1 | Y | 1B |
| SP | 5 | R1, TI | N | 23 |
| T2 | 6 | TF2, EXF2 | N | 2B |
| PCA | 7 | CF, CCFn n = 0–4 | N | 33 |

**NOTES:**
1. L = Level activated
2. T = Transition activated

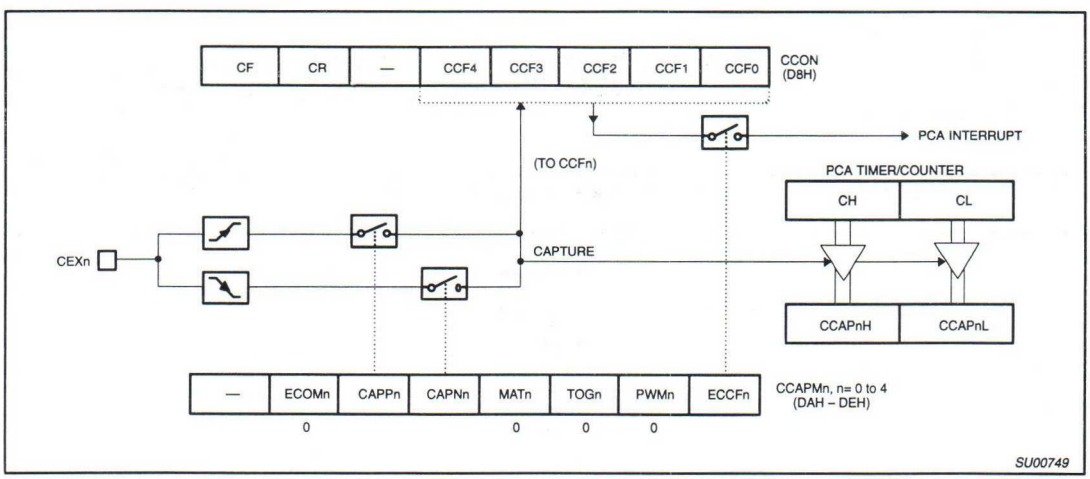



**Figure 10. PCA Capture Mode**

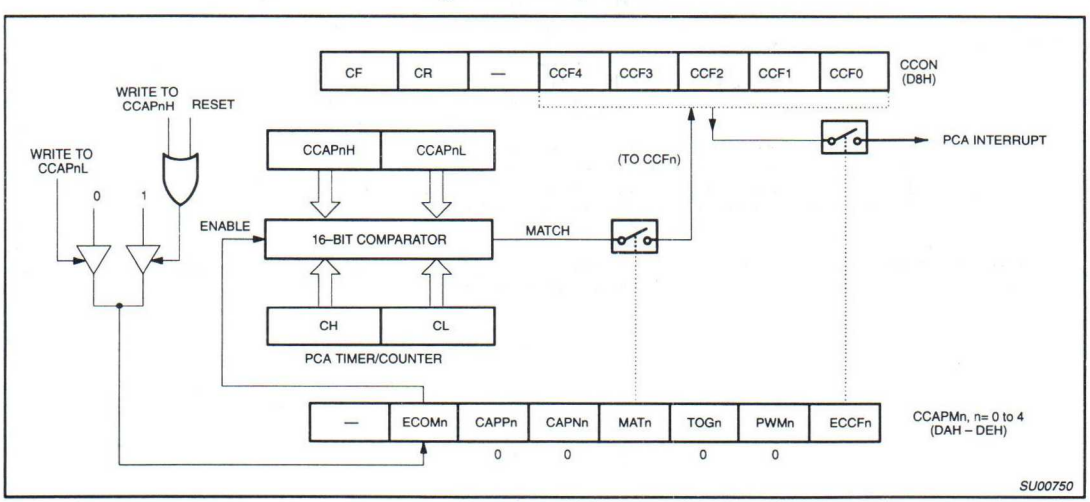



**Figure 11. PCA Compare Mode**

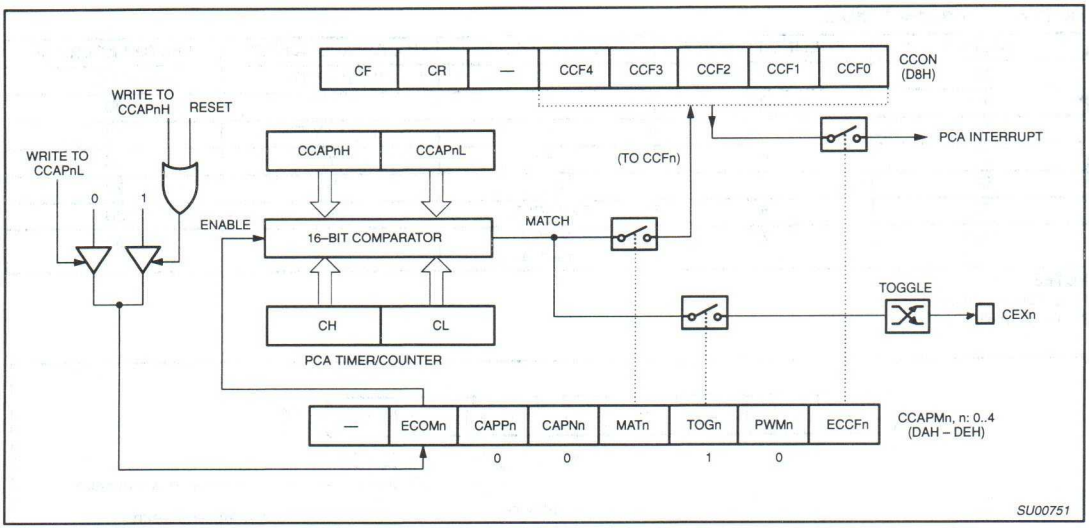



**Figure 12. PCA High Speed Output Mode**

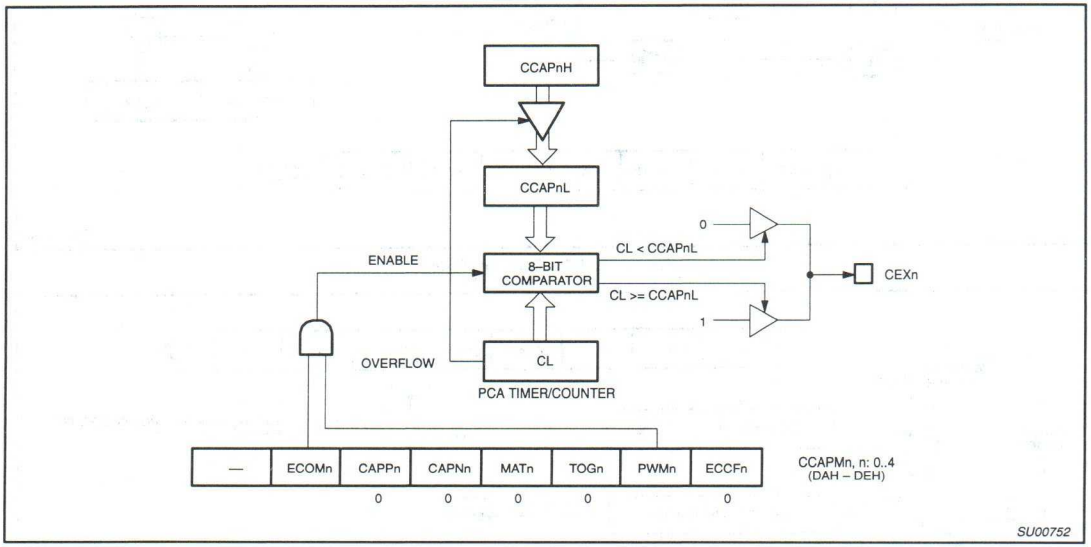



**Figure 13. PCA PWM Mode**

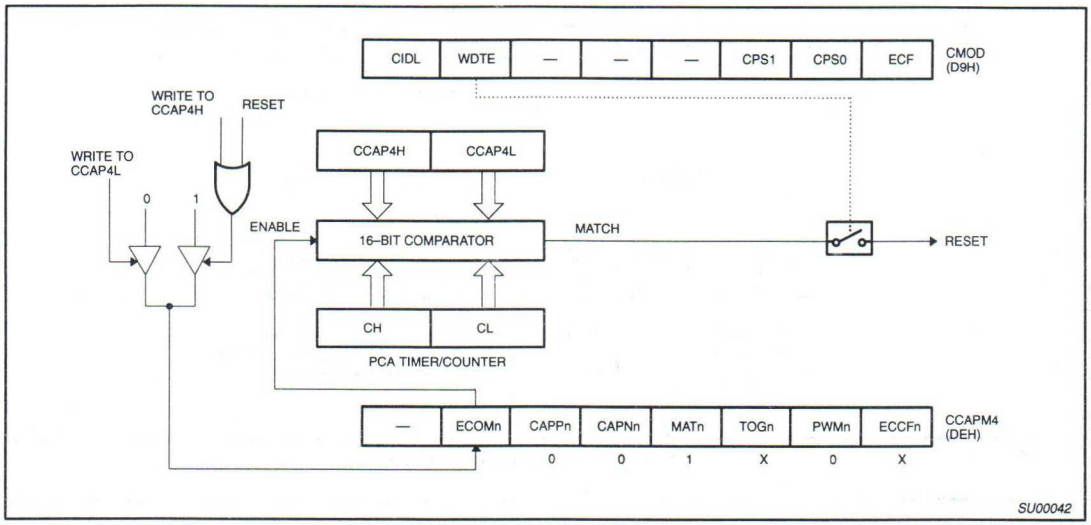



**Figure 14. PCA Watchdog Timer**

**SCON** Address = 98H Reset Value = 0000 0000B

Bit Addressable

| | SM0/FE | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

(SMOD0 = 0/1)*

| Symbol | Function |
|---|---|
| **FE** | Framing Error bit. This bit is set by the receiver when an invalid stop bit is detected. The FE bit is not cleared by valid frames but should be cleared by software. The SMOD0 bit must be set to enable access to the FE bit. |
| **SM0** | Serial Port Mode Bit 0, (SMOD0 must = 0 to access bit SM0) |
| **SM1** | Serial Port Mode Bit 1 |

| SM0 | SM1 | Mode | Description | Baud Rate** |
|---|---|---|---|---|
| 0 | 0 | 0 | shift register | $f_{OSC}/12$ |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{OSC}/64$ or $f_{OSC}/32$ |
| 1 | 1 | 3 | 9-bit UART | variable |

| | |
|---|---|
| **SM2** | Enables the Automatic Address Recognition feature in Modes 2 or 3. If SM2 = 1 then RI will not be set unless the received 9th data bit (RB8) is 1, indicating an address, and the received byte is a Given or Broadcast Address. In Mode 1, if SM2 = 1 then RI will not be activated unless a valid stop bit was received, and the received byte is a Given or Broadcast Address. In Mode 0, SM2 should be 0. |
| **REN** | Enables serial reception. Set by software to enable reception. Clear by software to disable reception. |
| **TB8** | The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired. |
| **RB8** | In modes 2 and 3, the 9th data bit that was received. In Mode 1, if SM2 = 0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used. |
| **TI** | Transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software. |
| **RI** | Receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software. |

**NOTE:**
*SMOD0 is located at PCON6.
**$f_{OSC}$ = oscillator frequency

SU00043

**Figure 15. SCON: Serial Port Control Register**

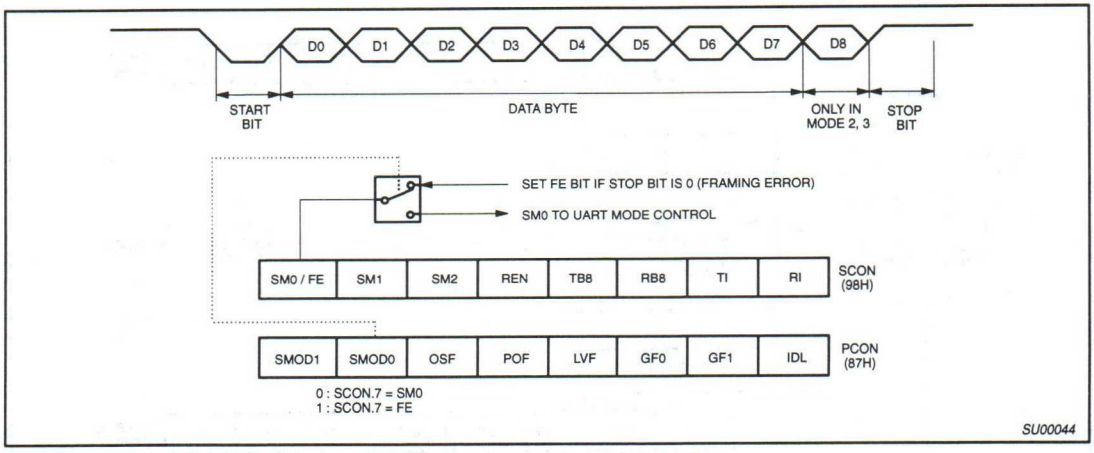



**Figure 16. UART Framing Error Detection**

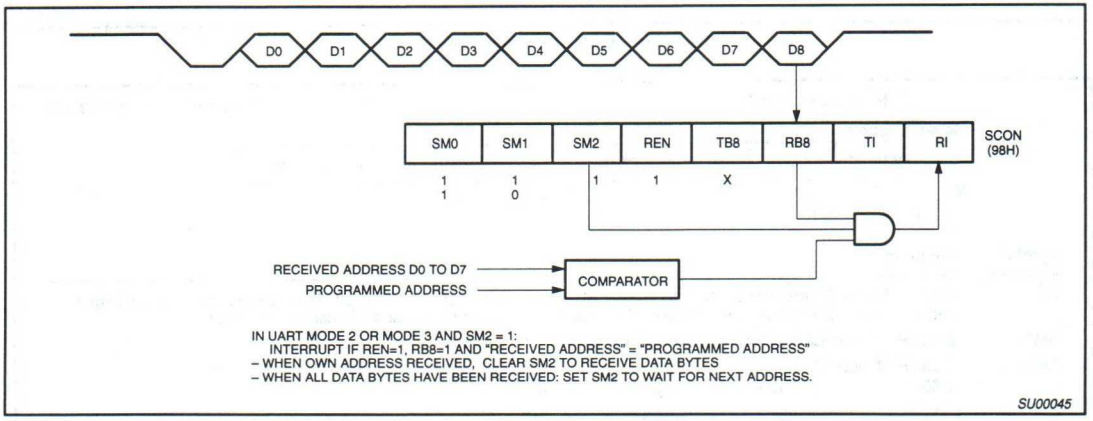



**Figure 17. UART Multiprocessor Communication, Automatic Address Recognition**

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | 0 to +70 or –40 to +85 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Maximum $I_{OL}$ per I/O pin | 15 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.5 | W |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$ | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | | 0 | | $0.2V_{CC}$–0.3 | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3[8] | $I_{OL}$ = 1.6mA2 | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$[8] | $I_{OL}$ = 3.2mA2 | | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3 [3] | $I_{OH}$ = –30µA | $V_{CC}$ – 0.7 | | | V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode), ALE[9], $\overline{PSEN}$[3] | $I_{OH}$ = –3.2mA | $V_{CC}$ – 0.7 | | | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3 | $V_{IN}$ = 0.45V | | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3[6] | See note 4 | | | –650 | µA |
| $I_{LI}$ | Input leakage current, port 0 | 0.45 $V_{IN}$ < $V_{CC}$ – 0.3 | | | ±10 | µA |
| $I_{CC}$ | Power supply current:<br>Active mode @ 16MHz<br>Idle mode @ 16MHz<br>Power-down mode | See note 5 | | <br>15<br>3<br>10 | <br>32<br>10<br>100 | <br>mA<br>mA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | 40 | | 225 | kΩ |
| $C_{IO}$ | Pin capacitance[10] (except EA) | | | | 15 | pF |

**NOTES:**

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
5. See Figures 25 through 28 for $I_{CC}$ test conditions.
6. This value applies to $T_{amb}$ = 0°C to +70°C. For $T_{amb}$ = –40°C to +85°C, $I_{TL}$ = –750µA.
7. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
8. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 15mA (*NOTE: This is 85°C specification.)
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 71mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
9. ALE is tested to $V_{OH1}$, except when ALE is off then $V_{OH}$ is the voltage specification.
10. Pin capacitance is characterized but not tested. Pin capacitance is less than 25pF. Pin capacitance of ceramic package is less than 15pF (except $\overline{EA}$ is 25pF).

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 18 | Oscillator frequency –4 –5 | | | 3.5 | 16 | MHz |
| $t_{LHLL}$ | 18 | ALE pulse width | 85 | | $2t_{CLCL}$–40 | | ns |
| $t_{AVLL}$ | 18 | Address valid to ALE low | 22 | | $t_{CLCL}$–40 | | ns |
| $t_{LLAX}$ | 18 | Address hold after ALE low | 32 | | $t_{CLCL}$–30 | | ns |
| $t_{LLIV}$ | 18 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}$–100 | ns |
| $t_{LLPL}$ | 18 | ALE low to $\overline{PSEN}$ low | 32 | | $t_{CLCL}$–30 | | ns |
| $t_{PLPH}$ | 18 | $\overline{PSEN}$ pulse width | 142 | | $3t_{CLCL}$–45 | | ns |
| $t_{PLIV}$[4] | 18 | $\overline{PSEN}$ low to valid instruction in | | 82 | | $3t_{CLCL}$–105 | ns |
| $t_{PXIX}$ | 18 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 18 | Input instruction float after $\overline{PSEN}$ | | 37 | | $t_{CLCL}$–25 | ns |
| $t_{AVIV}$ | 18 | Address to valid instruction in | | 207 | | $5t_{CLCL}$–105 | ns |
| $t_{PLAZ}$ | 18 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 19, 20 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}$–100 | | ns |
| $t_{WLWH}$ | 19, 20 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}$–100 | | ns |
| $t_{RLDV}$ | 19, 20 | $\overline{RD}$ low to valid data in | | 147 | | $5t_{CLCL}$–165 | ns |
| $t_{RHDX}$ | 19, 20 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 19, 20 | Data float after $\overline{RD}$ | | 65 | | $2t_{CLCL}$–60 | ns |
| $t_{LLDV}$ | 19, 20 | ALE low to valid data in | | 350 | | $8t_{CLCL}$–150 | ns |
| $t_{AVDV}$ | 19, 20 | Address to valid data in | | 397 | | $9t_{CLCL}$–165 | ns |
| $t_{LLWL}$ | 19, 20 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 137 | 237 | $3t_{CLCL}$–50 | $3t_{CLCL}$+50 | ns |
| $t_{AVWL}$ | 19, 20 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 175 | | $4t_{CLCL}$–130 | | ns |
| $t_{QVWX}$ | 19, 20 | Data valid to $\overline{WR}$ transition | 42 | | $t_{CLCL}$–50 | | ns |
| $t_{WHQX}$ | 19, 20 | Data hold after $\overline{WR}$ | 42 | | $t_{CLCL}$–50 | | ns |
| $t_{QVWH}$ | 20 | Data valid to $\overline{WR}$ high | 287 | | $7t_{CLCL}$–150 | | ns |
| $t_{RLAZ}$ | 19, 20 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 19, 20 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 40 | 87 | $t_{CLCL}$–40 | $t_{CLCL}$+40 | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 22 | High time | 12 | | 20 | | ns |
| $t_{CLCX}$ | 22 | Low time | 12 | | 20 | | ns |
| $t_{CLCH}$ | 22 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 22 | Fall time | | 20 | | 20 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 21 | Serial port clock cycle time | 1 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 21 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}$–133 | | ns |
| $t_{XHQX}$ | 21 | Output data hold after clock rising edge | 8 | | $2t_{CLCL}$–117 | | ns |
| $t_{XHDX}$ | 21 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 21 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}$–133 | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 8XC51FC to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.
4. See application note AN457.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{\text{PSEN}}$
Q – Output data
R – $\overline{\text{RD}}$ signal
t – Time
V – Valid
W – $\overline{\text{WR}}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ =Time for ALE low to $\overline{\text{PSEN}}$ low.

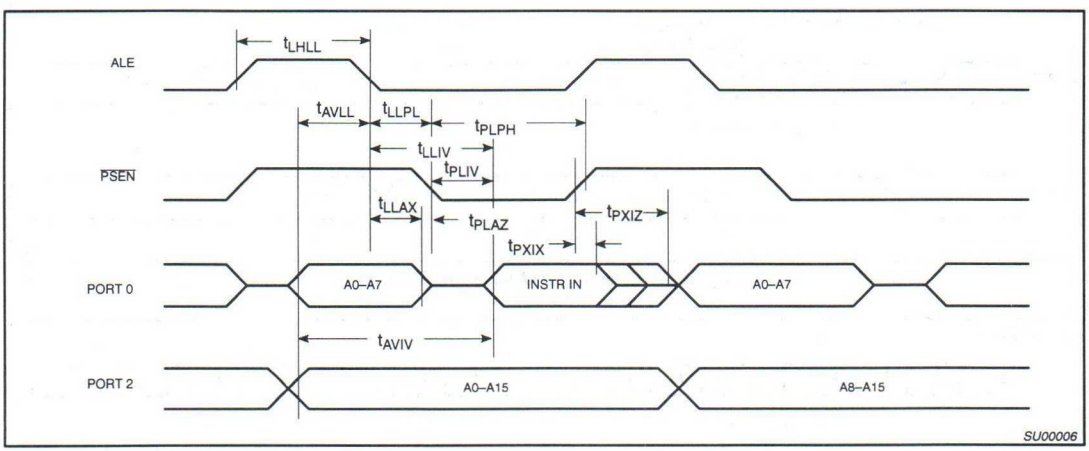



**Figure 18. External Program Memory Read Cycle**

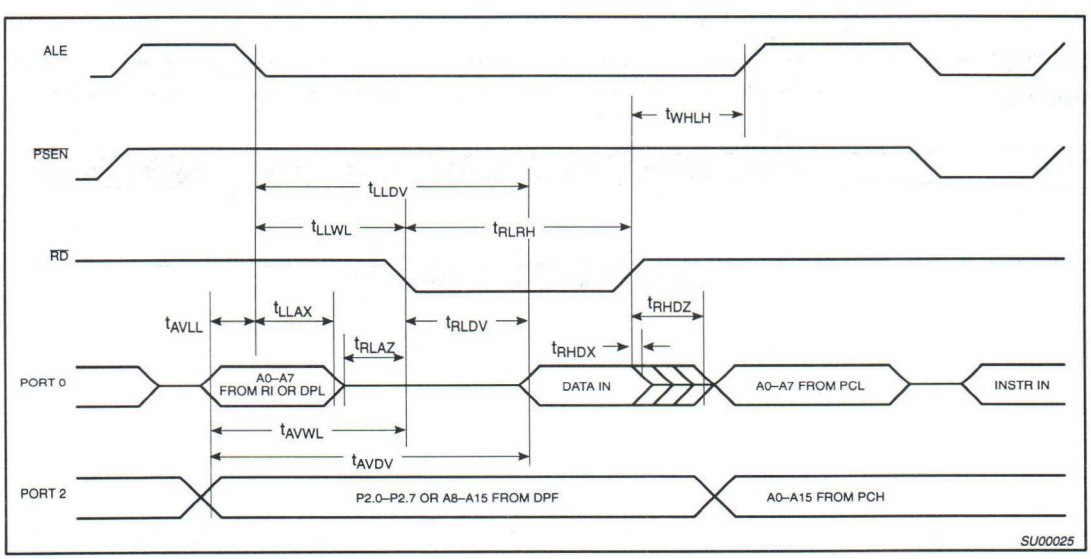



**Figure 19. External Data Memory Read Cycle**

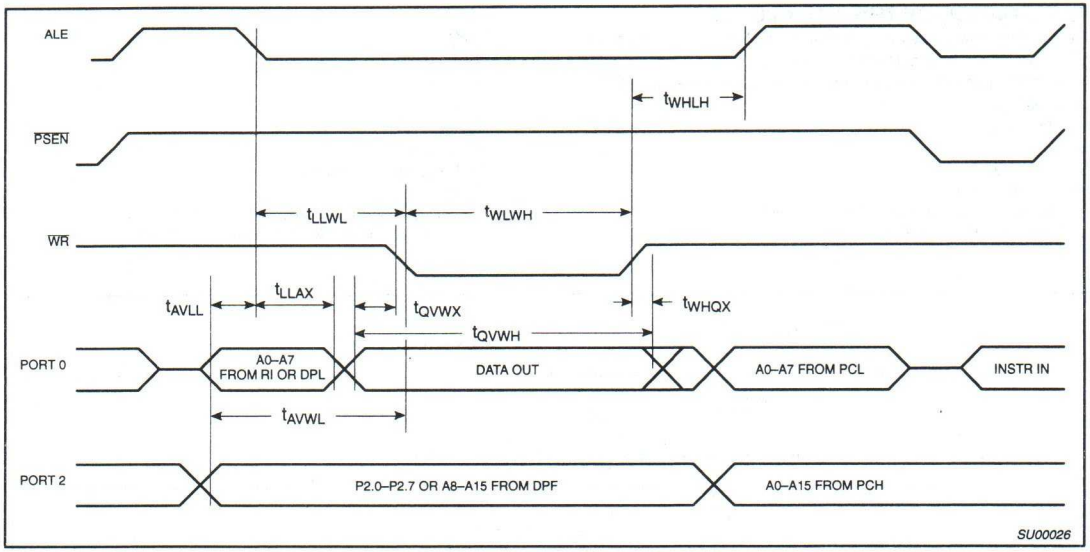



**Figure 20. External Data Memory Write Cycle**

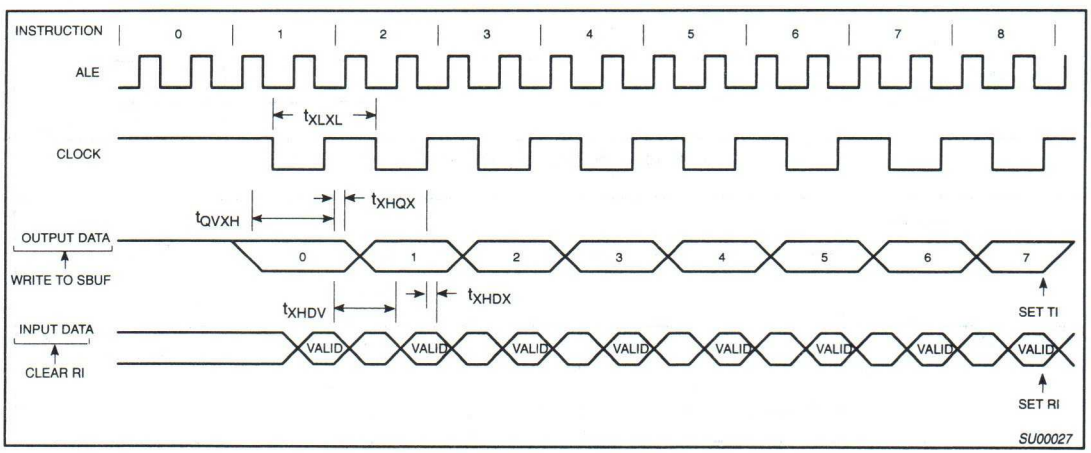



**Figure 21. Shift Register Mode Timing**

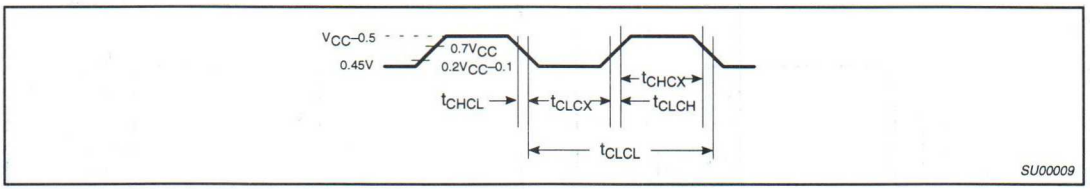



**Figure 22. External Clock Drive**

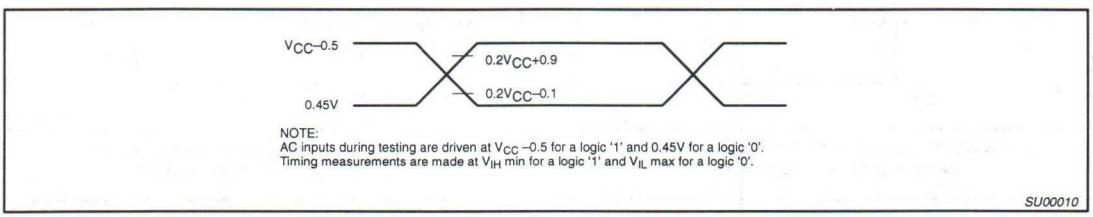



**Figure 23. AC Testing Input/Output**

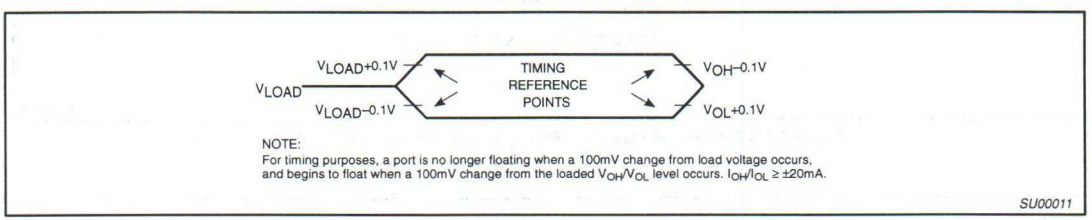



**Figure 24. Float Waveform**

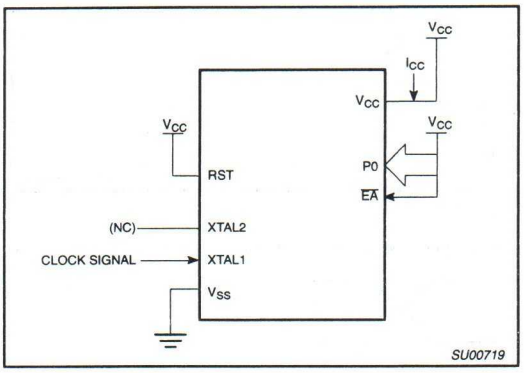



**Figure 25. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

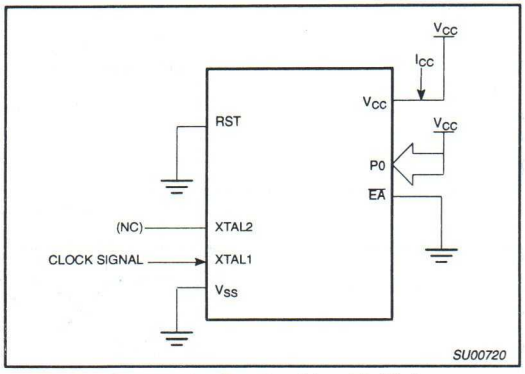



**Figure 26. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

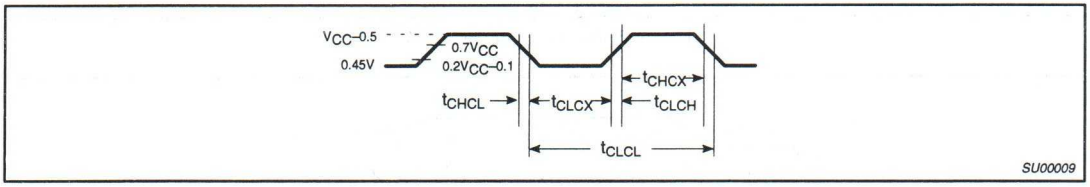



**Figure 27. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

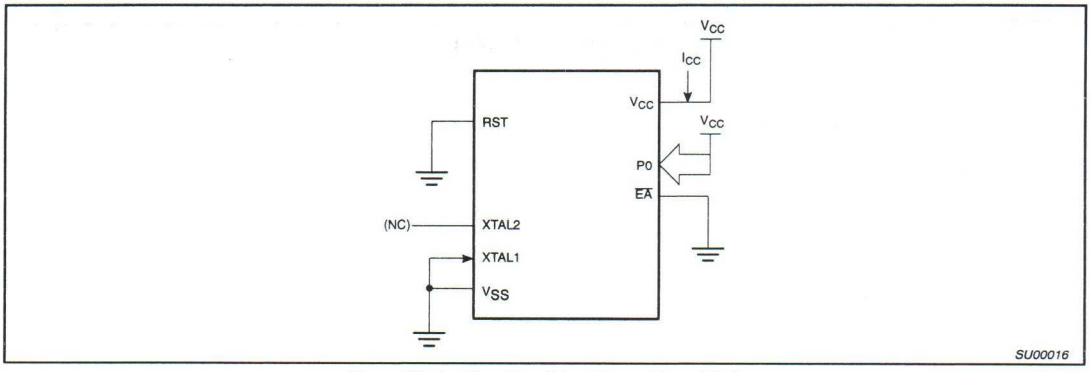



**Figure 28. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2V to 5.5V**

## EPROM CHARACTERISTICS

The 87C51FC is programmed by using a modified Improved Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C51FC contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C51FC manufactured by Philips.

Table 4 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the security bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 29 and 30. Figure 31 shows the circuit configuration for normal program memory verification.

### Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 29. Note that the 87C51FC is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1 and 2, as shown in Figure 29. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 3 specified in Table 4 are held at the 'Program Code Data' levels indicated in Table 4. The ALE/$\overline{PROG}$ is pulsed low 5 times as shown in Figure 30.

To program the encryption table, repeat the 25 pulse programming sequence for addresses 0 through 1FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the security bits, repeat the 25 pulse programming sequence using the 'Pgm Security Bit' levels. After one security bit is programmed, further programming of the code memory and encryption table is disabled. However, the other security bit can still be programmed.

Note that the $\overline{EA}$/$V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

**Program Verification**

If security bits 2 and 3 have not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 1 and 2 as shown in Figure 31. The other pins are held at the 'Verify Code Data' levels indicated in Table 4. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the 64 byte encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

**Reading the Signature Bytes**

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:
(030H) = 15H indicates manufactured by Philips
(031H) = B3H indicates 87C51FC
(060H) = FCH

### Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 4, and which satisfies the timing specifications, is suitable.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-s/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

### Security Bits

With none of the security bits programmed the code in the program memory can be verified. If the encryption table is programmed, the code will be encrypted when verified. When only security bit 1 (see Table 5) is programmed, MOVC instructions executed from external program memory are disabled from fetching code bytes from the internal memory, EA is latched on Reset and all further programming of the EPROM is disabled. When security bits 1 and 2 are programmed, in addition to the above, verify mode is disabled. When all three security bits are programmed, all of the conditions above apply and all external program memory execution is disabled.

### Encryption Array

64 bytes of encryption array are initially unprogrammed (all 1s).

™Trademark phrase of Intel Corporation.

**Table 4. EPROM Programming Modes**

| MODE | RST | $\overline{PSEN}$ | ALE/$\overline{PROG}$ | $\overline{EA}$/$V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 | P3.3 |
|---|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 | 1 |
| Pgm security bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 | 1 |
| Pgm security bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 | 1 |
| Pgm security bit 3 | 1 | 0 | 0* | $V_{PP}$ | 0 | 1 | 0 | 1 | 1 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V±10% during programming and verification.
* ALE/$\overline{PROG}$ receives 5 programming pulses (only for user array; 25 pulses for encryption or security bits) while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

**Table 5. Program Security Bits**

| PROGRAM LOCK BITS[1, 2] | | | | |
|---|---|---|---|---|
| | SB1 | SB2 | SB3 | PROTECTION DESCRIPTION |
| 1 | U | U | U | No Program Security features enabled. (Code verify will still be encrypted by the Encryption Array if programmed.) |
| 2 | P | U | U | MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory, $\overline{EA}$ is sampled and latched on Reset, and further programming of the EPROM is disabled. |
| 3 | P | P | U | Same as 2, also verify is disabled. |
| 4 | P | P | P | Same as 3, external execution is disabled. Internal data RAM is not accessible. |

**NOTES:**
1. P – programmed. U – unprogrammed.
2. Any other combination of the security bits is not defined.

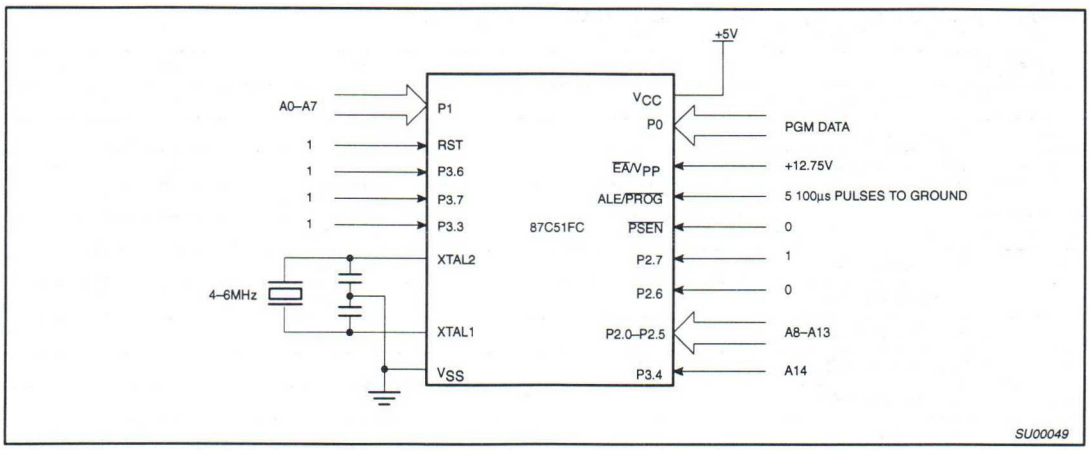



**Figure 29. Programming Configuration**

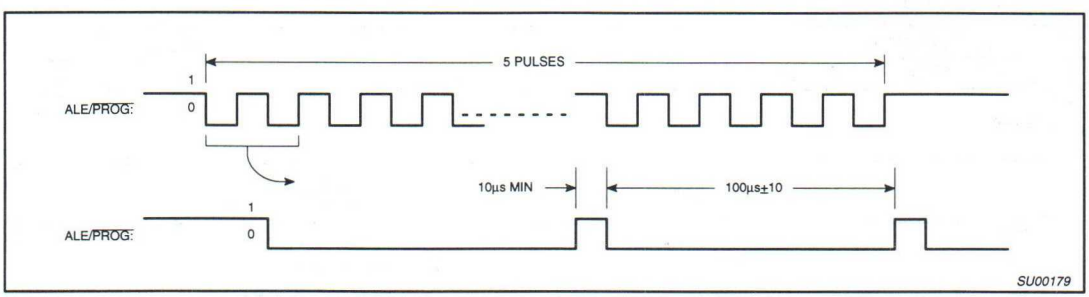



**Figure 30. PROG Waveform**

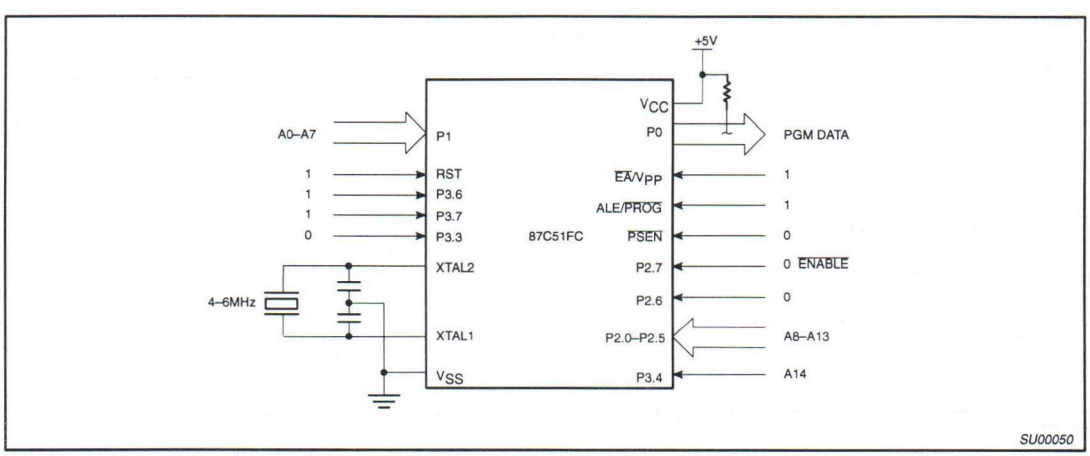


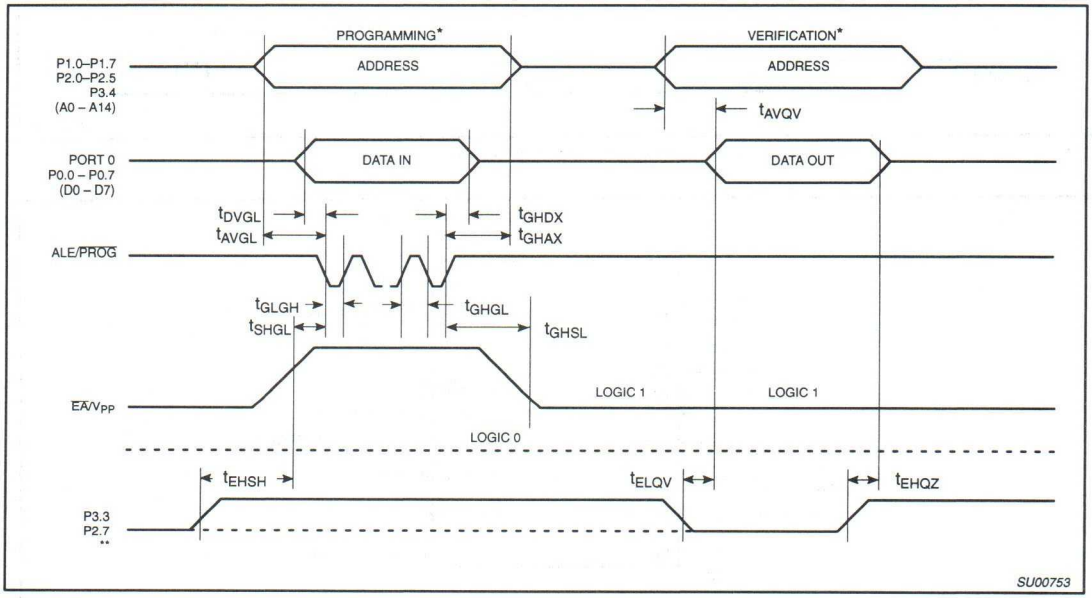

**Figure 31. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (See Figure 32)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 [1] | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |

**NOTE:**
1. Not tested.




**NOTES:**
* FOR PROGRAMMING VERIFICATION SEE FIGURE 29.
FOR VERIFICATION CONDITIONS SEE FIGURE 31.
** SEE TABLE 4.

**Figure 32. EPROM Programming and Verification**

# Low-voltage single-chip 8-bit microcontrollers 80CL31/80CL51

## FEATURES

- Full static 80C51 CPU
- 8-bit CPU, ROM, RAM, 1/0 in a single 40-lead DIL / mini-pack
- 4K x 8 ROM, expandable externally to 64K bytes
- 128 bytes RAM, expandable externally to 64K bytes
- Four 8-bit ports, 321/0 lines
- Two 16-bit timer / event counters
- External memory expandable up to 128K, external ROM up to 64K and / or RAM up to 64K
- On-chip oscillator suitable for RC, LC, quartz crystal or ceramic resonator
- Thirteen source, thirteen vector interrupt structure with two priority levels
- Full duplex serial port (UART)
- Enhanced architecture with:
  - non-page oriented instructions
  - direct addressing
  - four eight byte RAM register banks
  - stack depth up to 128 bytes
  - multiply, divide, subtract and compare instructions
- Power-Down and IDLE instructions
- Wake-up via external interrupts at Port 1
- Single supply voltage of 1.8V to 6.0V (5.0V ±10% for P80C51)
- Frequency range of 0 to 16MHz (3.5MHz to 16MHz for P80C51)
- Very low current consumption
- Operating temperature range: -40 to +85°C

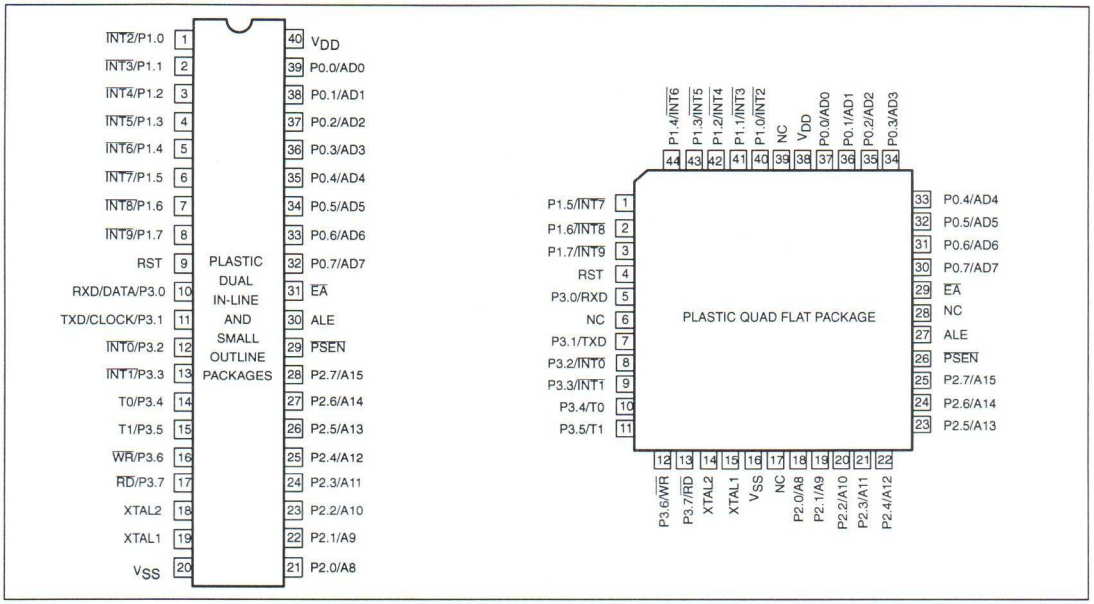
## DESCRIPTION

The 80CL51 is manufactured in an advanced CMOS technology. The instruction set of the 80CL51 is based on that of the 8051. The 80CL51 is a general purpose microcontroller especially suited for battery-powered applications. The device has low power consumption and a wide range of supply voltage. For emulation purposes, the 85CL000 (Piggy-back version) with 256 bytes of RAM is recommended. The 80CL51 has two software selectable modes of reduced activity for further power reduction: Idle and Power-down. The 80CL51 also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 46 two-byte, and 16 three-byte.

The P80CL31 is the ROMless version of the P80CL51. P80C51 is a 5V version of the low voltage P80CL51.

The P80CL31 is the ROMless version of the P80CL51. P80C51 is a 5V version of the low voltage P80CL51.

## PIN CONFIGURATIONS



## ORDERING INFORMATION

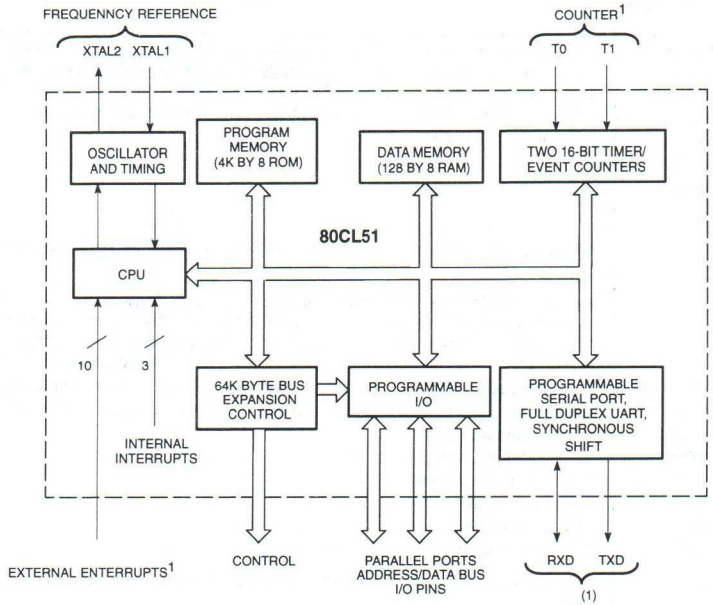
| PHILIPS PART ORDER NUMBER PART MARKING | | PHILIPS NORTH AMERICA [1] PART ORDER NUMBER | | TEMPERATURE RANGE °C AND PACKAGE | DRAWING NUMBER |
|---|---|---|---|---|---|
| ROMless | ROM | ROMless | ROM | | |
| P80CL31HFP | P80CL51HFP | P80CL31HFP N | P80CL51HFP N | –40 to +85; 40-lead Plastic Dual In-line Package (1.8V to 6V) | SOT129-1 |
| P80CL31HFT | P80CL51HFT | P80CL31HFT D | P80CL51HFT D | –40 to +85; 40-lead Plastic Small Outline Package (1.8V to 6V) | SOT158-1 |
| P80CL31HFH | P80CL51HFH | P80CL31HFH B | P80CL51HFH B | –40 to +85; 44-lead Plastic Quad Flat Package (1.8V to 6V) | SOT307-2 |
| | P80C51HFP | | P80C51HFP N | –40 to +85; 40-lead Plastic Dual In-line Package (5.0V ±10%) | SOT129-1 |
| | P80C51HFT | | P80C51HFT D | –40 to +85; 40-lead Plastic Small Outline Package (5.0V ±10%) | SOT158-1 |
| | P80C51HFH | | P80C51HFH B | –40 to +85; 44-lead Plastic Quad Flat Package (5.0V ±10%) | SOT307-2 |

**NOTE:**
1. Parts ordered by the Philips North America part number will be marked with the Philips part marking.

## PIN DESCRIPTIONS

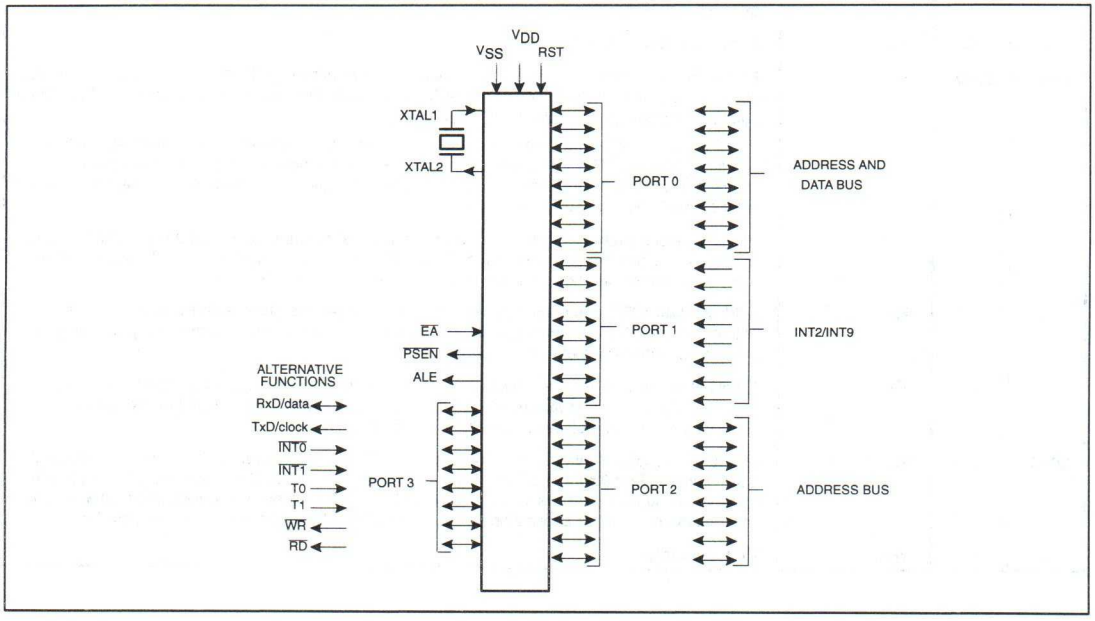
| PIN | | DESIGNATION | FUNCTION |
|---|---|---|---|
| QFP | DIP | | |
| 40 | 1 | P1.O/INT2 | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pullups. Port 1 pins that have 1s written to them are pulled HIGH by the internal pullups, and in that state can be used as inputs. The Port 1 output buffer can sink/source 4 LS TTL loads. As inputs, Port 1 pins that are externally pulled LOW will source current ($I_{IL}$ in the characteristics) due to the internal pullups. Port 1 also serves the alternative functions INT2 to INT9. |
| 41 | 2 | P1.1/INT3 | |
| 42 | 3 | P1.2/INT4 | |
| 43 | 4 | P1.3/INT5 | |
| 44 | 5 | P1.4/INT6 | |
| 1 | 6 | P1.5/INT7 | |
| 2 | 7 | P1.6/INT8 | |
| 3 | 8 | P1.7/INT9 | |
| 4 | 9 | RST | **Reset:** A high level on this pin for two machine cycles while the oscillator is running resets the device. |
| 5–13 | 10-17 | | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. The Port 3 output buffers can sink/source 4 LS TTL inputs. Port 3 pins that have 1s written to them are pulled HIGH by the internal pull ups, and in that state can be used as inputs. As inputs, Port 3 pins that are externally pulled LOW will source current ($I_{IL}$ in the characteristics) due to the internal pull ups. |
| 5 | 10 | P3.0/RXD/data | **RXD/data:** Serial port receiver data input (asynchronous)or data input/output (synchronous) |
| 7 | 11 | P3.1/TXD/clock | **TXD/clock:** Serial port transmitter data output (asynchronous) or clock output (synchronous) |
| 8 | 12 | P3.2/INT0 | **INT0:** External interrupt 0. |
| 9 | 13 | P3.3/INT1 | **INT1:** External interrupt 1. |
| 10 | 14 | P3.4/T0 | **T0:** Timer 0 external input. |
| 11 | 15 | P3.5/T1 | **T1:** Timer 1 external input. |
| 12 | 16 | P3.6/$\overline{WR}$ | **$\overline{WR}$**: External data memory write strobe. |
| 13 | 17 | P3.7/$\overline{RD}$ | **$\overline{RD}$**: External data memory read strobe. |
| 14 | 18 | XTAL2 | **Crystal output:** Output of the inverting amplifier of the oscillator. Left open when external clock is used.<br>**Crystal input:** Input to the inverting amplifier of the oscillator; also the input for an externally generated clock source. |
| 15 | 19 | XTAL1 | **Crystal input:** Input to the inverting amplifier of the oscillator; also the input for an externally generated clock source. |
| 16 | 20 | Vss | **Ground:** Circuit ground potential. |
| 18-25 | 21-28 | P2.0-P2.7 | **Port 2:** Port 2 is an 8-bit bidirectional 1/0 port with internal pullups. Port 2 pins that have 1s written to them are pulled HIGH by the internal pullups, and in that state can be used as inputs. The Port 2 output buffer can sink/source 4 LS TTL loads.<br>Port 2 emits the high-order address byte during accesses to external memory that use 1 6-bit addresses (MOVX @DPTR). In this application it uses the strong internal pullups when emitting 1s. During accesses to external memory that use 8-bit addresses (MOVX @Ri), Port 2 emits the contents of the P2 Special Function Register. |
| 26 | 29 | PSEN | **Program store enable output:** Read strobe to external program memory. When executing code out of external program memory, PSEN is activated twice each machine cycle. However, during each access to external data memory two PSEN activations are skipped. |
| 27 | 30 | ALE | **Address Latch Enable:** Output pulse for latching the low byte of the address during access to external memory. ALE is emitted at a constant rate of 1/6 of the oscillator frequency, and may be used for external timing or clocking purposes. |
| 29 | 31 | $\overline{EA}$ | **External Access:** When EA is held High the CPU executes out of internal program memory (unless the program counter exceeds 0FFFH). Holding EA LOW forces the CPU to execute out of external memory regardless of the value of the program counter. |
| 30-37 | 32-39 | P0.0-P00.7 | **Port 0:** Port 0 is an 8-bit open drain bidirectional I/O port. As an open drain output port it can sink 8 LS TTL loads. Port 0 pins that have 1s written to them float, and in that state will function as high impedance inputs. Port 0 is also the multiplexed low order address and data bus during access to external memory. In this application it uses strong internal pull-ups when emitting logic 1s. |
| 38 | 40 | $V_{DD}$ | **Power supply.** |

## BLOCK DIAGRAM




1. Pins shared with parallels ports pins.

## FUNCTIONAL DIAGRAM



## 1.0 FUNCTIONAL DESCRIPTION

### General

The 80CL51 is a stand-alone high-performance CMOS microcontroller designed for use in real-time applications such as instrumentation, industrial control, intelligent computer peripherals and consumer products.

The device provides hardware features, architectural enhancements and new instructions to function as a controller for applications requiring up to 64K bytes of program memory and/or up to 64K bytes of data storage.

The 80CL51 contains a non-volatile 4K byte × 8 read-only program memory; a static 128 byte × 8 read/write data memory; 32 1/0 lines; two 16-bit timer/event counters; a thirteen- source two priority-level, nested interrupt structure and on-chip oscillator and timing circuit.

The device has two software selectable modes of reduced activity for power reduction: IDLE and Power-down. The Idle mode freezes the CPU while allowing the RAM, timers, serial I/O and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative.

The P80C51 is a 5V version of the low voltage microcontroller P80CL51. Hereafter the generic term P80CL51 will be used for the functional description of both types. The special features of the P80C51 are handled in chapter 1.9.

### CPU timing

A machine cycle consists of a sequence of 6 states. Each state time lasts for two oscillator periods, thus a machine cycle takes 12 oscillator periods or 1µs if the oscillator frequency is 12MHz.

### 1.1 Memory organization

The 80CL51 has a 4K Program Memory (ROM) plus 128 bytes of Data Memory (RAM) on board. The device has separate address spaces for Program and Data Memory (see Memory Map). Using Ports P0 and P2, the 80CL51 can address up to 64K bytes of external memory. The CPU generates both read and write signals (RD and WR) for external Data Memory accesses, and the read strobe (PSEN) for external Program Memory.

#### 1.1.1 Program Memory

The 80CL51 contains 4K bytes of internal ROM. After reset the CPU begins execution at location 0000H. The lower 4K bytes of Program Memory can be implemented in either on- chip ROM or external Memory. If the EA pin is strapped to $V_{DD}$, then program memory fetches from addresses 000H through 0FFFH are directed to the internal ROM. Fetches from addresses 1000H through FFFFH are directed to external ROM. Program counter values greater than 0FFFH are automatically addressed to external memory regardless of the state of the EA pin.

#### 1.1.2 Data Memory

The 80CL51 contains 128 bytes of internal RAM and 25 Special Function Registers (SFR). The Memory Map below shows the internal Data Memory space divided into the Lower 128, the Upper 128, and the SFR space.

The lower 128 bytes of the internal RAM are organized as mapped in Figure 1. The lowest 32 bytes are grouped into 4 banks of 8 registers. Program instructions refer to these registers R0 through R7. Two bits in the Program Status Word select which register bank is in use. The next 16 bytes above the register banks form a block of bit-addressable memory space. The 128 bits in this area can be directly addressed by the single-bit manipulation instructions. The remaining registers (30H to 7FH) are directly and indirectly byte addressable.

#### 1.1.3 Special Function Registers

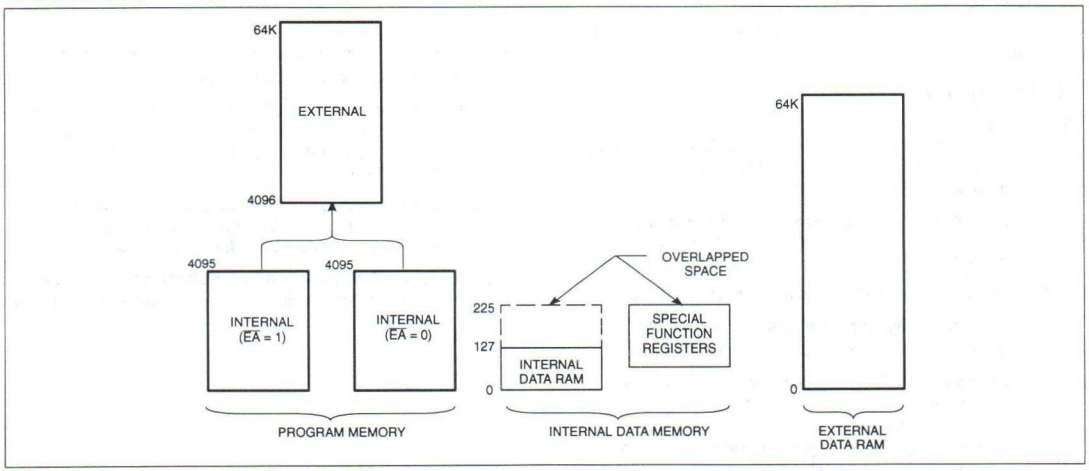
The upper 128 bytes are the address locations of the SFRs. Figure 2 shows the Special Function Register (SFR) space. SFRs include the port latches, timers, peripheral control, serial I/O registers, etc. These registers can only be accessed by direct addressing. There are 128 addressable locations in the SFR address space (SFRs with addresses divisible by eight).

#### 1.1.4 Addressing

The 80CL51 has five methods for addressing source operands:

- Register
- Direct
- Register-Indirect
- Immediate
- Base-Register-plus Index-Register-indirect

### MEMORY MAP



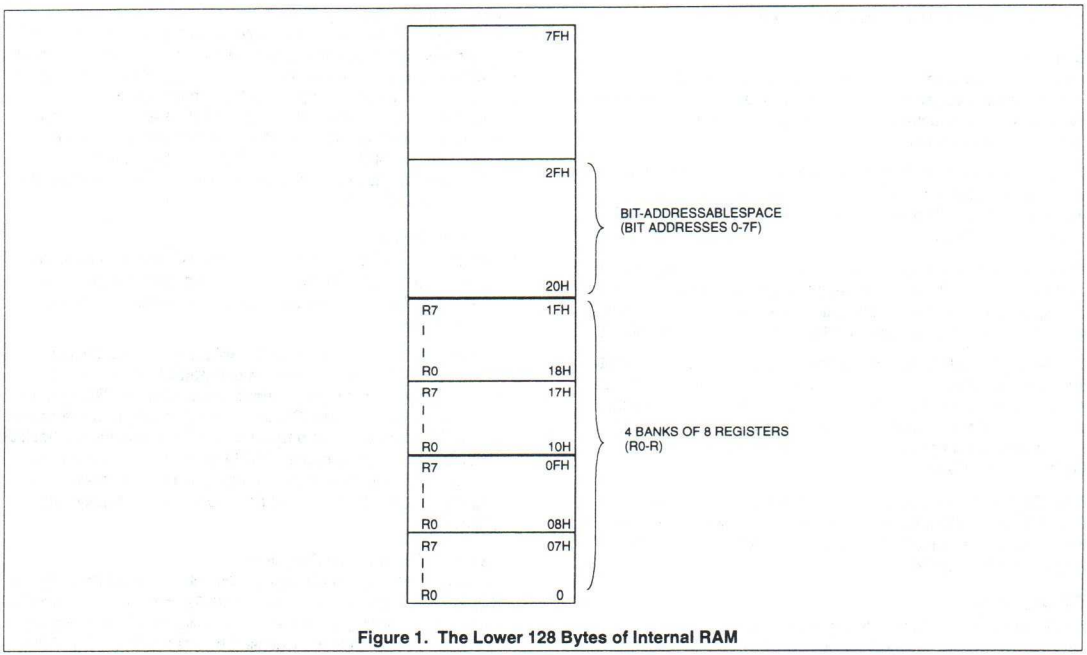



**Figure 1. The Lower 128 Bytes of Internal RAM**

The first three methods can be used for addressing destination operands. Most instructions have a "destination/source" filed that specifies data type, addressing methods and operands involved. For operations other than MOVs, the destination operand is also a source operand.

Access to memory addressing is as follows:

- Registers in one of the four register banks through register, direct or indirect.
- Internal RAM (128 bytes) through direct or register-indirect.
- Special Function Register through Direct.
- External data memory through Register-Indirect
- Program memory look-up tables through Base-Register-Plus Index-Register-Indirect.

## 1.2 I/O Facilities

### 1.2.1 Ports

The 80CL51 has 32 I/O lines treated as 32 individually addressable bits or as four parallel 8- bit addressable ports. Port 0, 1, 2 and 3 perform the following alternate functions:

Port 0: provides the multiplexed low-order address and data bus for expanding the device with standard memories and peripherals.

Port 1: provides the inputs for the external interrupts INT2/INT9.

Port 2: provides the high-order address when expanding the device with external program or data memory.

Port 3: pins can be configured individually to provide:
(1) external interrupt request inputs
(2) counter input
(3) control signals to read and write to external memories
(4) UART input and output

To enable a Port 3 pin alternate function, the Port 3 bit latch in its SFR must contain a logic 1.

Each port consists of a latch (Special Function Registers P0 to P3), an output driver and an input buffer. Ports 1,2,3 have internal pull ups. Figure 3(a) shows that the strong transistor p1 is turned on for only 2 oscillator periods after a 0-to-1 transition in the port latch. When on, it turns on p3 (a weak pull up) through the inverter. This inverter and p3 form a latch which hold the 1. In Port 0 the pull up p1 is only on when emitting 1s for external memory access. Writing a 1 to a Port 0 bit latch leaves both output transistors switched off so the pin can be used as a high-impedance input.

### 1.2.2 Port Options

The pins of port 1, port 2, and port 3 may be individually configured with one of the following options (see Figure 3):

Option 1: Standard Port; quasi-bidirectional I/O with pull up. The strong booster pull up p1 is turned on for two oscillator periods after a 0-to-1 transition in the port latch (see Figure 3(a)).

Option 2: Open drain; quasi-bidirectional I/O with n-channel open drain output. Use as an output requires the connection of an external pull up resistor (see Figure 3(c)).

Option 3: Push-Pull; output with drive capability in both polarities. Under this option, pins can only be used as outputs. See Figure 3(b).

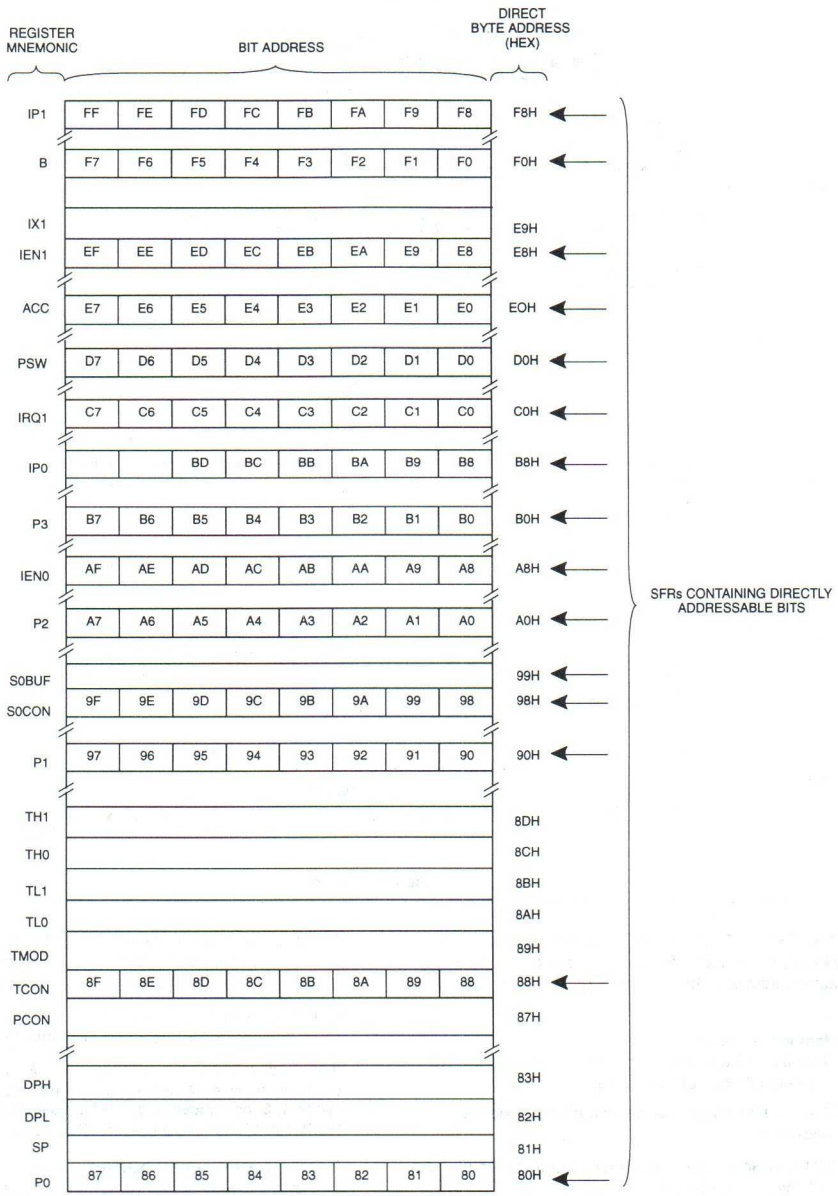



**Figure 2. Special Function Registers**

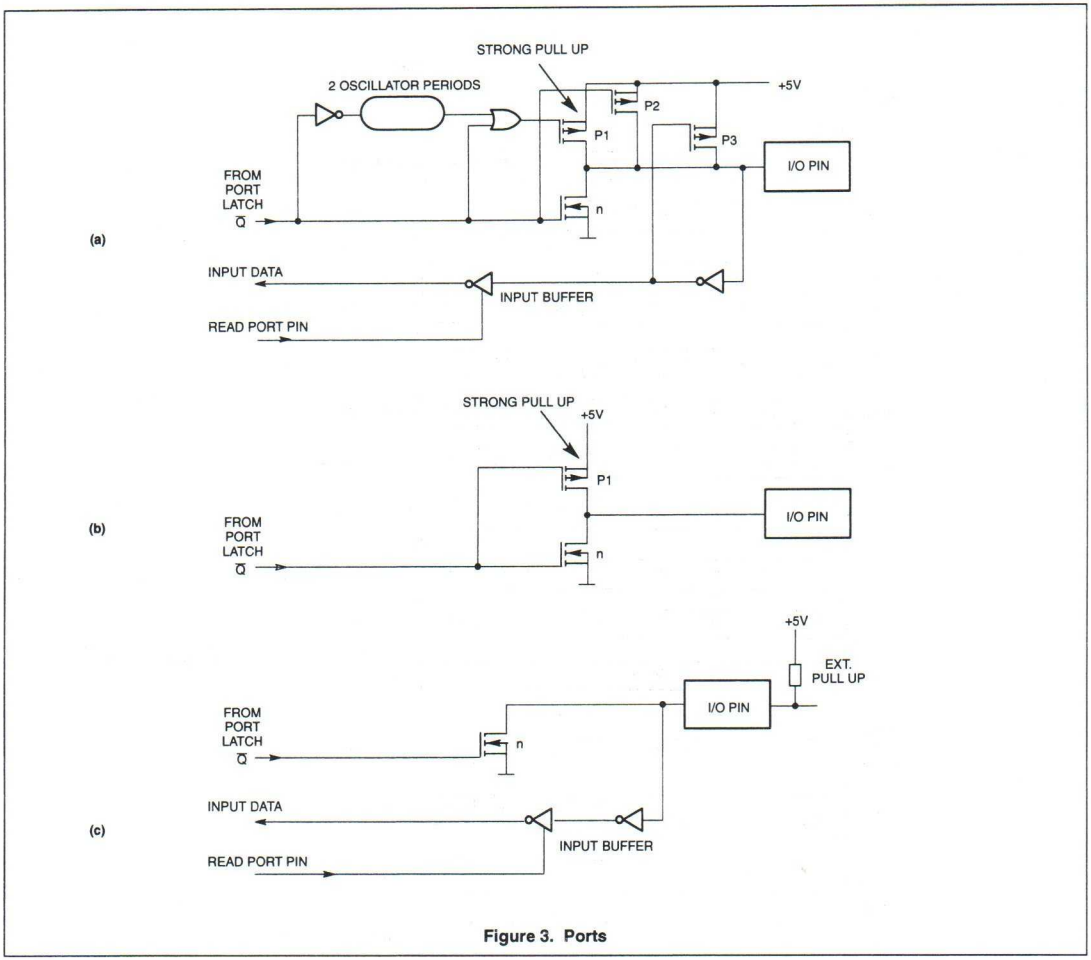



**Figure 3. Ports**

The definition of port options for port 0 is slightly different. Two cases have to be examined. First, accesses to external memory (EA=0 or access above the built -in memory boundary), second, I/O accesses.

**External Memory Accesses**

Option 1: True 0 and 1 are written as address to the external memory (strong pull up is used).

Option 2: An external pull up resistor is needed for external accesses.

Option 3: Not allowed for external memory access as the port can only be used as output.

**I/O Accesses**

Option 1: When writing a 1 to the port-latch, the strong pull up p1 will be on for 2 oscillator periods. No weak pull up exists. Without an external pull up, this option can be used as a high-impedance input.

Option 2: Open drain; quasi-bidirectional I/O with n-channel open drain output. Use as an output requires the connection of an external pull up resistor (see Figure 3(c)).

Option 3: Push-Pull; output with drive capability in both polarities. Under this option, pins can only be used as outputs.

Individual mask selection of the post-reset state is available on any of the above pins. Make your selection by appending "S" or "R" to option 1, 2, or 3 above (e.g. 1 S for a standard I/O to be set after RESET or 2R for an open-drain I/O to be reset after RESET).

## 1.3 Timer/event counter

The 80CL51 contains two 16-bit Timer/Counter registers, Timer 0 and Timer 1, which can perform the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupts requests

Timer 0 and Timer 1 can be independently programmed to operate as follows:

Mode 0 - 8-bit timer or counter with divide-by-32 prescaler

Mode 1 - 16-bit time-interval or event counter

Mode 2 - 8-bit time interval or event counter with automatic reload upon overflow

Mode 3 - Timer 0 establishes TL0 and TH0 as two separate counters.

In the "Timer" function, the register is incremented every machine cycle. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

In the "Counter" function, the register is incremented in response to a 1-to-0 transition. Since it takes 2 machine cycles (24 oscillator periods) to recognize a 1-to-0 transition, the maximum count rate is 1/24 of the oscillator frequency. To ensure a given level is sampled, it should be held for at least one full machine cycle.

## 1.4 Idle and Power-down operation

Idle mode operation permits the interrupt, serial port and timer blocks to continue functioning while the clock to the CPU is halted. The following functions remain active during Idle mode:

- Timer 0, Timer 1
- UART
- External interrupt

The Power-down operation freezes the oscillator. The Power-down mode can only be activated by setting the PD bit in the PCON register.

### 1.4.1 Power control register

Power-down and Idle modes are activated by software via the Special Function Register PCON. Its hardware address is 87H. PCON is byte addressable only.

| PCON | | |
|---|---|---|
| **BIT** | **POSITION** | **FUNCTION** |
| SMOD | PCON.7<br>PCON.4-PCON.6 | Double baud-rate bit, see description of the UART, chapter 1.5.<br>(reserved) |
| GF1 | PCON.3 | General purpose flag bit |
| GFO | PCON.2 | General purpose flag bit |
| PD | PCON.1 | Power-down activation bit |
| IDL | PCON.0 | Idle mode activation bit |

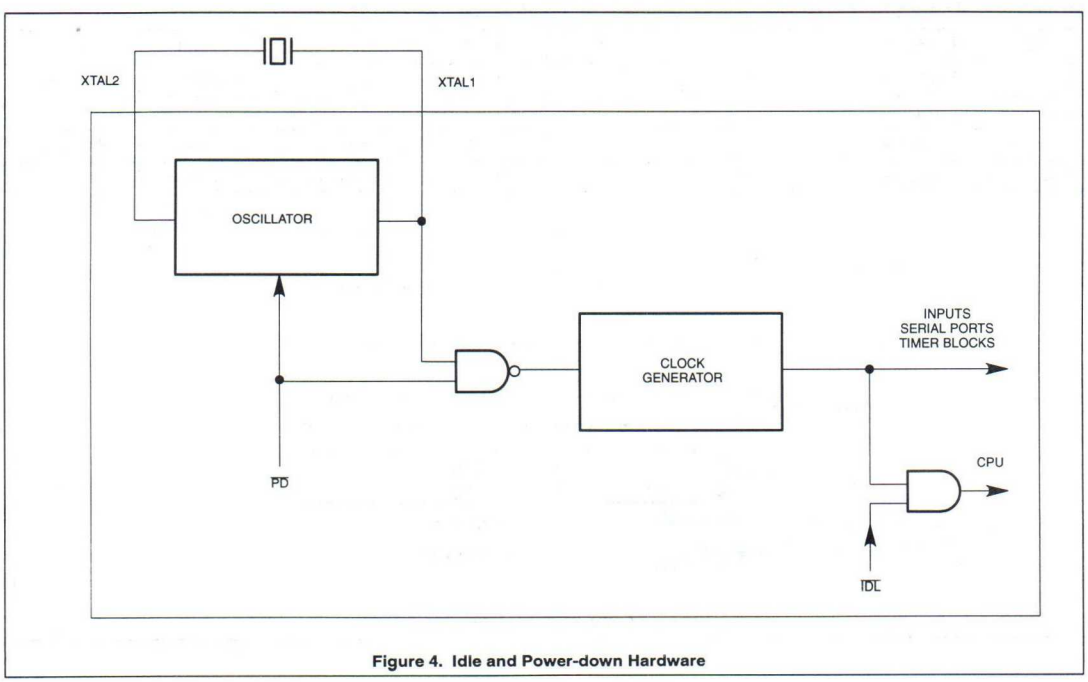



**Figure 4. Idle and Power-down Hardware**

### 1.4.2 Power-down mode

The instruction setting PCON.1 is the last executed prior to going into the Power-down mode. In Power-down mode the oscillator is stopped. The contents of the on-chip RAM and SFRs are preserved. The port pins output the values held by their respective SFRs. ALE and PSEN are held LOW.

In the Power-down mode $V_{DD}$ may be reduced to minimize power consumption. However, the supply voltage must not be reduced until Power-down mode is active, and must be restored before the hardware reset is applied and frees the oscillator. Reset must be held active until the oscillator has restarted and stabilized.

The wake-up operation after power-down in this controller has two basic approaches:

#### 1.4.2.1 Wake-up using INT2 to INT9

If INT2 to INT9 are enabled, the 80CL51 can be awakened from power-down mode with the external interrupts. To ensure that the oscillator is stable before the controller restarts, the internal clock will remain inactive for 1536 oscillator periods. This is controlled by an on-chip delay counter.

#### 1.4.2.2 Wake-up using RESET

To wake-up the 80CL51 the RESET pin has to be kept HIGH for a minimum of 24 oscillator periods. The on-chip delay counter is inactive. The user has to ensure that the oscillator is stable before any operation is attempted. Figure 5 illustrates the two possibilities for wake-up.

### 1.4.3 Idle mode

The instruction that sets PCON.0 is the last instruction executed before going into Idle mode. Once in the Idle mode, the internal clock is gated away from the CPU, but not from the Interrupt, Timer and Serial port functions. The CPU status is preserved along with the Stack Pointer, Program Counter, Program Status Word and Accumulator. The RAM and all other registers maintain their data during Idle mode. The port pins retain the logical states they held at Idle mode activation. ALE and PSEN hold at the logic HIGH level.

There are two methods used to terminate the Idle mode. Activation of any enabled interrupt will cause PCON to be cleared by hardware, terminating Idle mode. The interrupt is serviced, and following the instruction RETI, the next instruction to be executed will be the one following the instruction that put the device in the Idle mode.

Flag bits GF0 and GF1 may be used to determine whether the interrupt was received during normal execution or Idle mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When Idle mode is terminated by an interrupt, the service routine can examine the status of the flag bits.

The second method of terminating the Idle mode is with an external hardware reset. Since the oscillator is still running, the hardware reset is required to be active for only two machine cycles to complete the reset operation.

Reset redefines all SFRs, but does not affect the on-chip RAM.

The status of the external pins during Idle and Power-down mode is shown in Table 1. If the Power-down mode is activated while accessing external memory, port data held in the Special Function Register P2 is restored to Port 2. If the data is a logic 1, the port pin is held HIGH during the Power-down mode by the strong pull up transistor p1 (see Figure 3(a)).

**Table 1. Status of the External Pins During Idle and Power-down Mode**

| MODE | MEMORY | ALE | PSEN | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | internal | 1 | 1 | Port Data | Port Data | Port Data | Port Data |
| Idle | external | 1 | 1 | Floating | Port Data | Address | Port Data |
| Power-down | internal | 0 | 0 | Port Data | Port Data | Port Data | Port Data |
| Power-down | external | 0 | 0 | Floating | Port Data | Port Data | Port Data |

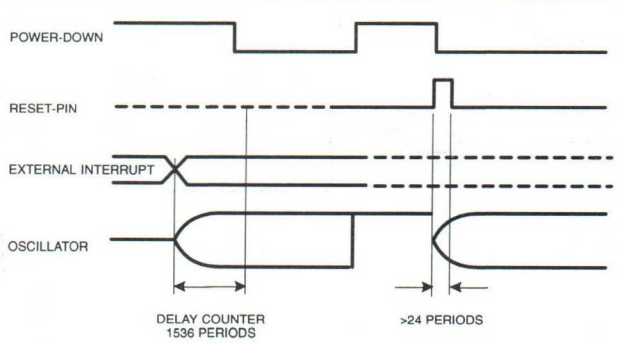



**Figure 5. Wake-up Operation**

## 1.5 Standard serial interface SI0: UART

This serial port is full duplex, meaning it can transmit and receive simultaneously. It is also receive-buffered, meaning it can commence reception of a second byte before a previously received byte has been read from the register. (However, if the first byte still hasn't been read by the time reception of the second byte is complete, one of the bytes will be lost). The serial port receive and transmit registers are both accessed at Special Function Register S0BUF. Writing to S0BUF loads the transmit register, and reading S0BUF loads the transmit register, and reading S0BUF accesses a physically separate receive register.

The serial port can operate in 4 modes:

Mode 0: Serial data enters and exits through RxD. TxD outputs the shift clock. 8 bits are transmitted/ received (LSB first). The baud is fixed at 1/12 the oscillator frequency.

Mode 1: 10 bits are transmitted (through TxD) or received (through RxD): a start bit (0), 8 data bits (LSB first), and a stop bit (1). On receive, the stop bit goes into RB8 in Special Function Register SCON. The baud rate is variable.

Mode 2: 11 bits are transmitted (through TxD) or received (through RxD): start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). On Transmit, the 9th data bit (TB8 in SCON) can be assigned the value of 0 or 1. Or, for example, the parity bit (P, in the PSW) could be moved into TB8. On receive, the 9th data bit goes into RB8 in Special Function Register SCON, while the stop bit is ignored. The baud rate is programmable to either 1/32 or 1/64 the oscillator frequency.

Mode 3: 11 bits are transmitted (through TxD) or received (through RxD): a start bit (0), 8 data bits (LSB first), a programmable 9th data bit and a stop bit (1). In fact, Mode 3 is the same as Mode 2 in all respects except baud rate. The baud rate in Mode 3 is variable.

In all four modes, transmission is initiated by any instruction that uses S0BUF as a destination register. Reception is initiated in Mode 0 by the condition RI = 0 and REN = 1. Reception is initiated in the other modes by the incoming start bit if REN = 1.

### 1.5.1 Multiprocessor communications

Modes 2 and 3 have a special provision for multiprocessor communications. In these modes, 9 data bits are received. The 9th one goes into RB8. Then comes a stop bit. The port can be programmed such that when the stop bit is received, the serial port interrupt will be activated only if RB8 = 1. This feature is enabled by setting bit SM2 in SCON. A way to use this feature in multiprocessor systems is as follows:

When the master processor wants to transmit a block of data to one of several slaves, it first sends out an address byte which identifies the target slave. An address byte differs from a data byte in that the 9th bit is 1 in an address byte and 0 in a data byte. With SM2 = 1, no slave will be interrupted by a data byte. An address byte, however, will interrupt all slaves, so that each slave can examine the received byte and see if it is being addressed. The addressed slave will clear its SM2 bit and prepare to receive the data bytes that will be coming. The slaves that weren't being addressed leave their SM2s set and go on about their business, ignoring the coming data bytes.

SM2 has no effect in Mode 0, and in Mode 1 can be used to check the validity of the stop bit. In a Mode 1 reception, if SM2 = 1, the receive interrupt will not be activated unless a valid stop bit is received.

### 1.5.2 Serial port control register

The serial port control and status register is the Special Function Register S0CON, shown in Figure 6. The register contains not only the mode selection bits, but also the 9th data bit for transmit and receive (TB8 and RB8), and the serial port interrupt bits (T1 and R1). See next page.

**Baud Rates**

The baud rate in Mode 0 is fixed: Mode 0 Baud Rate = Oscillator Frequency /12. The baud rate in Mode 2 depends on the value of bit SMOD in Special Function Register PCON. If SMOD = 0 (which is the value on reset), the baud rate is 1/64 the oscillator frequency. If SMOD = 1, the baud rate is 1/32 the oscillator frequency.

Mode 2 Baud Rate = $(2^{SMOD}/64)$(Oscillator Frequency)

The baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate.

**Using Timer 1 to generate baud rates**

When Timer 1 is used as the baud rate generator, the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate and the value of SMOD as follows:

$(2^{SMOD}/32)$(Timer 1 Overflow Rate)

The Timer 1 interrupt should be disabled in this application. The Timer itself can be configured for either "timer" or "counter" operation, and in any of its 3 running modes. In the most typical applications, it is configured for "timer operation, in the auto-reload mode (high nibble of TMOD = 0010B). In that case the baud rate is given by the formula:

Mode 1, 3 Baud Rate =

$\{(2^{SMOD}/32)$ (Oscillator Frequency)$\}$ / {12 (256 - (TH 1 )}

One can achieve very low baud rates with Timer 1 by leaving the Timer 1 interrupt enabled, and configuring this Timer to run as a 16-bit timer (high nibble of TMOD = 0001B), and using the Timer 1 interrupt to do a 16-bit software reload. Table 2 lists various commonly used baud rates and how they can be obtained from Timer 1.

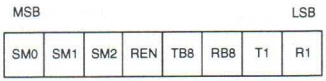
**More about Mode 0**

Figure 7 shows a simplified functional diagram of the serial port in Mode 0, and associated timing. Transmission is initiated by any instruction that uses S0BUF as a destination register. The "write to S0BUF" signal at S6P2 also loads a 1 into the 9th position of the transmit shift register and tells the TX Control block to commence a transmission. The internal timing is such that the one full machine cycle will elapse between "write to S0BUF", and activation of SEND.

SEND enables the output of the shift register to the alternate output function line of P3.0 and also enables SHIFT CLOCK to the alternate output function line of P3.1. SHIFT CLOCK is low during S3, S4, and S5 of every machine cycle, and high during S6, S1 and S2. At S6P2 of every machine cycle in which SEND is active, the contents of the transmit shift are shifted to the right one position.

As data bits shift out to the right, zeros come in from the left. When the MSB of the data byte is at the output position of the shift register, then the 1 that was initially loaded into the 9th position is just to the left of the MSB, and all positions to the left of that contain zeros. This condition flags the TX Control block to do one last shift and then deactivate SEND and set T1. Both of these actions occur at S1P1 of the 10th machine cycle after "write to S0BUF".

Reception is initiated by the condition REN = 1 and R1 = 0. At S6P2 of the next machine cycle, the RX Control unit writes the bits 11111110 to the receive shift register, and in the next clock phase activates RECEIVE.




Where SM0, SM1 specify the serial port mode, as follows:

| SM0 | SM1 | Mode | Description | Baud Rate |
|---|---|---|---|---|
| 0 | 0 | 0 | shift register | $f_{OSC}$/ 12 |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{OSC}$/64 or $f_{OSC}$/32 |
| 1 | 1 | 3 | 9-bit variable UART | |

**SM2** Enables the multiprocessor communication feature in Modes 2 and 3. In Mode 2 or 3, if SM2 is set to 1 then RI will not be activated if the received 9th data bit (RB8) is 0. In Mode 1, if SM2=1 then R1 will not be activated if a valid stopbit was not received. In Mode 0, SM2 should be 0.

**REN** Enables serial reception. Set by software to enable reception. Clear by software to disable reception.

**TB8** Is the 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired.

**RB8** In Modes 2 and 3, is the 9th data bit that was received. In Mode 1, it SM2=0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used.

**TI** Is transmit interrupt flag. Set by hardware at the end of the 8th time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software.

**RI** Receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception except (see SM2). Must be cleared by software.

**Figure 6. Serial Port control (SCON) Register**

**Table 2. Timer 1 Generated Commonly Used Baud Rates**

| | | | | TIMER 1 | |
|---|---|---|---|---|---|
| **BAUD RATE** | **$f_{OSC}$** | **SMOD** | **C/T** | **MODE** | **RELOAD VALUE** |
| Mode 0 Max: 1.33 Mb/s | 16 MHz | x | x | x | x |
| Mode 2 Max: 500 Kb/s | 16 MHz | 1 | x | x | x |
| Modes 1,3: 83.3 Kb/s | 16 MHz | 1 | 0 | 2 | FFH |
| 19.2 Kb/s | 11.059 MHz | 1 | 0 | 2 | FDH |
| 9.6 Kb/s | 11.059 MHz | 0 | 0 | 2 | FDH |
| 4.8 Kb/s | 11.059 MHz | 0 | 0 | 2 | FAH |
| 2.4 Kb/s | 11.059 MHz | 0 | 0 | 2 | F4H |
| 1.2 Kb/s | 11.059 MHz | 0 | 0 | 2 | E8H |
| 137.5 Kb/s | 11.986 MHz | 0 | 0 | 2 | 1DH |
| 110 | 6 MHz | 0 | 0 | 2 | 72H |
| 110 | 12 MHz | 0 | 0 | 1 | FEEBH |

RECEIVE enables SHIFT CLOCK to the alternate output function line of P3.1. SHIFT Clock makes transitions at S3P1 and S6P1 of every machine cycle. at S6P2 of every machine cycle in which RECEIVE is active, the contents of the receive shift register are shifted to the left one position. The value that comes in from the right is the value that was sampled at the P3.0 pin at S5P2 of the same machine cycle.

As data bits come in from the right, 1s shift out to the left. When the 0 that was initially loaded into the right-most position arrives at the left-most position in the shift register, it flags the RX Control block to do one last shift and load S0BUF. At S1P1 of the 10th machine cycle after the write to SCON that cleared RI, RECEIVE is cleared as RI is set.

**More about Mode 1**

Ten bits are transmitted (through TxD), or received (through RxD): a start bit (0), 8 data bits (LSB first), and a stop bit (1 ). On receive, the stop bit goes into RB8 in SCON. In the 8051 the baud rate is determined by the Timer 1 overflow rate.

Figure 8 shows a simplified functional diagram of the serial port in Mode 1, and associated timings for transmit/receive.

Transmission is initiated by any instruction that uses S0BUF as a destination register. The "write to S0BUF" signal also loads a 1 into the 9th bit position of the transmit shift register and flags the TX Control unit that a transmission is requested. Transmission actually commences at S1P1 of the machine cycle following the next rollover in the divide-by-16 counter. (Thus, the bit times are synchronized to the divide-by-16 counter, not to the "write to S0BUF" signal).

The transmission begins with activation of SEND which sends the start bit to pin TxD. One bit time later, DATA is activated, enabling the transmission of the output bit of the transmit shift register to TxD. The first shift pulse occurs one bit time after that.

As data bits shift out to the right, zeros are clocked in from the left. When the MSB of the data byte is at the output position of the shift register, then the 1 that was initially loaded into the 9th position is just to the left of the MSB, and all positions to the left of that contain zeros. This condition flags the TX Control unit to do one last shift and then deactivate SEND and set TI. This occurs at the 10th divide-by-16 rollover after "write to S0BUF". Reception is initiated by a detected 1 -to-0 transition at RxD. For this purpose RxD is sampled at a rate of 16 times whatever baud rate has been established. When a transition is detected, the divide-by-16 counter is immediately reset, and 1FFH is written into the input shift register. Resetting the divide-by-16 counter aligns its rollovers with the boundaries of the incoming bit times. The 16 states of the counter divide each bit time into 16th. At the 7th, 8th, and 9th counter states of each bit time, the bit detector samples the value of RxD. The value accepted is the value that was seen in at least 2 of the 3 samples. This is done for noise rejection. If the value accepted during the first bit time is not 0, the receive circuits are reset and the unit goes back to looking for another 1-to-0 transition. This is to provide rejection of false start bits. If the start bit proves valid, it is shifted into the input shift register, and reception of the rest of the frame will proceed.

As data bits come in from the right, 1s shift out to the left. When the start bit arrives at the left-most position in the shift register, (which in mode 1 is a 9-bit register), it flags the RX Control block to do one last shift, loads S0BUF and RB8, and set RI. The signal to load S0BUF and RB8, and to set RI, will generated if, and only if, the following conditions are met at the time the final shift pulse is generated.

1. R1 = 0, and
2. Either SM2 = 0, or the received stop bit = 1

If either of these two conditions is not met, the received frame is irretrievably lost. If both conditions are met, the stop bit goes into RB8, the 8 data bits go into S0BUF, and RI is activated. At this time, whether the above conditions are met or not, the unit goes back to looking for a 1-to-0 transition in RxD.

**More about modes 2 and 3**

Eleven bits are transmitted (through TxD), or received (through RxD): a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). On transmit, the 9th data bit (TB8) can be assigned the value of 0 or 1. On receive, the 9th data bit goes into RB8 in SCON. The baud rate is programmable to either 1/32 or 1/64 the oscillator frequency in Mode 2. Mode 3 may have a variable baud rate generated from Timer 1.

Figures 9 and 10 show a functional diagram of the serial port in Modes 2 and 3. The receive portion is exactly the same as in Mode 1. The transmit portion differs from Mode 1 only in the 9th bit of the transmit shift register.

Transmission is initiated by any instruction that uses S0BUF as a destination register. The "write to S0BUF" signal also loads TB8 into the 9th bit position of the transmit shift register and flags the TX Control unit that a transmission is requested. Transmission commences at S1P1 of the machine cycle following the next rollover in the divide-by-16 counter (thus, the bit times are synchronized to the divide-by-16 counter, not to the "write to S0BUF" signal). The transmission begins with activation of SEND, which puts the start bit at TxD. One bit time later, DATA is activated, which enables the output bit of the transmit shift register to TxD. One bit time later, DATA is activated, which enables the output bit of the transmit shift register to TxD. The first shift pulse occurs one bit time after that. The first shift clocks a 1 (the stop bit) into the 9th bit position of the shift register. Thereafter, only zeros are clocked in. Thus, as data bits shift out to the right, zeros are clocked in from the left. Then TB8 is at the output position of the shift register, then the stop bit is just to the left of TB8, and all positions to the left of that contains zeros. This condition flags the TX Control unit to do one last shift and then deactivate SEND and set TI. This occurs at the 11th divide-by-16 rollover after "write to S0BUF".

Reception is initiated by a detected 1-to-0 transition at RxD. For this purpose RxD is sampled at a rate of 16 times whatever baud rate has been established. When a transition is detected, the divide-by-16 counter is immediately reset, and 1FFFH is written to the input shift register.

At the 7th, 8th and 9th counter states of each bit time, the bit detector samples the value of RxD. The value accepted is the value that was seen in at least 2 of the 3 samples. If the value accepted during the first bit time is not 0, the receive circuits are reset and the unit goes back to looking for another 1-to-0 transition. If the start bit proves valid, it is shifted into the input shift register, and reception of the rest of the frame will proceed. As data bits come in from the right, 1s shift out to the left. When the start bit arrives at the left-most position in the shift register (which in Modes 2 and 3 is a 9-bit register), it flags the RX Control block to do one last shift, load S0BUF and RB8, and set RI.

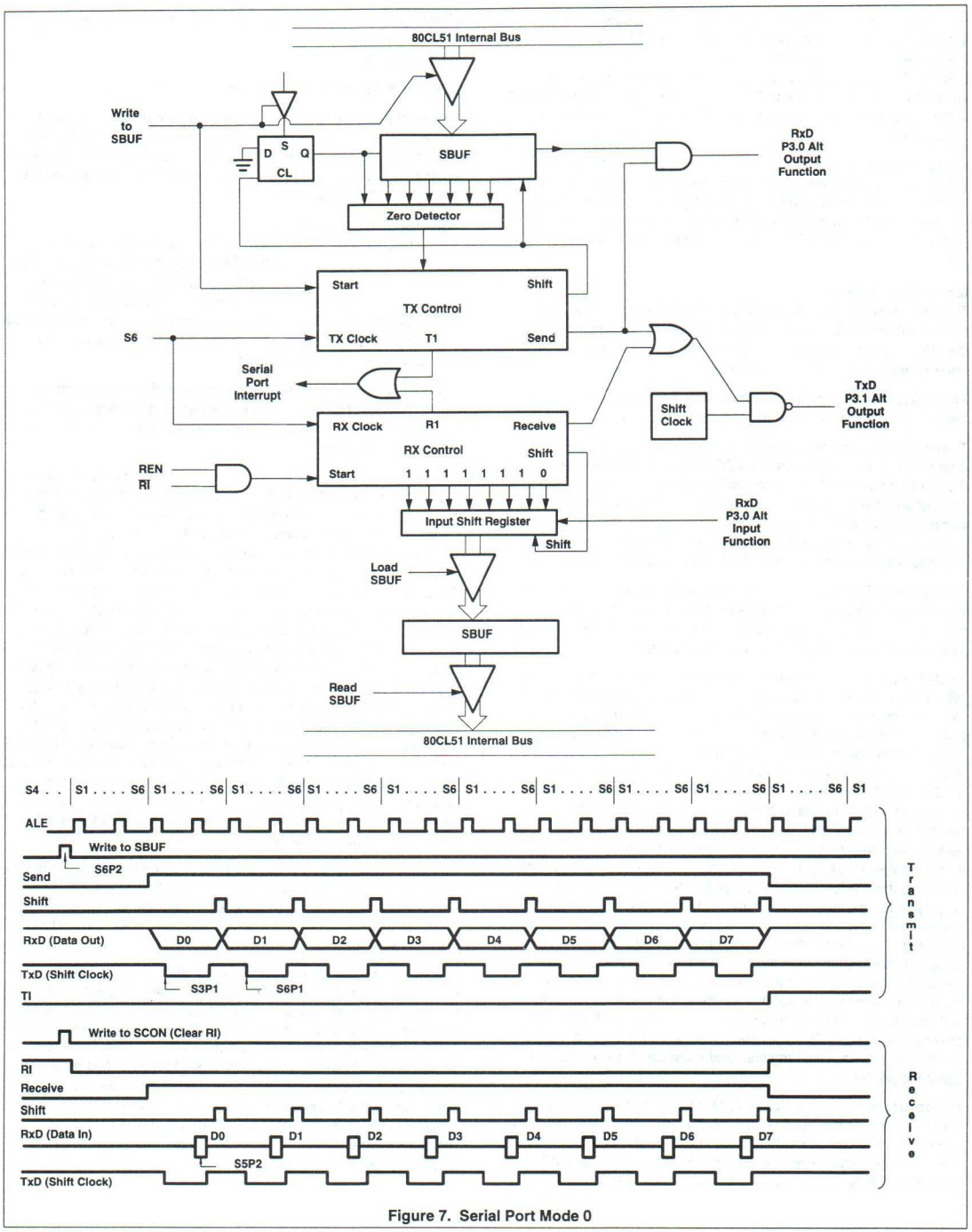



**Figure 7. Serial Port Mode 0**

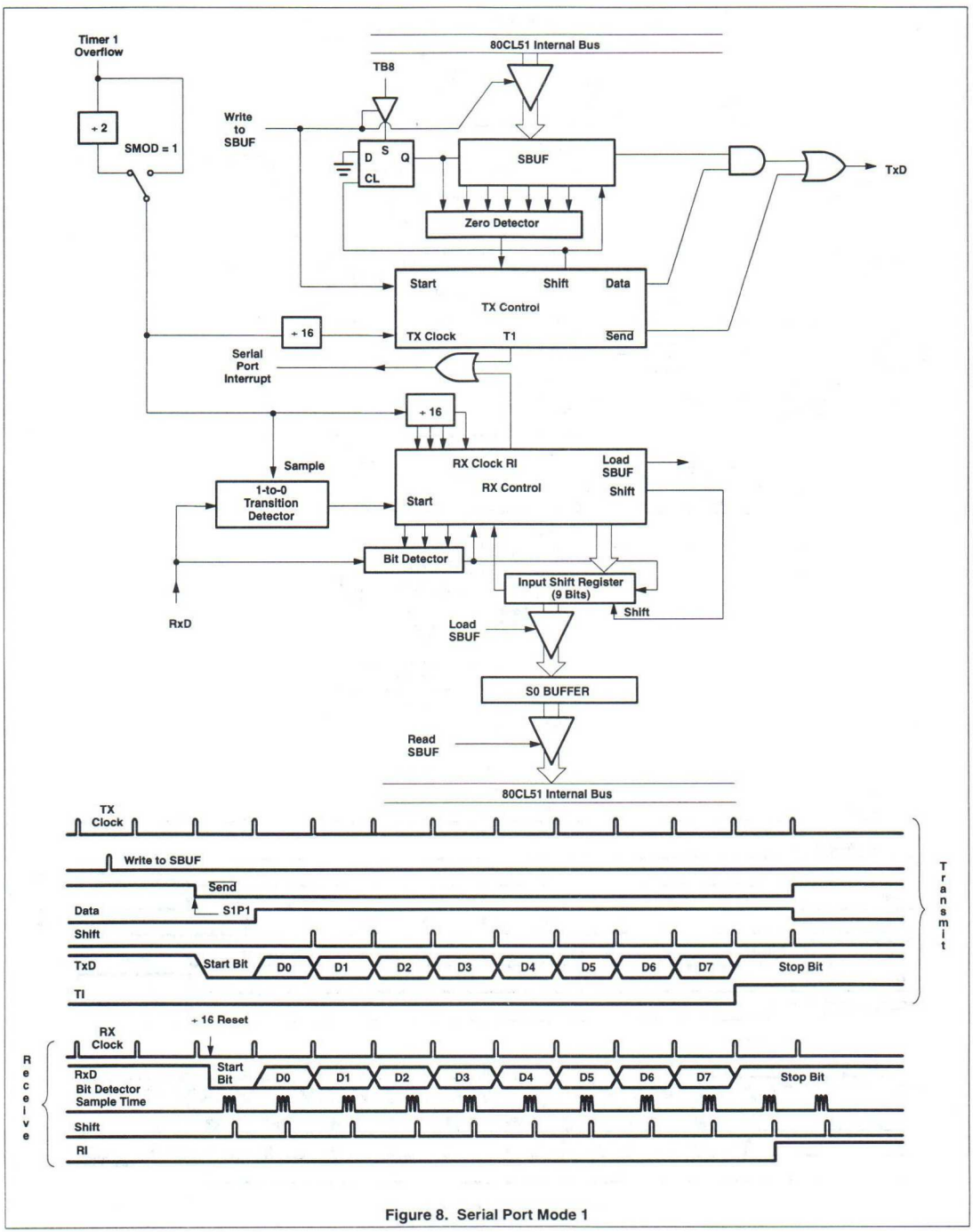



**Figure 8. Serial Port Mode 1**

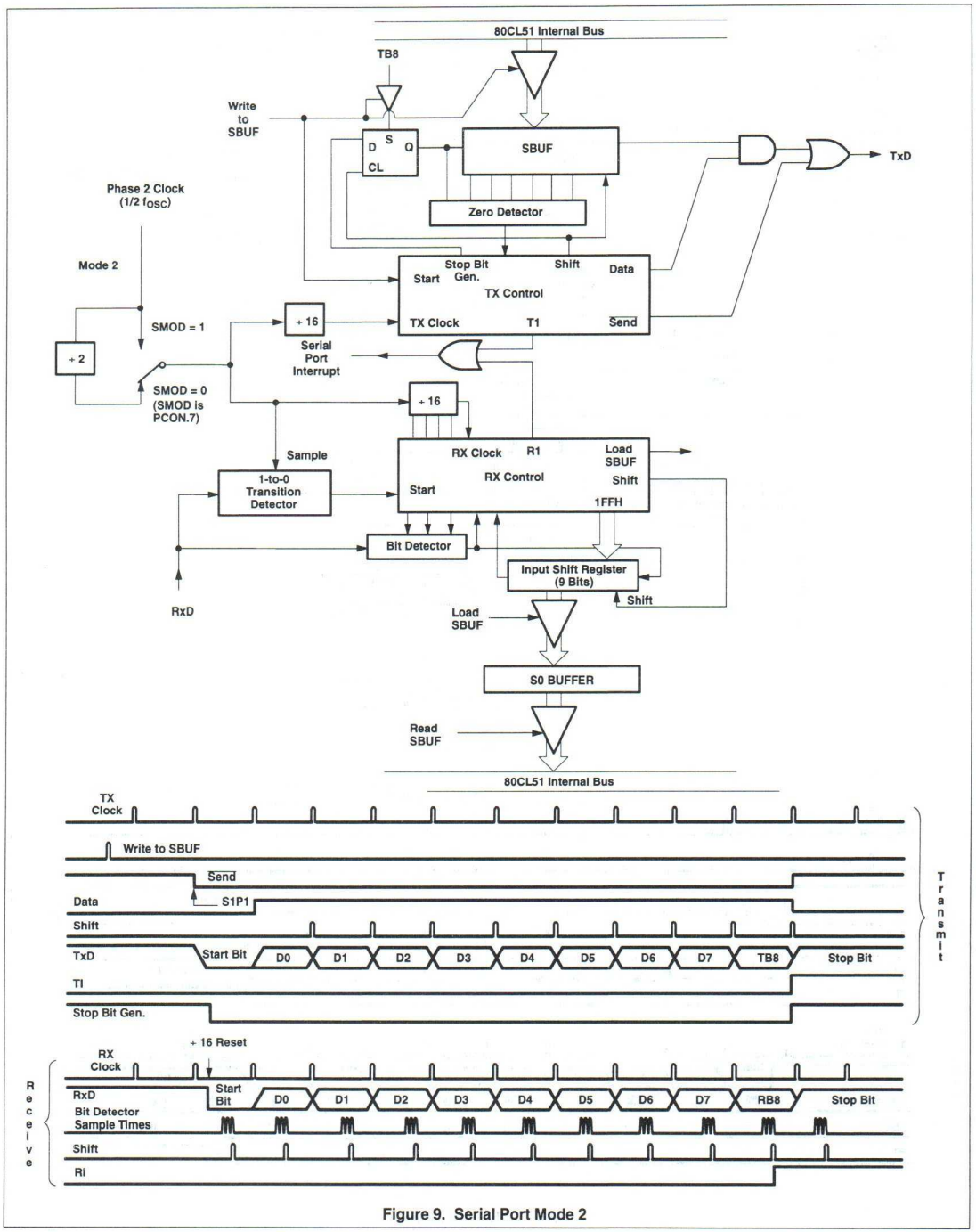



**Figure 9. Serial Port Mode 2**

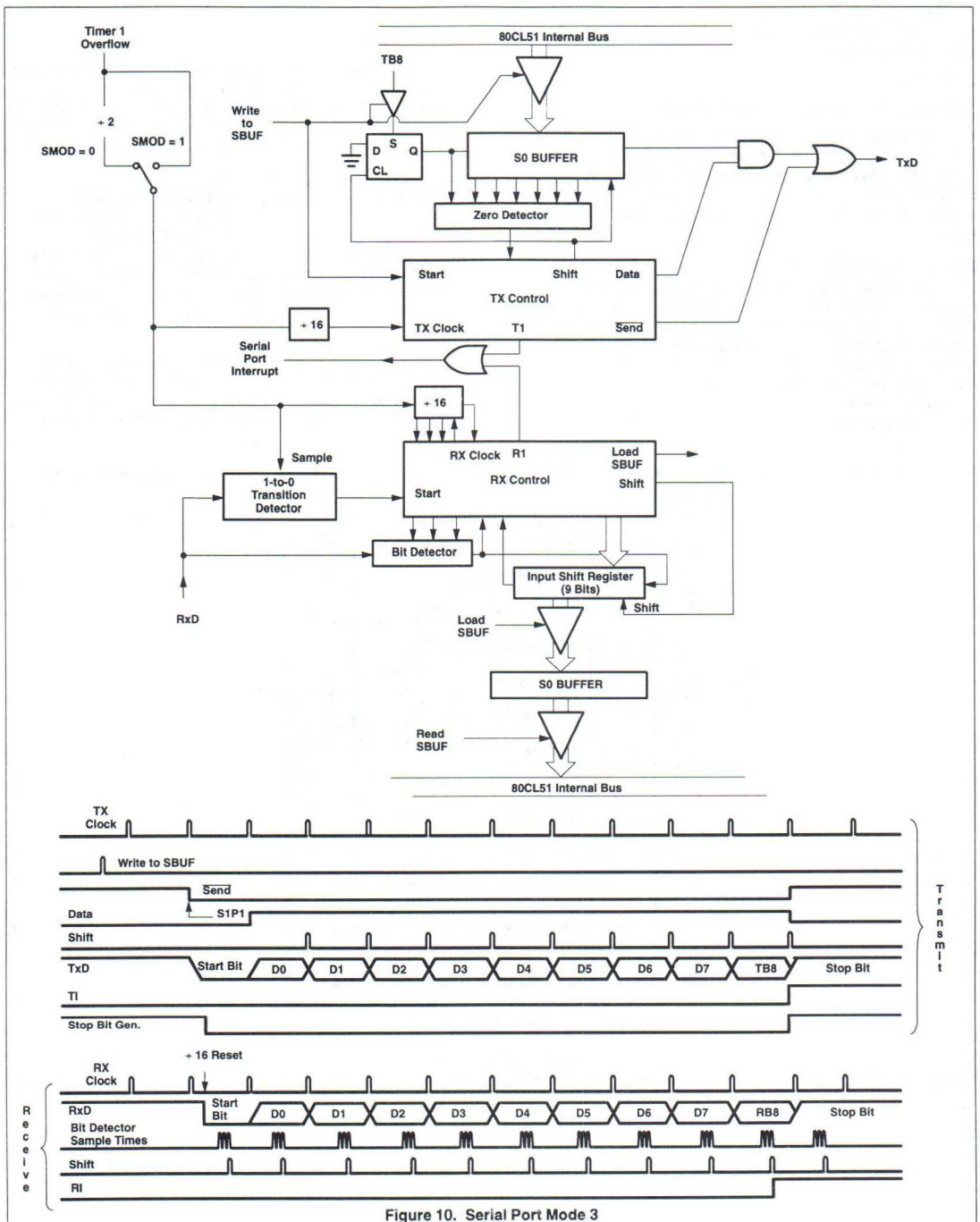



**Figure 10. Serial Port Mode 3**

The signal to load S0BUF and RB8, and to set RI, will be generated if, and only if, the following conditions are met at the time the final shift pulse is generated.

1. RI = 0, and
2. Either SM2 = 0 or the received 9th data bit = 1

If either of these conditions is not met, the received frame is irretrievably lost, and RI is not set. If both conditions are met, the received 9th data bit goes into RB8, and the first 8 data bits 90 into S0BUF. One bit time later, whether the above conditions were met or not, the unit goes back to looking for a 1-to-0 transition at the RxD input.

## 1.6 Interrupt System

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronous to do execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution, a multiple-source, two-priority-level, nested interrupt system is provided. The 80CL51 acknowledges interrupt requests from thirteen sources as follows:

- INT0 and INT1
- Timer 0 and Timer 1
- UART serial I/O
- INT2 to INT9 (Port 1)

Each interrupt vectors to a separate location in program memory for its service routine. Each source can be individually enabled or disabled by corresponding bits in the Interrupt Enable Registers (IE, IEO). The priority level is selected via the Interrupt Priority register (IP0, IP1). All enabled sources can be globally disabled or enabled.

### 1.6.1 External Interrupts INT2/INT9

Port 1 lines serve an alternative purpose as eight additional interrupts INT2 to INT9. When enabled, each of these lines may "wake-up" the device from Power-down mode. Using the IX1 register, each pin may be initialized to either active HIGH or LOW. IRQ1 is the interrupt request flag register. Each flag, if the interrupt is enabled, will be set on an interrupt request but must be cleared by software, i.e. via the interrupt software or when the interrupt is disabled.

The Port 1 interrupts are level sensitive. A Port 1 interrupt will be recognized when a level (HIGH or LOW depending on Interrupt Polarity Register IX1) on P1x is held active for at least one machine cycle. The Interrupt Request is not served until the next machine cycle.

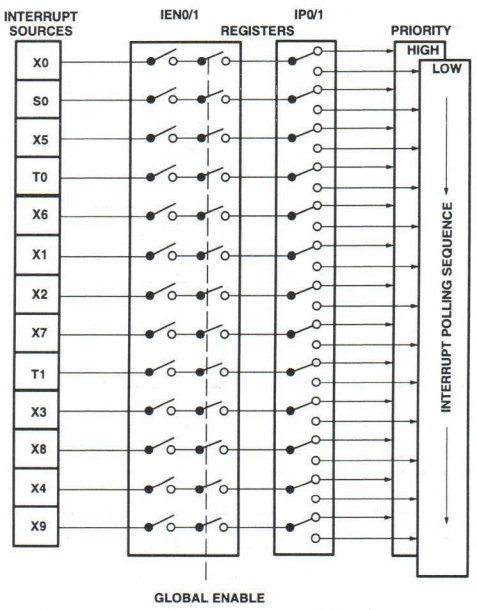



**Figure 11. Interrupt System**

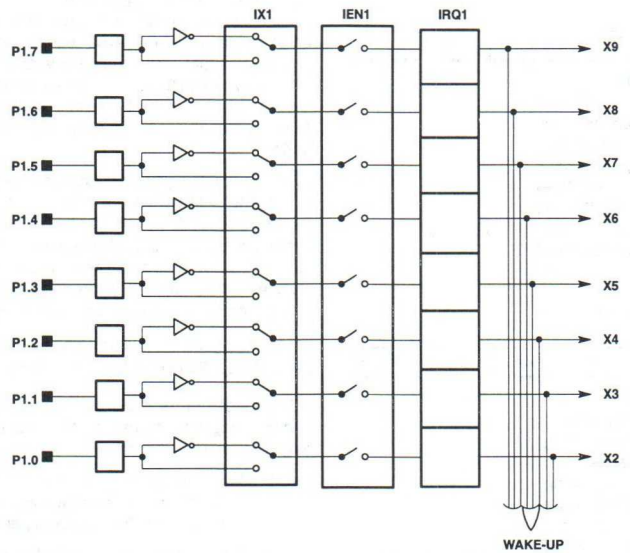



**Figure 12. External Interrupt Configuration**

**Interrupt enable register IEN0, IEN1**

**IEN0 (A8H)**

| EA | - | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 |
|---|---|---|---|---|---|---|---|

| Bit | Symbol | Function |
|---|---|---|
| IEN0.7 | EA | General enable/disable control<br>0 = no interrupt is enabled<br>1 = any individually enabled interrupt will be accepted |
| IEN0.6 | - | Unused |
| IEN0.5 | ES1 | Unused |
| IEN0.4 | ES0 | Enable UART SIO interrupt |
| IEN0.3 | ET1 | Enable timer T1 interrupt |
| IEN0.2 | EX1 | Enable external interrupt |
| IEN0.1 | ET0 | Enable Timer T0 interrupt |
| IEN0.0 | EX0 | Enable external interrupt 0 |

**IEN1 (E8H)**

| EX9 | EX8 | EX7 | EX6 | EX5 | EX4 | EX3 | EX2 |
|---|---|---|---|---|---|---|---|

| Bit | Symbol | Function |
|---|---|---|
| IEN1.7 | EX9 | Enable external interrupt 9 |
| IEN1.6 | EX8 | Enable external interrupt 8 |
| IEN1.5 | EX7 | Enable external interrupt 7 |
| IEN1.4 | EX6 | Enable external interrupt 6 |
| IEN1.3 | EX5 | Enable external interrupt 5 |
| IEN1.2 | EX4 | Enable external interrupt 4 |
| IEN1.1 | EX3 | Enable external interrupt 3 |
| IEN1.0 | EX2 | Enable external interrupt 2 |

where 0 = interrupt disabled
1 = interrupt enabled

**Interrupt priority register IP0, IP1**

**IP0 (B8H)**

| - | - | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 |
|---|---|---|---|---|---|---|---|

| Bit | Symbol | Function |
|---|---|---|
| IP0.7 | - | Unused |
| IP0.6 | - | Unused |
| IP0.5 | PS1 | Unused |
| IP0.4 | PS0 | UART SIO interrupt |
| IP0.3 | PT1 | Timer 1 interrupt priority level |
| IP0.2 | PX1 | External interrupt 1 priority level |
| IP0.1 | PT0 | Timer 0 interrupt priority level |
| IP0.0 | PX0 | External interrupt 0 priority level |

**IP1 (B8H)**

| PX9 | PX8 | PX7 | PX6 | PX5 | PX4 | PX3 | PX2 |
|---|---|---|---|---|---|---|---|

| Bit | Symbol | Function |
|---|---|---|
| IP1.7 | PX9 | External interrupt 9 priority level |
| IP1.6 | PX8 | External interrupt 8 priority level |
| IP1.5 | PX7 | External interrupt 7 priority level |
| IP1.4 | PX6 | External interrupt 6 priority level |
| IP1.3 | PX5 | External interrupt 5 priority level |
| IP1.2 | PX4 | External interrupt 4 priority level |
| IP1.1 | PX3 | External interrupt 3 priority level |
| IP1.0 | PX2 | External interrupt 2 priority level |

Interrupt priority is as follows:
0 = low priority
1 = high priority

**Interrupt polarity register IX1**
**IX1 (E9H)**

| IL9 | IL8 | IL7 | IL6 | IL5 | IL4 | IL3 | IL2 |
|---|---|---|---|---|---|---|---|

| Bit | Symbol | Function |
|---|---|---|
| IX1.7 | IL9 | External interrupt 9 polarity level |
| IX1.6 | IL8 | External interrupt 8 polarity level |
| IX1.5 | IL7 | External interrupt 7 polarity level |
| IX1.4 | IL6 | External interrupt 6 polarity level |
| IX1.3 | IL5 | External interrupt 5 polarity level |
| IX1.2 | IL4 | External interrupt 4 polarity level |
| IX1.1 | IL3 | External interrupt 3 polarity level |
| IX1.0 | IL2 | External interrupt 2 polarity level |

**Interrupt request flag register IRQ1**
**IRQ1 (C0H)**

| IQ9 | IQ8 | IQ7 | IQ6 | IQ5 | IQ4 | IQ3 | IQ2 |
|---|---|---|---|---|---|---|---|

| Bit | Symbol | Function |
|---|---|---|
| IRQ1.7 | IQ9 | External interrupt 9 request flag |
| IRQ1.6 | IQ8 | External interrupt 8 request flag |
| IRQ1.5 | IQ7 | External interrupt 7 request flag |
| IRQ1.4 | IQ6 | External interrupt 6 request flag |
| IRQ1.3 | IQ5 | External interrupt 5 request flag |
| IRQ1.2 | IQ4 | External interrupt 4 request flag |
| IRQ1.1 | IQ3 | External interrupt 3 request flag |
| IRQ1.0 | IQ2 | External interrupt 2 request flag |

### 1.6.2 Interrupt Vectors

| | Vector | Source |
|---|---|---|
| X0 | 0003H | External 0 |
| S0 | 0023H | UART SIO |
| X5 | 0053H | External 5 |
| T0 | 000BH | Timer 0 |
| X6 | 005BH | External 6 |
| X1 | 0013H | External 1 |
| X2 | 003BH | External 2 |
| X7 | 0063H | External 7 |
| T1 | 001BH | Timer 1 |
| X3 | 0043H | External 3 |
| X8 | 006BH | External 8 |
| X4 | 004BH | External 4 |
| X9 | 0073H | External 9 |

**Interrupt priority**
Each interrupt priority source can be set to either high or low priority. If both priorities are requested simultaneously, the controller will branch to the high priority vector.

A low priority interrupt can only be interrupted by a high priority interrupt. A high priority interrupt routine cannot be interrupted.

### 1.6.3 Related registers
The following registers are used in conjunction with the interrupt system:

| Register | Function |
|---|---|
| IX1 | Interrupt polarity register |
| IRQ1 | Interrupt enable register |
| IEN1 | Interrupt enable register (INT2-INT9) |
| IP0 | Interrupt priority register |
| IP1 | Interrupt priority register (INT2-INT9) |

## 1.7 Oscillator registers
The on-chip circuitry of the 80CL51 is a single-stage inverting amplifier biased by an internal feedback resistor (Figure 13). For operation as standard quartz oscillator, no external components are needed except at 32 KHz. When using external capacitors, ceramic resonators, coils and RC networks to drive the oscillator, five different configurations are supported (see Figure 14 and oscillator options).

In the Power-down mode the oscillator is stopped XTAL1 is pulled HIGH. The oscillator inverter is switched off to ensure no current will flow regardless of the voltage at XTAL1. To drive the device with an external clock source, apply the external clock signal to XTAL1, and leave XTAL2 to float, as shown in Figure 14(f). There are no requirements on the duty cycle of the external clock, since the input to the internal clocking circuitry is split sing a flip-flop.

The following options are provided for optimum on-chip oscillator performance. Please state option when ordering.

### 1.7.1 Oscillator options (see Figure 14)
The following options are provided for optimum on-chip oscillator performance. Please state option when ordering.

Osc.1: Figure 14(c): An option for 32 kHz clock applications with external trimmer for frequency adjustment. A 4.7 MQ bias resistor is needed for use in parallel with the crystal.

Osc. 2: Figure 14(e): An option for low-power, low-frequency operations using LC components.

Osc. 3: An option for medium frequency range applications.

Osc. 4: An option for high frequency range applications.

RC: Figure 14(g): An option for an RC oscillator.

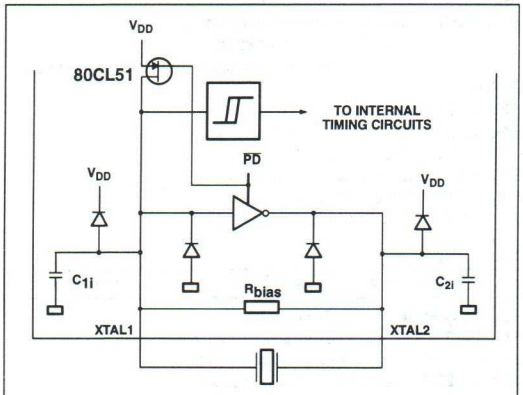



**Figure 13. Oscillator**

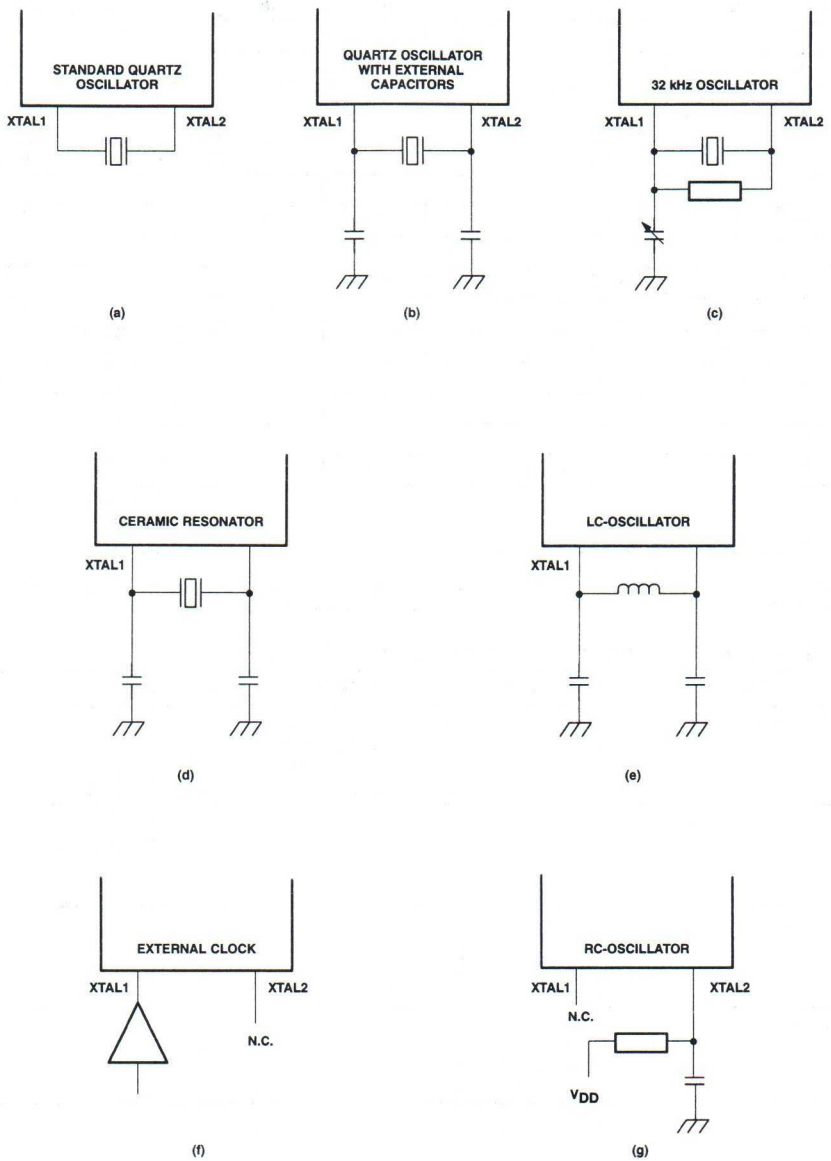



**Figure 14. Alternative Oscillator Configurations**

## OSCILLATOR TYPE SELECTION GUIDE

| RESONATOR | f(MHz) | OPTION | C1 EXT. (pF) | | C2 EXT. (pF) | | MAX. RESONATOR SERIES RESISTANCE |
|---|---|---|---|---|---|---|---|
| | | | MIN. | MAX. | MIN. | MAX. | |
| Quartz | 0.032 | OSC. 1 | 0 | 0 | 5 | 15 | 15 kΩ [1] |
| Quartz | 1.0 | OSC. 2 | 0 | 30 | 0 | 30 | 600 Ω |
| Quartz | 3.58 | OSC. 2 | 0 | 15 | 0 | 15 | 100 Ω |
| Quartz | 4.0 | OSC. 2 | 0 | 20 | 0 | 20 | 75 Ω |
| Quartz | 6.0 | OSC. 3 | 0 | 10 | 0 | 10 | 60 Ω |
| Quartz | 10.0 | OSC. 4 | 0 | 15 | 0 | 15 | 60 Ω |
| Quartz | 12.0 | OSC. 4 | 0 | 10 | 0 | 10 | 40 Ω |
| Quartz | 16.0 | OSC. 4 | 0 | 15 | 0 | 15 | 20 Ω |
| PXE | 0.455 | OSC. 2 | 40 | 50 | 40 | 50 | 10 Ω |
| PXE | 1.0 | OSC. 2 | 15 | 50 | 15 | 50 | 100 Ω |
| PXE | 3.58 | OSC. 2 | 0 | 40 | 0 | 40 | 10 Ω |
| PXE | 4.0 | OSC. 2 | 0 | 40 | 0 | 40 | 10 Ω |
| PXE | 6.0 | OSC. 2 | 0 | 20 | 0 | 20 | 5 Ω |
| PXE | 10.0 | OSC. 3 | 0 | 15 | 0 | 15 | 6 Ω |
| PXE | 12.0 | OSC. 4 | 10 | 40 | 10 | 40 | 6 Ω |
| LC | | OSC. 2 | 20 | 90 | 20 | 90 | 10 µH = 1 Ω<br>100 µH = 5 Ω<br>1 mH = 75 Ω |

**NOTES:**
1. 32 kHz quartz crystals with a series resistance higher than 15 kΩ will reduce the guaranteed supply voltage range to 2.5 -3.5V.
2. The equivalent circuit data of the internal oscillator compares with that of matched crystals.

## OSCILLATOR EQUIVALENT CIRCUIT PARAMETERS (SEE FIGURE 15)

| SYMBOL | PARAMETER | OPTION | CONDITION | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|---|---|
| $g_m$ | Transconductance | Osc.1 | T = +25 °C; $V_{DD}$ = 4.5V | - | 15 | - | µs |
| $g_m$ | | Osc.2 | T = +25 °C; $V_{DD}$ = 4.5V | 200 | 600 | 1000 | µs |
| $g_m$ | | Osc.3 | T = +25 °C; $V_{DD}$ = 4.5V | 400 | 1500 | 4000 | µs |
| $g_m$ | | Osc.4 | T = +25 °C; $V_{DD}$ = 4.5V | 1000 | 4000 | 10000 | µs |
| $C1_i$ | Input Capacitance | Osc.1 | | - | 3.0 | - | pF |
| $C1_i$ | | Osc. 2 | | - | 8.0 | - | pF |
| $C1_i$ | | Osc. 3 | | - | 8.0 | - | pF |
| $C1_i$ | | Osc. 4 | | - | 8.0 | - | pF |
| $C2_i$ | Output Capacitance | Osc.1 | | - | 23 | - | pF |
| $C2_i$ | | Osc. 2 | | - | 8.0 | - | pF |
| $C2_i$ | | Osc. 3 | | - | 8.0 | - | pF |
| $C2_i$ | | Osc. 4 | | - | 8.0 | - | pF |
| R2 | Output Capacitance | Osc.1 | | - | 3800 | - | kΩ |
| R2 | | Osc. 2 | | - | 65 | - | kΩ |
| R2 | | Osc. 3 | | - | 18 | - | kΩ |
| R2 | | Osc. 4 | | - | 5.0 | - | kΩ |

### 1.7.2 RC Oscillator (see Figure 16)

The externally adjustable RC-oscillator has a frequency range from 100 kHz to 500 kHz.

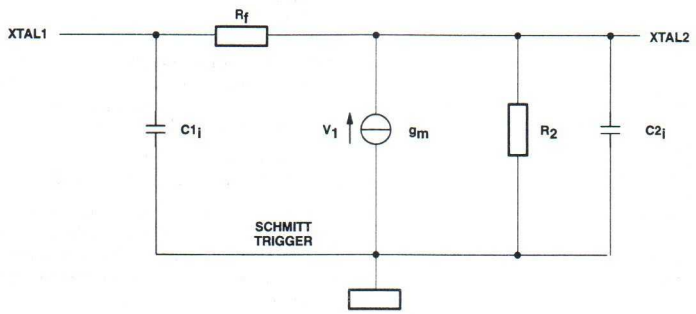



**Figure 15. Equivalent Circuit Diagram**

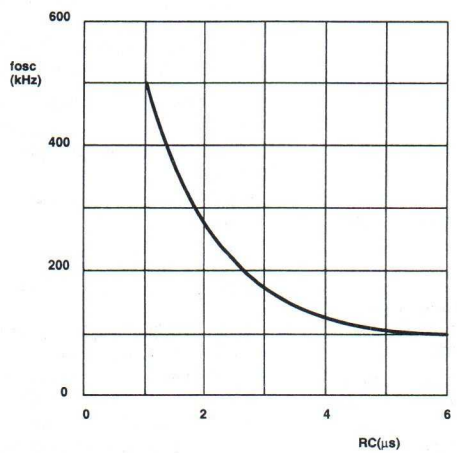



**Figure 16. Frequency as a Function of RC**

## 1.8 Reset Circuitry

To initialize the 80CL51, a reset is performed by either of two methods:

- via the RST pin
- via a power-on-reset

It leaves the internal registers as follows:

| REGISTER | CONTENT |
|---|---|
| ACC | 0000 0000 |
| B | 0000 0000 |
| DPL | 0000 0000 |
| DPH | 0000 0000 |
| IEN0 | 0000 0000 |
| IEN1 | 0000 0000 |
| IP0 | XX00 0000 |
| IP1 | 0000 0000 |
| IX1 | 0000 0000 |
| IRQ1 | 0000 0000 |
| PCH | 0000 0000 |
| PCL | 0000 0000 |
| PCON | 0XXX 0000 |
| PSW | 0000 0000 |
| P0-P3 | 1111 1111 |
| S0BUF | XXXX XXXX |
| S0CPN | 0000 0000 |
| SP | 0000 0111 |
| TCON | 0000 0000 |
| TH0, TH1 | 0000 0000 |
| TL0, TH1 | 0000 0000 |
| TL0, TL1 | 0000 0000 |
| TMOD | 0000 0000 |

The reset state of the port pins is mask- programmable and can therefore be defined by the user.

The standard reset value for port P0-P3 is 1111 1111.

The reset input to the 80CL51 is RST pin 9. A Schmitt trigger qualifies the input for noise rejection. The output of the Schmitt trigger is sampled by the reset circuitry every machine cycle.

A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods), while the oscillator is running. The CPU responds by generating an internal reset. Port pins adopt their reset state immediately after RST goes HIGH. During reset ALE and PSEN are held HIGH.

The external reset is asynchronous to the internal clock. The RST pin is sampled during State 5, Phase 2 of every machine cycle. After a HIGH is detected at the RST pin, an internal reset is repeated every cycle until RST goes LOW.

The internal RAM is not affected by reset. When $V_{DD}$ is turned on the RAM contents are indeterminate.

### 1.8.1 Power-on reset

The 80CL51 contains on-chip circuitry which switch the port pins to the customer defined logic level as soon as $V_{DD}$ exceeds 1.3V. As soon as the minimum supply voltage is reached, the oscillator will start up. However, to ensure that the oscillator is stable before the controller starts, the clock signals are gated away from the CPU for a further 1536 oscillator periods. During that time the CPU is held in a reset state.

A hysteresis of approximately 50 mV at a typical power-on switching level of 1.3 V will ensure correct operation.

The on-chip Power-on circuitry can be switched off via the mask option "OFF". This option reduces the power-down current to typically 800µA and can be chosen if external reset circuitry is used. For applications not requiring the internal reset option, "OFF" should be chosen.

An automatic reset can be obtained at power-on by connecting the RST pin to $V_{DD}$ via a 10µF capacitor. At power-on, the voltage on the RST pin is equal to $V_{DD}$ minus the capacitor voltage, and decreases from $V_{DD}$ as the capacitor discharges through the internal resistor $R_{RST}$ to ground. The larger the capacitor, the more slowly $V_{RST}$ decreases $V_{RST}$ must remain above the lower threshold of the Schmitt trigger long enough to effect a complete reset. The time required is the oscillator start-up time, plus 2 machine cycles.

## 1.9 P80CL31: ROMless version of P80CL51

The P80CL31 is a low voltage ROMless version of the P80CL51 microcontroller. The mask options on the P80CL31 are fixed as follows:

- Port options: all ports have option "1S", i.e., standard port, high after reset
- Oscillator option: OSC3
- Power-on Reset option: OFF

## 1.10 P80C51: 5V standard version

The P80C51 is a 5V version of the low voltage P80CL51 microcontroller. All functional features of the P80CL51 are maintained in the P80C51 with the exception of the mask options. The mask options on the P80C51 are as follows:

- Port options: all ports have option "1S", i.e., standard port, high after reset.
- Oscillator options: OSC3
- Power-on Reset option: OFF

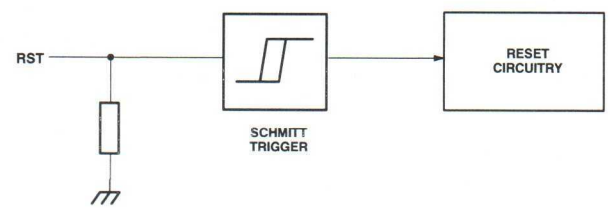



**Figure 17. Reset Configuration at RST Pin**

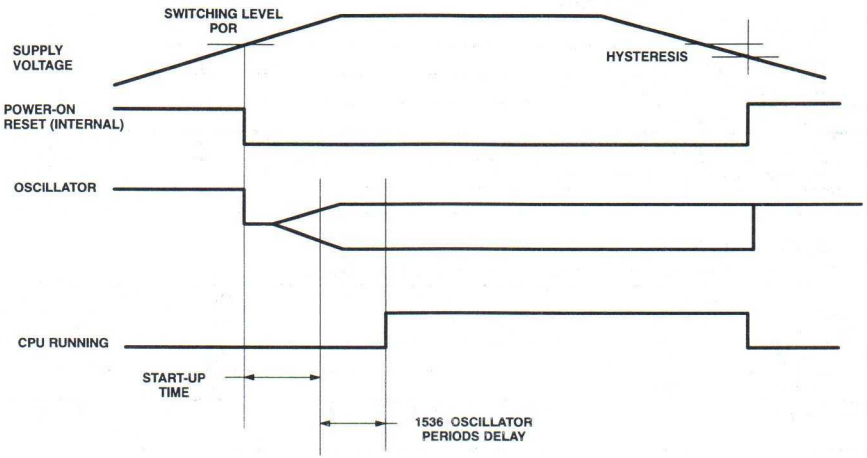



**Figure 18. Power-on Reset Switching Level**

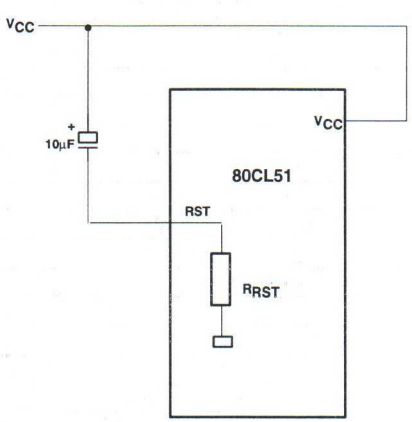



**Figure 19. Recommended Power-on Reset Circuitry**

## 2.0 RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

| SYMBOL | PARAMETER | MIN. | MAX. | UNIT |
|---|---|---|---|---|
| $V_{DD}$ | Supply voltage (pin 40) | -0.5 | + 6.5 | V |
| $V_I$ | All input voltages | -0.5 | $V_{DD}$+0.5 | V |
| $I_I$, $I_O$ | DC current into any input or output | - | 5 | mA |
| $P_{TOT}$ | Total power dissipation | - | 300 | mW |
| $T_{STG}$ | Storage temperature range | -65 | +150 | °C |
| $T_{AMB}$ | Operating ambient temperature range | -40 | +85 | °C |
| $T_J$ | Operating junction temperature | - | 125 | °C |

## 3.0 DC CHARACTERISTICS P80CL31/P80CL51

$V_{SS}$ = 0V; $T_{AMB}$ = -40 to +85°C; all voltages with respect to $V_{SS}$ unless otherwise specified.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|---|
| $V_{DD}$ | Supply voltage | $V_{SS}$ = 0V | 1.8 | - | 6.0 | V |
| $V_{DD}$ | RAM retention in power down mode | | 1.0 | – | – | V |
| **Supply current operating (Note 1, Note 4)** | | | | | | |
| $I_{DD}$ | OSC 1 option | $f_{clk}$ = 32 KHz; $V_{DD}$ = 1.8V $T_{AMB}$ - 25°C | - | - | 50 | µA |
| $I_{DD}$ | OSC 2 option | $f_{clk}$ = 3.58 MHz; $V_{DD}$ = 3V | - | - | 2.5 | mA |
| $I_{DD}$ | OSC 3 option | $f_{clk}$ = 16 MHz; $V_{DD}$ = 5V | - | - | 24 | mA |
| $I_{DD}$ | OSC 4 option | $f_{clk}$ = 16 MHz; $V_{DD}$ = 5V | - | - | 26 | mA |
| **Idle Mode (Note 2, Note 4)** | | | | | | |
| $I_{DD}$ | OSC 1 option | $f_{clk}$ = 32 KHz; $V_{DD}$ = 1.8V $T_{AMB}$ = 25°C | - | - | 25 | µA |
| $I_{DD}$ | OSC 2 option | $f_{clk}$ = 3.58 MHz; $V_{DD}$ = 3V | - | - | 1.0 | mA |
| $I_{DD}$ | OSC 3 option | $f_{clk}$ = 16 MHz; $V_{DD}$ = 5V | - | - | 10 | mA |
| $I_{DD}$ | OSC 4 option | $f_{clk}$ = 16 MHz; $V_{DD}$ = 5V | - | - | 12 | mA |
| $I_{PD}$ | Power down (Note 3, Note 4) | $V_{DD}$ = 1.8V, $T_{AMB}$ = 25°C | - | | 10 | µA |
| **Inputs** | | | | | | |
| $V_{IL}$ | Input voltage LOW | | $V_{SS}$ | - | 0.3$V_{DD}$ | V |
| $V_{IH}$ | Input voltage HIGH | | 0.7$V_{DD}$ | - | $V_{DD}$ | V |
| $I_{IL}$ | Input current logic 0 (Port 1, 2, 3) | $V_{DD}$ = 5V, $V_{IN}$ = 0.4V | - | - | 100 | µA |
| | | $V_{DD}$ = 2.5V, $V_{IN}$ = 0.4V | - | - | 50 | µA |
| $I_{TL}$ | Input current logic 1 to 0 transition (Port 1, 2, 3) | $V_{DD}$ = 5V, $V_{IN}$ = $V_{DD}$/2 | - | - | 1.0 | mA |
| | | $V_{DD}$ = 2.5V, $V_{IN}$ = $V_{DD}$/2 | - | - | 500 | µA |
| +/$I_{IL}$ | Input leakage current (Port 0, EA) | $V_{SS}$ < $V_I$ < $V_{DD}$ | - | - | 10 | µA |
| **Outputs** | | | | | | |
| $I_{OL}$ | Output sink current LOW | $V_{DD}$ = 5V, $V_{OL}$ = 0.4V | 1.6 | - | - | mA |
| | | $V_{DD}$ = 2.5V, $V_{OL}$ = 0.4V | 0.7 | - | - | mA |
| -$I_{OH}$ | Output source current HIGH (push-pull options only) | $V_{DD}$ = 5V; $V_{OH}$ = $V_{DD}$ -0.4V | 1.6 | - | - | mA |
| | | $V_{DD}$ = 2.5V; $V_{OH}$ = $V_{DD}$ -0.4V | 0.7 | - | - | mA |
| $R_{RST}$ | RST pull-down resistor | | 10 | – | 200 | kΩ |

**NOTES:**

1. The operating supply current is measured with all output pins disconnected; XTAL 1 driven with $t_r$ = $t_f$ = 10ns; $V_{IL}$ = $V_{SS}$; $V_{IH}$ = $V_{DD}$; XTAL 2 not connected; EA = RST = Port 0 = $V_{DD}$; all open drain outputs connected to $V_{SS}$.
2. The idle mode supply current is measured with all output pins disconnected; XTAL 1 driven with $t_r$ = $t_f$ = 10ns; $V_{IL}$ = $V_{SS}$. XTAL 2 not connected; EA = Port 0 = $V_{DD}$; RST = $V_{SS}$; all open drain outputs connected to $V_{SS}$.
3. The power-down current is measured with all output pins disconnected; XTAL 1 not connected; EA = Port 0 = $V_{DD}$; RST = $V_{SS}$; all open drain outputs connected to $V_{SS}$.
4. Circuits with Power-on Reset option "OFF" are tested at $V_{DD}$ minimum = 1.8V; with option "ON" (typically 1.3V) they are tested at $V_{DD}$ minimum = 2.3V. Please note, option "ON" is only available on P80CL51.

## 4.0 DC CHARACTERISTICS P80C51

$V_{SS}$ = 0V; $V_{DD}$ = 5V ± 10%; $f_{clk}$ = 3.5 to 16MHz; $T_{AMB}$ = -40 to +85°C; all voltages with respect to $V_{SS}$ unless otherwise specified.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|---|
| $V_{DD}$ | Supply voltage | $V_{SS}$ = 0V | 4.5 | - | 5.5 | V |
| **Supply Current** | | | | | | |
| $I_{DD}$ | Operating (Note 1) | $f_{CLK}$ = 16MHz, $V_{DD}$ = 5V | – | – | 24 | mA |
| $I_{DD}$ | Idle mode (Note 2) | $f_{CLK}$ = 16MHz, $V_{DD}$ = 5V | – | – | 10 | mA |
| $I_{PD}$ | Power down (Note 3) | $V_{DD}$ = 5V | – | – | 50 | µA |
| **Inputs** | | | | | | |
| $V_{IL}$ | Input voltage LOW | | $V_{SS}$ | – | $0.3V_{DD}$ | V |
| $V_{IH}$ | Input voltage HIGH | | $0.7V_{DD}$ | – | $V_{DD}$ | V |
| $I_{IL}$ | Input current logic 0 (Port 1, 2, 3) | $V_{IN}$ = 0.4V | – | – | 100 | µA |
| $I_{IL}$ | Input current logic 1 to 0 transition (Port 1, 2, 3) | $V_{IN} = V_{DD}/2$ | – | – | 1.0 | mA |
| $I_{IL}$ | Input leakage current (Port 0, EA) | $V_{SS} < V_I < V_{DD}$ | – | – | 10 | µA |
| **Outputs** | | | | | | |
| $I_{OL}$ | Output sink current LOW | $V_{OL}$ = 0.4V | 1.6 | – | – | mA |
| $I_{OH}$ | Output source current HIGH (push-pull options only) | $V_{OH} = V_{DD} - 0.4V$ | 1.6 | – | – | mA |
| $R_{RST}$ | RST pull-down resistor | | 10 | – | 200 | kΩ |

**NOTES:**

1. The operating supply current is measured with all output pins disconnected; XTAL 1 driven with $t_R = t_F$ = 10ns; $V_{IL} = V_{SS}$; $V_{IH} = V_{DD}$; XTAL 2 not connected; EA = RST = Port 0 = $V_{DD}$; all open drain outputs connected to $V_{SS}$.
2. The idle mode supply current is measured with all output pins disconnected; XTAL 1 driven with $t_R = t_F$ = 10ns; $V_{IL} = V_{SS}$. XTAL 2 not connected; EA = Port 0 = $V_{DD}$; RST = $V_{SS}$; all open drain outputs connected to $V_{SS}$.
3. The power-down current is measured with all output pins disconnected; XTAL 1 not connected; EA = Port 0 = $V_{DD}$; RST = $V_{SS}$; all open drain outputs connected to $V_{SS}$.
4. Please note, option "ON" is only available on P80CL51.

## 5.0 AC CHARACTERISTICS

$V_{DD}$ = 5 V; $V_{SS}$ = 0V; $T_{amb}$ = -40 to +85°C; $C_L$ = 50 pF for Port 0, ALE and PSEN; $C_L$ = 40pF for all other outputs, unless otherwise specified.

### PROGRAM MEMORY (See Figure 20)

| SYMBOL | PARAMETER | | | VARIABLE CLOCK | |
|---|---|---|---|---|---|
| | | MIN. | TYP. | MAX. | UNIT |
| $t_{LL}$ | ALE pulse duration | $2T_{CK}$-40 | - | - | ns |
| $t_{AL}$ | Address set-up time to ALE | $T_{CK}$-40 | - | - | ns |
| $t_{LA}$ | Address hold time to ALE | $T_{CK}$-35 | - | - | ns |
| $t_{LC}$ | Time from ALE to control pulse PSEN | $T_{CK}$-25 | - | - | ns |
| $t_{LIV}$ | Time from ALE to valid instruction input | - | - | $4T_{CK}$-100 | ns |
| $t_{CC}$ | Control pulse duration PSEN | $3T_{CK}$-35 | - | - | ns |
| $t_{CIV}$ | Time from PSEN to valid instruction input | - | - | $3T_{CK}$-125 | ns |
| $t_{CI}$ | Input instruction hold time after PSEN | 0 | - | - | ns |
| $t_{CIF}$ | Input instruction float delay after PSEN | - | - | $T_{CK}$-20 | ns |
| $t_{AIV}$ | Address to valid instruction input | - | - | $5T_{CK}$-115 | ns |
| $t_{AFC}$ | Address float time to PSEN | 0 | - | - | ns |

### EXTERNAL DATA MEMORY (See Figures 21 and 22)

| SYMBOL | PARAMETER | | | VARIABLE CLOCK | |
|---|---|---|---|---|---|
| | | MIN. | TYP. | MAX. | UNIT |
| $t_{RR}$ | RD pulse duration | $6T_{CK}$-100 | - | - | ns |
| $t_{WW}$ | WR pulse duration | $6T_{CK}$-100 | - | - | ns |
| $t_{LA}$ | Address hold time after ALE | $T_{CK}$-35 | - | - | ns |
| $t_{RD}$ | RD to valid data input | $T_{CK}$-35 | - | $5T_{CK}$-165 | ns |
| $t_{DFR}$ | Data float delay after RD | - | - | $2T_{CK}$-70 | ns |
| $t_{LD}$ | Time from ALE to valid data input | - | - | $8T_{CK}$-150 | ns |
| $t_{AD}$ | Address to valid data input | - | - | $9T_{CK}$-165 | ns |
| $t_{LW}$ | Time from ALE to RD and WR | $3T_{CK}$-50 | - | $3T_{CK}$+50 | ns |
| $t_{AW}$ | Time from address to RD and WR | $4T_{CK}$-130 | - | - | ns |
| $t_{WHLH}$ | Time from RD or WR HIGH to ALE HIGH | $T_{CK}$-40 | - | $T_{CK}$-40 | ns |
| $t_{DWX}$ | Data valid to WR transition | $T_{CK}$-60 | - | - | ns |
| $t_{DW}$ | Data set-up time before WR | $T_{CK}$-150 | - | - | ns |
| $t_{WD}$ | Data hold time after WR | $T_{CK}$-50 | - | - | ns |
| $t_{WAFR}$ | Address float delay after RD (Note 1) | - | - | 12 | ns |

**NOTE:**

1. Interfacing the 80CL51 or P80C51 to devices with float times up to 75ns is permitted. This limited bus connection will not cause damage to Port 0 drivers.

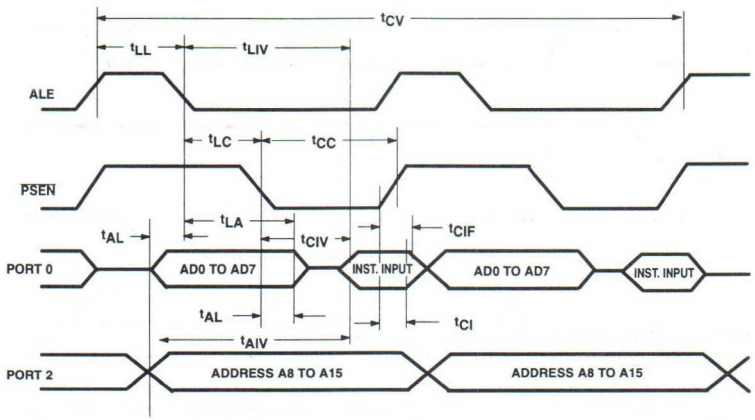



Figure 20. Read from Program Memory

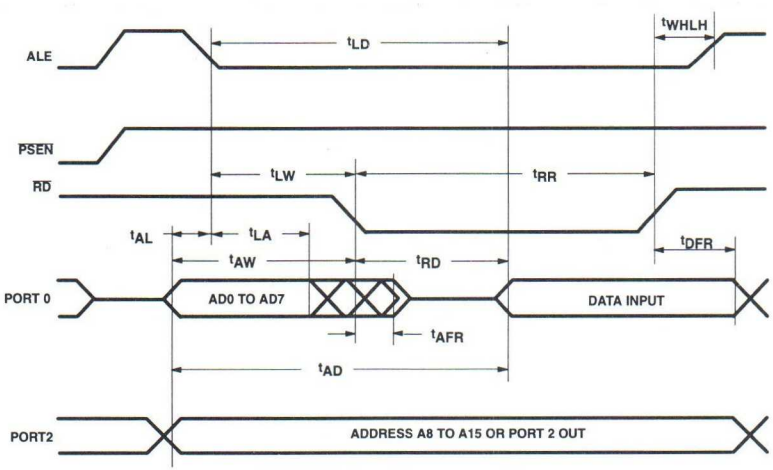



Figure 21. Read from Data Memory

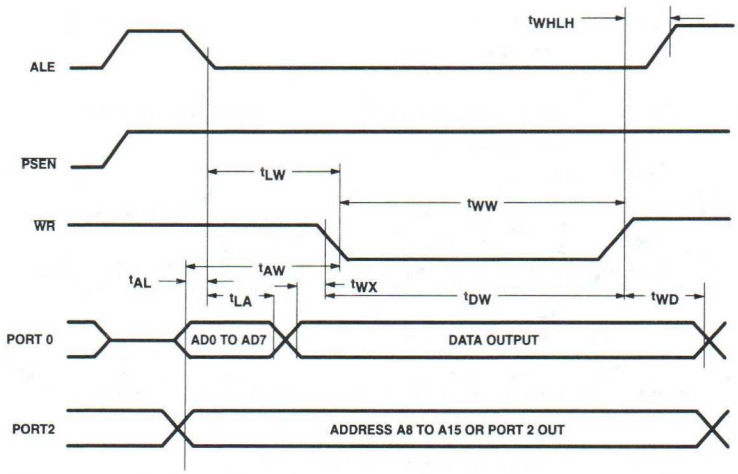



**Figure 22. Write to Data Memory**

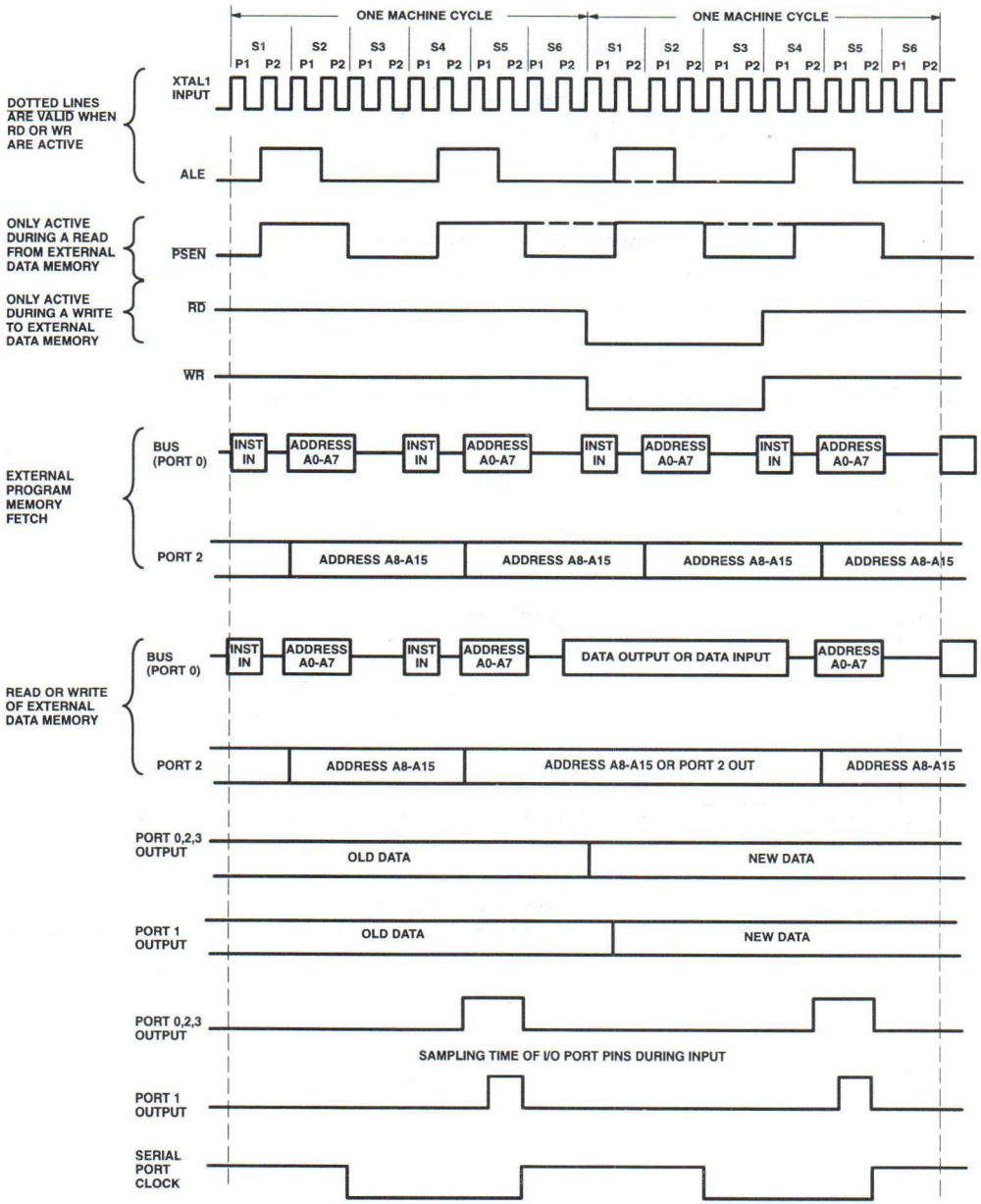



**Figure 23. Instruction Cycle Timing**

## 6.0 CHARACTERISTICS CURVES

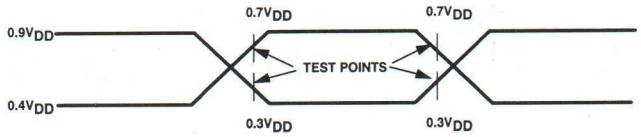



Figure 24. AC Testing Input Waveform

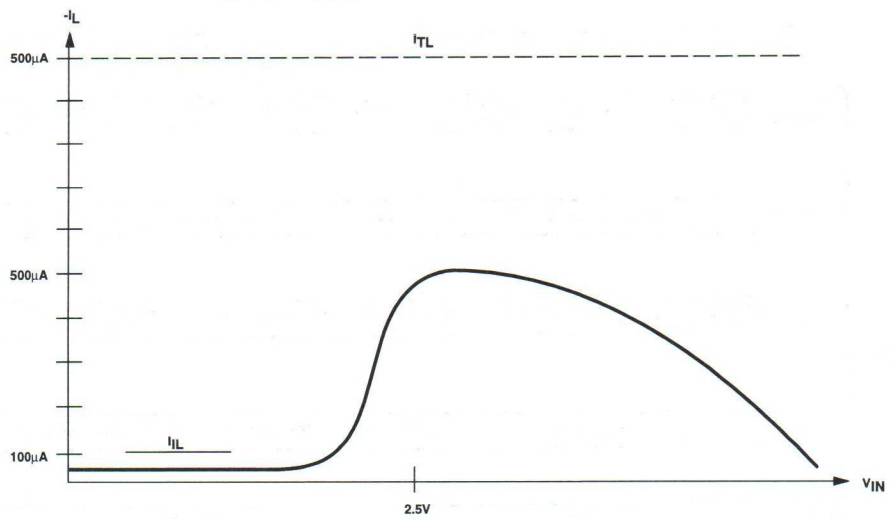



Figure 25. Input Current at $V_{DD}$ = 5V

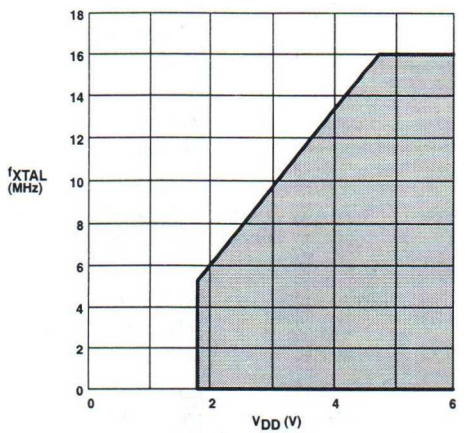



Figure 26. P80CL51/31 Frequency Operating Range

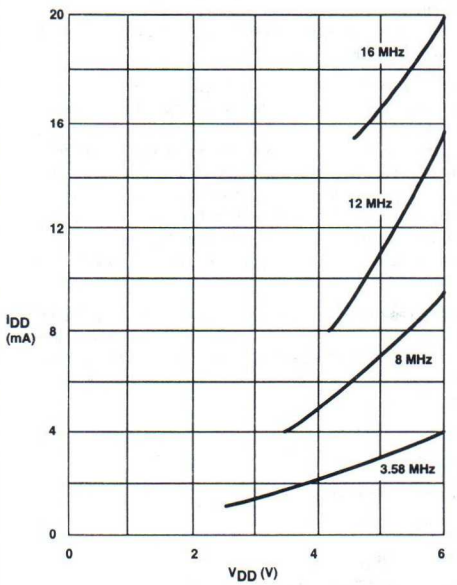



Figure 27. P80CL51/31 Typical Operating Current vs Frequency and $V_{DD}$, $T_{amb}$ = 25°C.

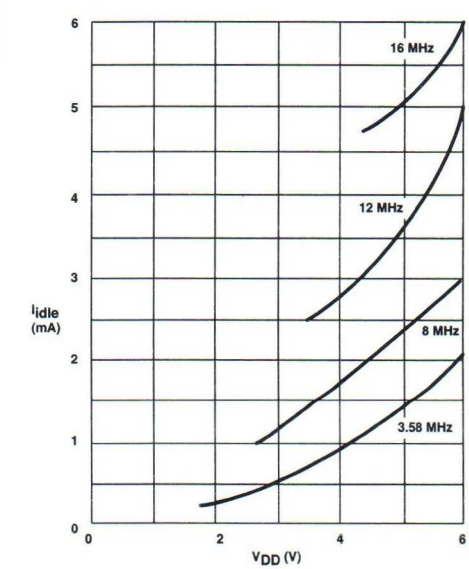



Figure 28. P80CL51/31 Typical Idle Current vs Frequency and $V_{DD}$, $T_{amb}$ = 25°C.

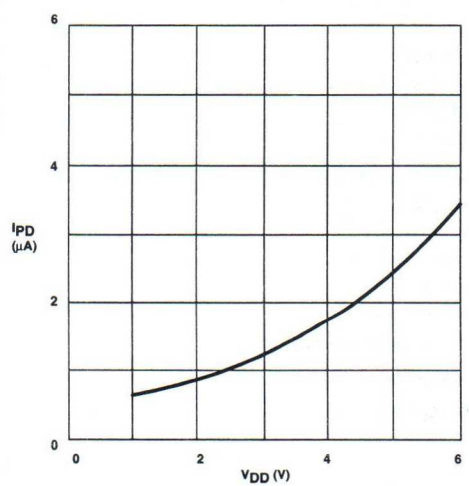



Figure 29. P80CL51/31 Typical Power-Down Current vs Frequency and $V_{DD}$, $T_{amb}$ = 25°C.

# CMOS single-chip 3.0V 8-bit microcontrollers 87L51FA/87L51FB

## DESCRIPTION

The 87L51FA and 87L51FB Single-Chip 3.0V 8-Bit Microcontrollers are manufactured in an advanced CMOS process and are derivatives of the 80C51 microcontroller family. The 87L51FA/B has the same instruction set as the 80C51.

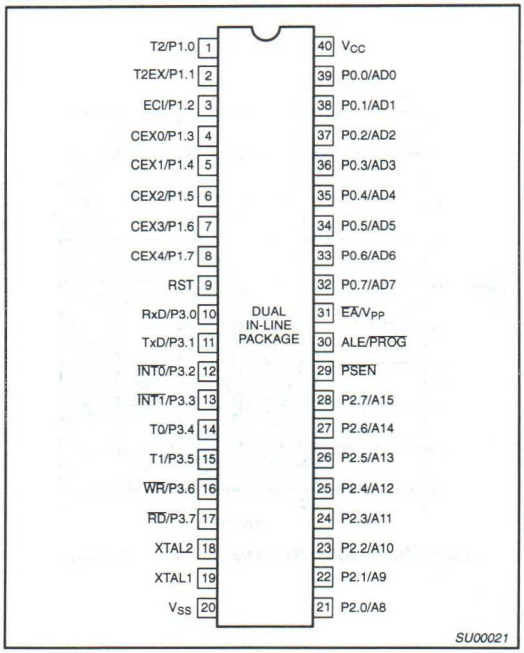
This device provides architectural enhancements that make it applicable in a variety of applications for general control systems. The 87L51FA contains 8k × 8 memory and the 87L51FB contains 16K × 8 memory, a volatile 256 × 8 read/write data memory, four 8-bit I/O ports, three 16-bit timer/event counters, a Programmable Counter Array (PCA), a multi-source, two-priority-level, nested interrupt structure, an enhanced UART and on-chip oscillator and timing circuits. For systems that require extra capability, the 87L51FA/B can be expanded using standard 3.3V TTL compatible memories and logic.

Its added features make it an even more powerful microcontroller for applications that require pulse width modulation, high-speed I/O and up/down counting capabilities such as motor control. It also has a more versatile serial channel that facilitates multiprocessor communications.

## FEATURES

- 80C51 central processing unit
- 3.0 to 4.5V $V_{CC}$ range
- 8k × 8 EPROM (87L51FA)
  16k × 8 EPROM (87L51FB)
  - Expandable externally to 64k bytes
  - Quick Pulse programming algorithm
  - Two level program security system
- 256 × 8 RAM, expandable externally to 64k bytes
- Three 16-bit timer/counters
  - T2 is an up/down counter
- Programmable Counter Array (PCA)
  - High speed output
  - Capture/compare
  - Pulse Width Modulator
  - Watchdog Timer
- Four 8-bit I/O ports
- Full-duplex enhanced UART
  - Framing error detection
  - Automatic address recognition
- Power control modes
  - Idle mode
  - Power-down mode
- Once (On Circuit Emulation) Mode
- Five package styles
- OTP package available

## PIN CONFIGURATIONS

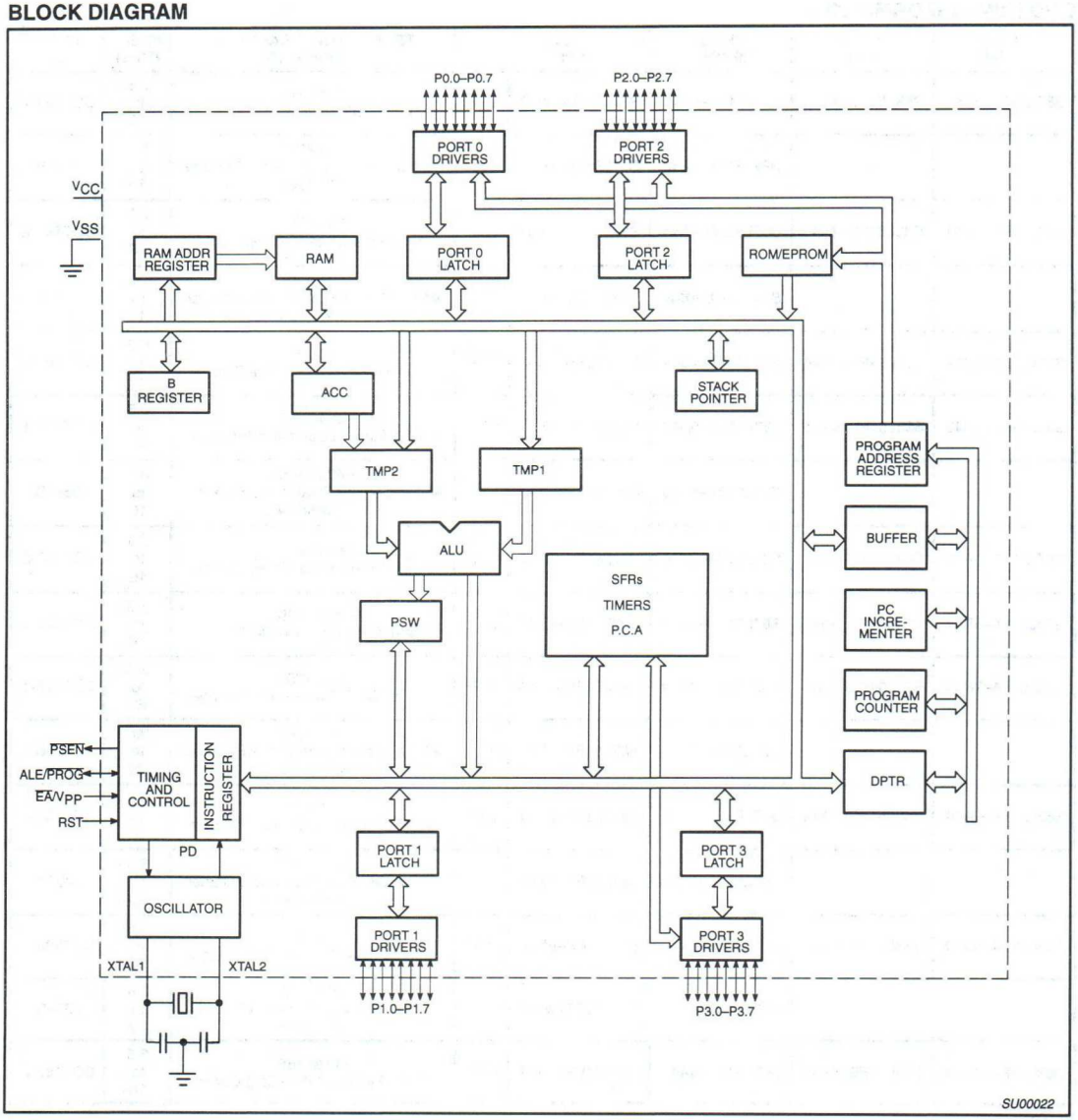


## ORDERING INFORMATION

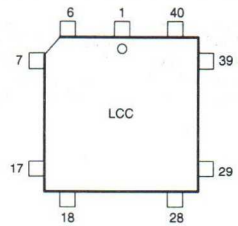
| 8k × 8 ROM[1] | 6k × 8 ROM[1] | 8k × 8 EPROM[2] | 16k × 8 EPROM[2] | | TEMPERATURE RANGE °C AND PACKAGE | FREQ. (MHz) | DWG. # |
|---|---|---|---|---|---|---|---|
| S83L51FA–4N40 | S83L51FB–4N40 | S87L51FA–4N40 | S87L51FB–4N40 | OTP | 0 to +70, 40-Pin Plastic Dual In-line Package | 3.5 to 16 | SOT129-1 |
| | | S87L51FA–4F40 | S87L51FB–4F40 | UV | 0 to +70, 40-Pin Ceramic Dual In-line Package w/Window | 3.5 to 16 | 0590B |
| S83L51FA–4A44 | S83L51FB–4A44 | S87L51FA–4A44 | S87L51FB–4A44 | OTP | 0 to +70, 44-Pin Plastic Leaded Chip Carrier | 3.5 to 16 | SOT187-2 |
| | | S87L51FA–4K44 | S87L51FB–4K44 | UV | 0 to +70, 44-Pin Ceramic Leaded Chip Carrier w/Window | 3.5 to 16 | 1472A |
| S83L51FA–4B44 | S83L51FB–4B44 | S87L51FA–4B44 | S87L51FB–4B44 | OTP | 0 to +70, 44-Pin Plastic Quad Flat Pack | 3.5 to 16 | SOT307-2 |
| S83L51FA–5N40 | S83L51FB–5N40 | S87L51FA–5N40 | S87L51FB–5N40 | OTP | –40 to +85, 40-Pin Plastic Dual In-line Package | 3.5 to 16 | SOT129-1 |
| | | S87L51FA–5F40 | S87L51FB–5F40 | UV | –40 to +85, 40-Pin Ceramic Dual In-line Package w/Window | 3.5 to 16 | 0590B |
| S87L51FA–5A44 | S87L51FB–5A44 | S87L51FA–5A44 | S87L51FB–5A44 | OTP | –40 to +85, 44-Pin Plastic Leaded Chip Carrier | 3.5 to 16 | SOT187-2 |
| S83L51FA–5B44 | S83L51FB–5B44 | S87L51FA–5B44 | S87L51FB–5B44 | OTP | –40 to +85, 44-Pin Plastic Quad Flat Pack | 3.5 to 16 | SOT307-2 |
| S83L51FA–7N40 | S83L51FB–7N40 | S87L51FA–7N40 | S87L51FB–7N40 | OTP | 0 to +70, 40-Pin Plastic Dual In-line Package | 3.5 to 20 | SOT129-1 |
| | | S87L51FA–7F40 | S87L51FB–7F40 | UV | 0 to +70, 40-Pin Ceramic Dual In-line Package w/Window | 3.5 to 20 | 0590B |
| S83L51FA–7A44 | S83L51FB–7A44 | S87L51FA–7A44 | S87L51FB–7A44 | OTP | 0 to +70, 44-Pin Plastic Leaded Chip Carrier | 3.5 to 20 | SOT187-2 |
| | | S87L51FA–7K44 | S87L51FB–7K44 | UV | 0 to +70, 44-Pin Ceramic Leaded Chip Carrier w/Window | 3.5 to 20 | 1472A |
| S83L51FA–8N40 | S83L51FB–8N40 | S87L51FA–8N40 | S87L51FB–8N40 | OTP | –40 to +85, 40-Pin Plastic Dual In-line Package | 3.5 to 20 | SOT129-1 |
| | | S87L51FA–8F40 | S87L51FB–8F40 | UV | –40 to +85, 40-Pin Ceramic Dual In-line Package w/Window | 3.5 to 20 | 0590B |
| S83L51FA–8A44 | S83L51FB–8A44 | S87L51FA–8A44 | S87L51FB–8A44 | OTP | –40 to +85, 44-Pin Plastic Leaded Chip Carrier | 3.5 to 20 | SOT187-2 |

**NOTES:**
1. Contact Philips for information on low voltage Mask-ROM versions.
   The 83C51FA and 83C51FB are specified for 2.7V–5.5V operation @ 16MHz.
2. OTP = One Time Programmable EPROM. UV = Erasable EPROM.

## BLOCK DIAGRAM



## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

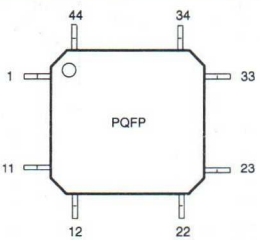



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | NC* | 16 | P3.4/T0 | 31 | P2.7/A15 |
| 2 | P1.0/T2 | 17 | P3.5/T1 | 32 | $\overline{PSEN}$ |
| 3 | P1.1/T2EX | 18 | P3.6/$\overline{WR}$ | 33 | ALE/$\overline{PROG}$ |
| 4 | P1.2/ECI | 19 | P3.7/$\overline{RD}$ | 34 | NC* |
| 5 | P1.3/CEX0 | 20 | XTAL2 | 35 | $\overline{EA}$/$V_{PP}$ |
| 6 | P1.4/CEX1 | 21 | XTAL1 | 36 | P0.7/AD7 |
| 7 | P1.5/CEX2 | 22 | $V_{SS}$ | 37 | P0.6/AD6 |
| 8 | P1.6/CEX3 | 23 | NC* | 38 | P0.5/AD5 |
| 9 | P1.7/CEX4 | 24 | P2.0/A8 | 39 | P0.4/AD4 |
| 10 | RST | 25 | P2.1/A9 | 40 | P0.3/AD3 |
| 11 | P3.0/RxD | 26 | P2.2/A10 | 41 | P0.2/AD2 |
| 12 | NC* | 27 | P2.3/A11 | 42 | P0.1/AD1 |
| 13 | P3.1/TxD | 28 | P2.4/A12 | 43 | P0.0/AD0 |
| 14 | P3.2/$\overline{INT0}$ | 29 | P2.5/A13 | 44 | $V_{CC}$ |
| 15 | P3.3/$\overline{INT1}$ | 30 | P2.6/A14 | | |

* DO NOT CONNECT

SU00023

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS




| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | P1.5/CEX2 | 16 | $V_{SS}$ | 31 | P0.6/AD6 |
| 2 | P1.6/CEX3 | 17 | NC* | 32 | P0.5/AD5 |
| 3 | P1.7/CEX4 | 18 | P2.0/A8 | 33 | P0.4/AD4 |
| 4 | RST | 19 | P2.1/A9 | 34 | P0.3/AD3 |
| 5 | P3.0/RxD | 20 | P2.2/A10 | 35 | P0.2/AD2 |
| 6 | NC* | 21 | P2.3/A11 | 36 | P0.1/AD1 |
| 7 | P3.1/TxD | 22 | P2.4/A12 | 37 | P0.0/AD0 |
| 8 | P3.2/$\overline{INT0}$ | 23 | P2.5/A13 | 38 | $V_{CC}$ |
| 9 | P3.3/$\overline{INT1}$ | 24 | P2.6/A14 | 39 | NC* |
| 10 | P3.4/T0 | 25 | P2.7/A15 | 40 | P1.0/T2 |
| 11 | P3.5/T1 | 26 | $\overline{PSEN}$ | 41 | P1.1/T2EX |
| 12 | P3.6/$\overline{WR}$ | 27 | ALE/$\overline{PROG}$ | 42 | P1.2/ECI |
| 13 | P3.7/$\overline{RD}$ | 28 | NC* | 43 | P1.3/CEX0 |
| 14 | XTAL2 | 29 | $\overline{EA}$/$V_{PP}$ | 44 | P1.4/CEX1 |
| 15 | XTAL1 | 30 | P0.7/AD7 | | |

* DO NOT CONNECT

SU00024

## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER DIP | LCC | QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification and receives code bytes during EPROM programming. External pull-ups are required during program verification. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also receives the low-order address byte during program memory verification. Alternate functions include: |
| | 1 | 2 | 40 | I | **T2 (P1.0):** Timer/Counter 2 external count input/Clockout |
| | 2 | 3 | 41 | I | **T2EX (P1.1):** Timer/Counter 2 Reload/Capture/Direction Control |
| | 3 | 4 | 42 | I | **ECI (P1.2):** External Clock Input to the PCA |
| | 4 | 5 | 43 | I/O | **CEX0 (P1.3):** Capture/Compare External I/O for PCA module 0 |
| | 5 | 6 | 44 | I/O | **CEX1 (P1.4):** Capture/Compare External I/O for PCA module 1 |
| | 6 | 7 | 1 | I/O | **CEX2 (P1.5):** Capture/Compare External I/O for PCA module 2 |
| | 7 | 8 | 2 | I/O | **CEX3 (P1.6):** Capture/Compare External I/O for PCA module 3 |
| | 8 | 9 | 3 | I/O | **CEX4 (P1.7):** Capture/Compare External I/O for PCA module 4 |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. Some Port 2 pins receive the high order address bits during EPROM programming and verification. |

## PIN DESCRIPTIONS (Continued)

| MNEMONIC | PIN NUMBER DIP | PIN NUMBER LCC | PIN NUMBER QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{CC}$. |
| ALE/$\overline{PROG}$ | 30 | 33 | 27 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the 87L51FA/FB is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}/V_{PP}$ | 31 | 35 | 29 | I | **External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H and 1FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 1FFFH. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. If security bit 1 is programmed, $\overline{EA}$ will be internally latched on Reset. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTE:**
To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher than $V_{CC}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

### TIMER 2

This is a 16-bit up or down counter, which can be operated as either a timer or event counter. It can be operated in one of three different modes (autoreload, capture or as the baud rate generator for the UART).

In the autoreload mode the Timer can be set to count up or down by setting or clearing the bit DCEN in the T2CON Special Function Register. The SFR's RCAP2H and RCAP2L are used to reload the Timer upon overflow or a 1-to-0 transition on the T2EX input (P1.1).

In the Capture mode Timer 2 can either set TF2 and generate an interrupt or capture its value. To capture Timer 2 in response to a 1-to-0 transition on the T2EX input, the EXEN2 bit in the T2CON must be set. Timer 2 is then captured in SFR's RCAP2H and RCAP2L.

As the baud rate generator, Timer 2 is selected by setting TCLK and/or RCLK in T2CON. As the baud rate generator Timer 2 is incremented at $^1/_2$ the oscillator frequency.

### ENHANCED UART

The 87L51FA/FB UART has all of the capabilities of the standard 80C51 UART plus Framing Error Detection and Automatic Address Recognition. As in the 80C51, all four modes of operation are supported as well as the 9th bit in modes 2 and 3 that can be used to facilitate multiprocessor communication.

The Framing Error Detection allows the UART to look for missing stop bits. If a Stop bit is missing, the FE bit in the SCON SFR is set. The FE bit can be checked after each transmission to detect communication errors. The FE bit can only be cleared by software and is not affected by a valid stop bit.

Automatic Address Recognition is used to reduce the CPU service time for the serial port. The CPU only needs to service the UART when it is addressed and, with this done by the on-chip circuitry, the need for software overhead is greatly reduced. This mode works similar to the 9-bit communication mode, except that it uses only 8 bits and the Stop bit is used to cause the RI bit to be set. There are two SFRs associated with this mode. They are SADDR, which holds the slave address and SADEN, which contains a mask that allows selective masking of the slave address so that broadcast addresses can be used.

## PROGRAMMABLE COUNTER ARRAY

The PCA is a sophisticated free-running 16 bit Timer/Counter that drives 5 modules that can be individually configured as Capture inputs, software timers, high speed outputs, or pulse width modulated outputs. In addition, module 4 can be configured as a software controlled watchdog timer.

The Timer portion of the PCA can be configured to run in one of four different modes. The modes are: $^1/_2$ the oscillator frequency, $^1/_4$ the oscillator frequency, Timer 0 overflows, or from the ECI input.

For the Capture/Compare mode each of the modules has a pair of registers associated with it called CCAPnH and CCAPnL (where n = 0, 1, 2, 3, 4 depending on the module). Both positive and negative transitions can be captured. This means that the PCA has the flexibility to measure phase differences, duty cycles, pulse widths and a wide variety of other digital pulse characteristics.

In the 16-bit software timer mode each of the modules can generate an interrupt upon a compare.

For applications that require accurate pulse widths and edges the PCA modules can be used as High Speed Outputs (HSO). The PCA toggles the appropriate CEXn pin when there is a match between the PCA timer and the modules compare registers.

The pulse width modulator mode for the PCA allows the conversion of digital information into analog signals. Each of the 5 modules can be used in this mode. The frequency of the PWM depends on the clock source for the PCA. The 8-bit PWM output is generated by comparing the low byte of the PCA (CL) with the module's CCAPnL SFR. When CL < CCAPnL, the output is high. When CL > CCAPnL, the output is low.

## POWER OFF FLAG

The Power Off Flag (POF) is set by on-chip circuitry when the $V_{CC}$ level on the 87L51FA/FB rises from 0 to 3.3V. The POF bit can be set or cleared by software allowing a user to determine if the reset is the result of a power-on or a warm start after powerdown. The $V_{CC}$ level must remain above 2.0V for the POF to remain unaffected by the $V_{CC}$ level.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

### Reset

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up.

### Idle Mode

In the idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

### Power-Down Mode

To save even more power, a Power Down mode can be invoked by software. In this mode, the oscillator is stopped and the instruction that invoked Power Down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the Power Down mode is terminated.

On the 87L51FA/FB either a hardware reset or external interrupt can use an exit from Power Down. Reset redefines all the SFRs but does not change the on-chip RAM. An external interrupt allows both the SFRs and the on-chip RAM to retain their values.

To properly terminate Power Down the reset or external interrupt should not be executed before $V_{CC}$ is restored to its normal operating level and must be held active long enough for the oscillator to restart and stabilize (normally less than 10ms).

With an external interrupt, INT0 and INT1 must be enabled and configured as level-sensitive. Holding the pin low restarts the oscillator but bringing the pin back high completes the exit. Once the interrupt is serviced, the next instruction to be executed after RETI will be the one following the instruction that put the device into Power Down.

### Design Consideration

- When the idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal rest algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

### ONCE™ Mode

The ONCE ("On-Circuit Emulation") Mode facilitates testing and debugging of systems using the 87L51FA/FB without the 87L51FA/FB having to be removed from the circuit. The ONCE Mode is invoked by:

1. Pull ALE low while the device is in reset and $\overline{PSEN}$ is high;
2. Hold ALE low as RST is deactivated.

While the device is in ONCE Mode, the Port 0 pins go into a float state, and the other port pins and ALE and $\overline{PSEN}$ are weakly pulled high. The oscillator circuit remains active. While the 87L51FA/FB is in this mode, an emulator or test CPU can be used to drive the circuit. Normal operation is restored when a normal reset is applied.

# CMOS single-chip 8-bit microcontrollers — 80C32/87C52

## DESCRIPTION

The Philips 80C32/87C52 is a high-performance microcontroller fabricated with Philips high-density CMOS technology. The Philips CMOS technology combines the high speed and density characteristics of HMOS with the low power attributes of CMOS. Philips epitaxial substrate minimizes latch-up sensitivity.

The 87C52 contains an 8k × 8 EPROM and the 80C32 is ROMless. Both contain a 256 × 8 RAM, 32 I/O lines, three 16-bit counter/timers, a six-source, two-priority level nested interrupt structure, a serial I/O port for either multi-processor communications, I/O expansion or full duplex UART, and on-chip oscillator and clock circuits.

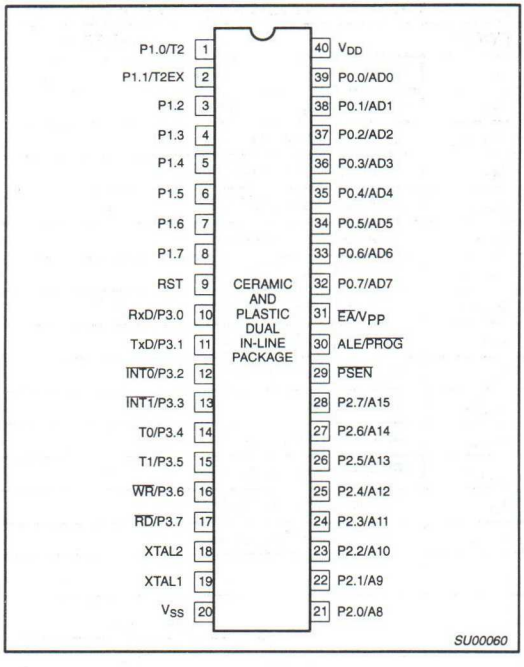
In addition, the 80C32/87C52 has two software selectable modes of power reduction—idle mode and power-down mode. The idle mode freezes the CPU while allowing the RAM, timers, serial port, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

See 80C52/80C54/80C58 datasheet for ROM device specifications.

## FEATURES

- 80C51 based architecture
- 8032 compatible
  - 8k × 8 EPROM (87C52)
  - ROMless (80C32)
  - 256 × 8 RAM
  - Three 16-bit counter/timers
  - Full duplex serial channel
  - Boolean processor
- Memory addressing capability
  - 64k ROM and 64k RAM
- Power control modes:
  - Idle mode
  - Power-down mode
- CMOS and TTL compatible
- Three speed ranges:
  - 3.5 to 16MHz
  - 3.5 to 24MHz
  - 3.5 to 33MHz
- Five package styles
- Extended temperature ranges
- OTP package available

## PIN CONFIGURATIONS



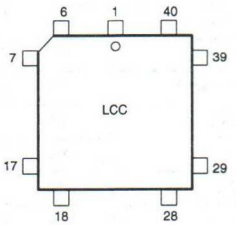
## ORDERING INFORMATION

| ROMless | EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQ MHz | DRAWING NUMBER |
|---|---|---|---|---|---|
| P80C32EBP N | P87C52EBP N | OTP | 0 to +70, Plastic Dual In-line Package | 16 | SOT129-1 |
| P80C32EBA A | P87C52EBA A | OTP | 0 to +70, Plastic Leaded Chip Carrier | 16 | SOT187-2 |
| | P87C52EBF FA | UV | 0 to +70, Ceramic Dual In-line Package | 16 | 0590B |
| | P87C52EBL KA | UV | 0 to +70, Ceramic Leaded Chip Carrier | 16 | 1472A |
| P80C32EBB B | P87C52EBB B | OTP | 0 to +70, Plastic Quad Flat Pack | 16 | SOT307-2 |
| P80C32EFP N | P87C52EFP N | OTP | –40 to +85, Plastic Dual In-line Package | 16 | SOT129-1 |
| P80C32EFA A | P87C52EFA A | OTP | –40 to +85, Plastic Leaded Chip Carrier | 16 | SOT187-2 |
| | P87C52EFF FA | UV | –40 to +85, Ceramic Dual In-line Package | 16 | 0590B |
| P80C32EFB B | P87C52EFB B | OTP | –40 to +85, Plastic Quad Flat Pack | 16 | SOT307-2 |
| P80C32IBP N | P87C52IBP N | OTP | 0 to +70, Plastic Dual In-line Package | 24 | SOT129-1 |
| P80C32IBA A | P87C52IBA A | OTP | 0 to +70, Plastic Leaded Chip Carrier | 24 | SOT187-2 |
| P80C32IBB B | | | 0 to +70, Plastic Quad Flat Pack | 24 | SOT307-2 |
| | P87C52IBF FA | UV | 0 to +70, Ceramic Dual In-line Package | 24 | 0590B |
| | P87C52IBL KA | UV | 0 to +70, Ceramic Leaded Chip Carrier | 24 | 1472A |
| P80C32IFP N | P87C52IFP N | OTP | –40 to +85, Plastic Dual In-line Package | 24 | SOT129-1 |
| P80C32IFA A | P87C52IFA A | OTP | –40 to +85, Plastic Leaded Chip Carrier | 24 | SOT187-2 |
| P80C32IFB B | | | –40 to +85, Plastic Quad Flat Pack | 24 | SOT307-2 |
| | P87C52IFF FA | UV | –40 to +85, Ceramic Dual In-line Package | 24 | 0590B |
| P80C32NBA A | | | 0 to +70, Plastic Leaded Chip Carrier | 33 | SOT187-2 |
| P80C32NBP N | | | 0 to +70, Plastic Dual In-line Package | 33 | SOT129-1 |
| P80C32NBB B | | | 0 to +70, Plastic Quad Flat Pack | 33 | SOT307-2 |
| P80C32NFA A | | | –40 to +85, Plastic Leaded Chip Carrier | 33 | SOT187-2 |
| P80C32NFP N | | | –40 to +85, Plastic Dual In-line Package | 33 | SOT129-1 |
| P80C32NFB B | | | –40 to +85, Plastic Quad Flat Pack | 33 | SOT307-2 |

**NOTE:**

1. OTP = One Time Programmable EPROM. UV = UV erasable EPROM
2. For 33MHz ROM 80C52 operation, see 80C52/80C54/80C58 data sheet.

## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

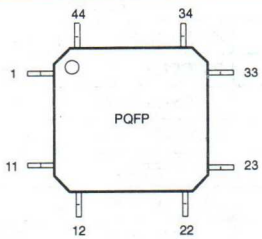



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | NC* | 16 | T0/P3.4 | 31 | P2.7/A15 |
| 2 | T2/P1.0 | 17 | T1/P3.5 | 32 | $\overline{PSEN}$ |
| 3 | T2EX/P1.1 | 18 | $\overline{WR}$/P3.6 | 33 | ALE/$\overline{PROG}$ |
| 4 | P1.2 | 19 | $\overline{RD}$/P3.7 | 34 | NC* |
| 5 | P1.3 | 20 | XTAL2 | 35 | $\overline{EA}$/$V_{PP}$ |
| 6 | P1.4 | 21 | XTAL1 | 36 | P0.7/AD7 |
| 7 | P1.5 | 22 | $V_{SS}$ | 37 | P0.6/AD6 |
| 8 | P1.6 | 23 | NC* | 38 | P0.5/AD5 |
| 9 | P1.7 | 24 | P2.0/A8 | 39 | P0.4/AD4 |
| 10 | RST | 25 | P2.1/A9 | 40 | P0.3/AD3 |
| 11 | RxD/P3.0 | 26 | P2.2/A10 | 41 | P0.2/AD2 |
| 12 | NC* | 27 | P2.3/A11 | 42 | P0.1/AD1 |
| 13 | TxD/P3.1 | 28 | P2.4/A12 | 43 | P0.0/AD0 |
| 14 | $\overline{INT0}$/P3.2 | 29 | P2.5/A13 | 44 | $V_{CC}$ |
| 15 | $\overline{INT1}$/P3.3 | 30 | P2.6/A14 | | |

* DO NOT CONNECT

SU00061

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS

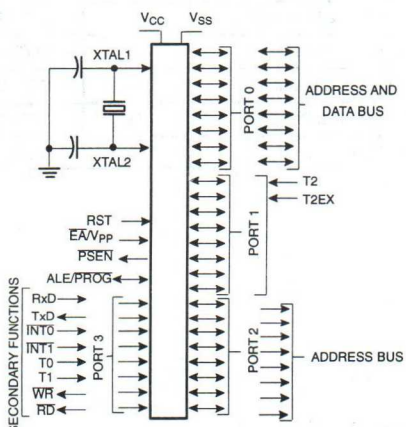



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | P1.5 | 16 | $V_{SS}$ | 31 | P0.6/AD6 |
| 2 | P1.6 | 17 | NC* | 32 | P0.5/AD5 |
| 3 | P1.7 | 18 | P2.0/A8 | 33 | P0.4/AD4 |
| 4 | RST | 19 | P2.1/A9 | 34 | P0.3/AD3 |
| 5 | RxD/P3.0 | 20 | P2.2/A10 | 35 | P0.2/AD2 |
| 6 | NC* | 21 | P2.3/A11 | 36 | P0.1/AD1 |
| 7 | TxD/P3.1 | 22 | P2.4/A12 | 37 | P0.0/AD0 |
| 8 | $\overline{INT0}$/P3.2 | 23 | P2.5/A13 | 38 | $V_{CC}$ |
| 9 | $\overline{INT1}$/P3.3 | 24 | P2.6/A14 | 39 | NC* |
| 10 | T0/P3.4 | 25 | P2.7/A15 | 40 | T2/P1.0 |
| 11 | T1/P3.5 | 26 | $\overline{PSEN}$ | 41 | T2EX/P1.1 |
| 12 | $\overline{WR}$/P3.6 | 27 | ALE/$\overline{PROG}$ | 42 | P1.2 |
| 13 | $\overline{RD}$/P3.7 | 28 | NC* | 43 | P1.3 |
| 14 | XTAL2 | 29 | $\overline{EA}$/$V_{PP}$ | 44 | P1.4 |
| 15 | XTAL1 | 30 | P0.7/AD7 | | |

* DO NOT CONNECT

SU00062

## LOGIC SYMBOL




SU00063

## BLOCK DIAGRAM

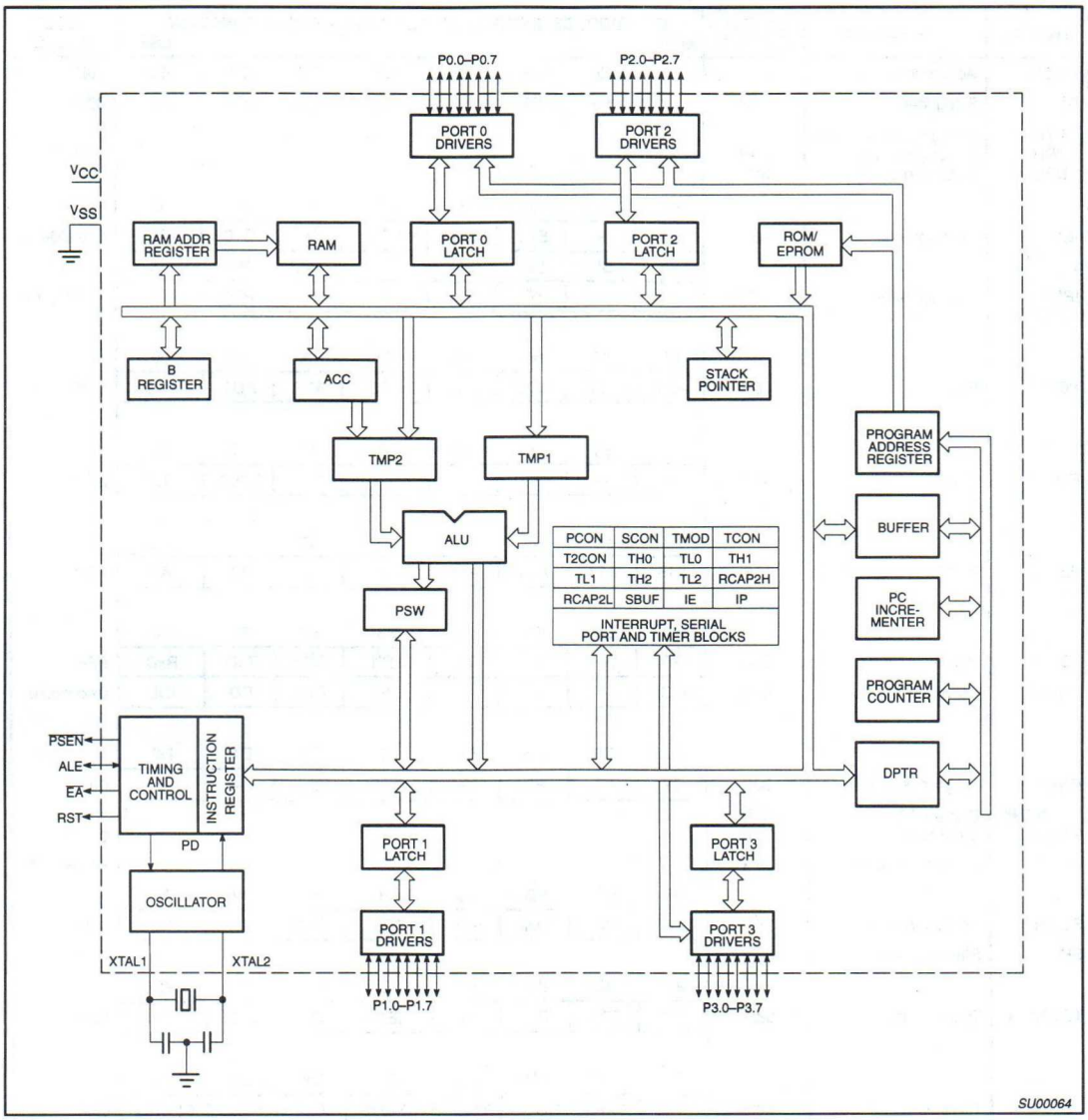


**Table 1. 8XC52 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes) | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE* | Interrupt enable | A8H | EA | – | ET2 | ES | ET1 | EX1 | ET0 | EX0 | 0x000000B |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP* | Interrupt priority | B8H | – | – | PT2 | PS | PT1 | PX1 | PT0 | PX0 | xx000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | – | – | – | – | – | – | T2EX | T2 | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TxD | RxD | FFH |
| PCON[1] | Power control | 87H | SMOD | – | – | – | GF1 | GF0 | PD | IDL | 0xxxxxxxB |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| RCAP2H# | Capture high | CBH | | | | | | | | | 00H |
| RCAPL# | Capture low | CAH | | | | | | | | | 00H |
| SBUF | Serial data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial controller | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| T2CON*# | Timer 2 control | C8H | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/$\overline{T2}$ | CP/$\overline{RL2}$ | 00H |
| TH0 | Timer high 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer high 1 | 8DH | | | | | | | | | 00H |
| TH2# | Timer high 2 | CDH | | | | | | | | | 00H |
| TL0 | Timer low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer low 1 | 8BH | | | | | | | | | 00H |
| TL2# | Timer low 2 | CCH | | | | | | | | | 00H |
| TMOD | Timer mode | 89H | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 | 00H |

* Bit addressable

\# SFRs are modified from or added to the 80C51 SFRs.

1. Bits GF1, GF0, PD, and IDL of the PCON register are not implemented in the NMOS 8XC52.

## PIN DESCRIPTION

| MNEMONIC | PIN NO. DIP | PIN NO. LCC | PIN NO. QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification in the 87C52. External pull-ups are required during program verification. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Pins P1.0 and P1.1 also. Port 1 also receives the low-order address byte during program memory verification. Port 1 also serves alternate functions for timer 2: |
| | 1 | 2 | 40 | I | **T2 (P1.0):** Timer/counter 2 external count input. |
| | 2 | 3 | 41 | I | **T2EX (P1.1):** Timer/counter 2 trigger input. |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{CC}$. |
| ALE/$\overline{PROG}$ | 30 | 33 | 27 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the device is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}$/$V_{PP}$ | 31 | 35 | 29 | I | **External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H to 1FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 1FFFH. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

## DIFFERENCES FROM THE 80C51

### Special Function Registers

The special function register space is the same as the 80C51 except that the 80C32/87C52 contains the additional special function registers T2CON, RCAP2L, RCAP2H, TL2, and TH2. Since the standard 80C51 on-chip functions are identical in the 8XC52, the SFR locations, bit locations, and operation are likewise identical. The only exceptions are in the interrupt mode and interrupt priority SFRs (see Table 1).

### Timer/Counters

In addition to timer/counters 0 and 1 of the 80C51, the 80C32/87C52 contains timer/counter 2. Like timers 0 and 1, timer 2 can operate as either an event timer or as an event counter. This is selected by bit C/T2 in the special function register T2CON (see Figure 1). It has three operating modes: capture, auto-load, and baud rate generator, which are selected by bits in the T2CON as shown in Table 2.

In the Capture Mode there are two options which are selected by bit EXEN2 in T2CON. If EXEN2 = 0, then Timer 2 is a 16-bit timer or counter which upon overflowing sets bit TF2, the Timer 2 overflow bit, which can be used to generate an interrupt. If EXEN2 = 1, then Timer 2 still does the above, but with the added feature that a 1-to-0 transition at external input T2EX causes the current value in the Timer 2 registers, TL2 and TH2, to be captured into registers RCAP2L and RCAP2H, respectively. (RCAP2L and RCAP2H are new special function registers in the 80C52.) In addition, the transition at T2EX causes bit EXF2 in T2CON to be set, and EXF2 like TF2 can generate an interrupt. The Capture Mode is illustrated in Figure 2.

In the auto-reload mode, there are again two options, which are selected by bit EXEN2 in T2CON. If EXEN2 = 0, then when Timer 2 rolls over it not only sets TF2 but also causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2L and RCAP2H, which are preset by software. If EXEN2 = 1, then Timer 2 still does the above, but with the added feature that a 1-to-0 transition at external input T2EX will also trigger the 16-bit reload and set EXF2. The auto-reload mode is illustrated in Figure 3.

The baud rate generation mode is selected by RCLK = 1 and/or TCLK = 1. It will be described in conjunction with the serial port.

### Serial Port

The serial port of the 8XC52 is identical to that of the 80C51 except that counter/timer 2 can be used to generate baud rates.

In the 8XC52, Timer 2 is selected as the baud rate generator by setting TCLK and/or RCLK in T2CON (see Figure 1). Note that the baud rate for transmit and receive can be simultaneously different. Setting RCLK and/or TCLK puts Timer into its baud rate generator mode, as shown in Figure 4.

The baud rate generator mode is similar to the auto-reload mode, in that a rollover in TH2 causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2H and RCAP2L, which are preset by software.

Now, the baud rates in Modes 1 and 3 are determined by Timer 2's overflow rate as follows:

$$\text{Modes 1, 3 Baud Rate} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

The timer can be configured for either "timer" or "counter" operation. In the most typical applications, it is configured for "timer" operation (C/T2 = 0). "Timer" operation is a little different for Timer 2 when it's being used as a baud rate generator. Normally, as a timer it would increment every machine cycle (thus at 1/12 the oscillator frequency). As a baud rate generator, however, it increments every state time (thus at 1/2 the oscillator frequency). In that case the baud rate is given by the formula:

$$\text{Modes 1, 3 Baud Rate} = \frac{\text{Oscillator Frequency}}{32 \times [65536 - (\text{RCAP2H, RCAP2L})]}$$

where (RCAP2H, RCAP2L) is the content of RCAP2H and RCAP2L taken as a 16-bit unsigned integer.

| (MSB) | | | | | | | (LSB) |
|---|---|---|---|---|---|---|---|
| TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/T2 | CP/RL2 |

| Symbol | Position | Name and Significance |
|---|---|---|
| TF2 | T2CON.7 | Timer 2 overflow flag set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK or TCLK = 1. |
| EXF2 | T2CON.6 | Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software. |
| RCLK | T2CON.5 | Receive clock flag. When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in modes 1 and 3. RCLK = 0 causes Timer 1 overflow to be used for the receive clock. |
| TCLK | T2CON.4 | Transmit clock flag. When set, causes the serial port to use Timer 2 overflow pulses for its transmit clock in modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock. |
| EXEN2 | T2CON.3 | Timer 2 external enable flag. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX. |
| TR2 | T2CON.2 | Start/stop control for Timer 2. A logic 1 starts the timer. |
| C/T2 | T2CON.1 | Timer or counter select. (Timer 2)<br>0 = Internal timer (OSC/12)<br>1 = External event counter (falling edge triggered). |
| CP/RL2 | T2CON.0 | Capture/Reload flag. When set, captures will occur on negative transitions at T2EX if EXEN2 = 1. When cleared, auto-reloads will occur either with Timer 2 overflows or negative transitions at T2EX when EXEN2 = 1. When either RCLK = 1 or TCLK = 1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow. |

SU00065

**Figure 1. Timer/Counter 2 (T2CON) Control Register**

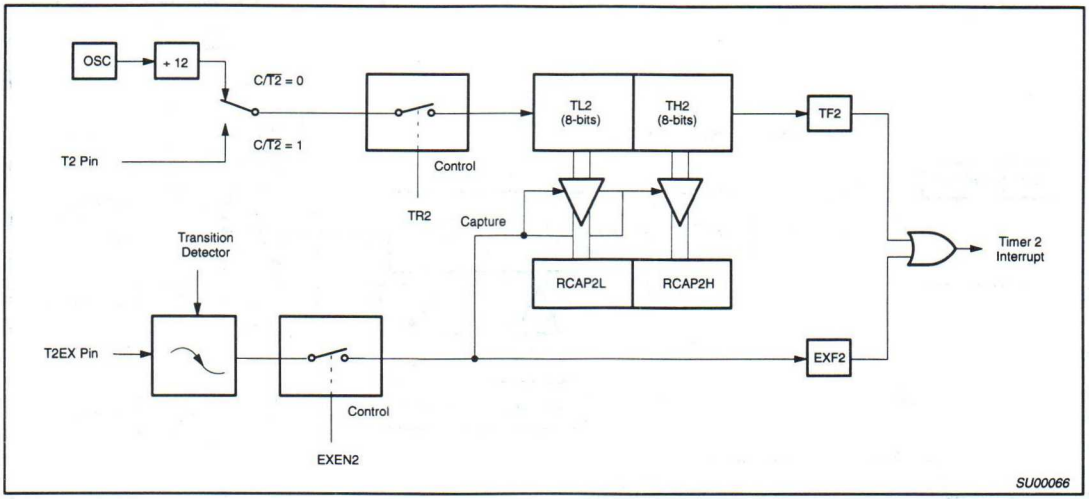



**Figure 2. Timer 2 in Capture Mode**

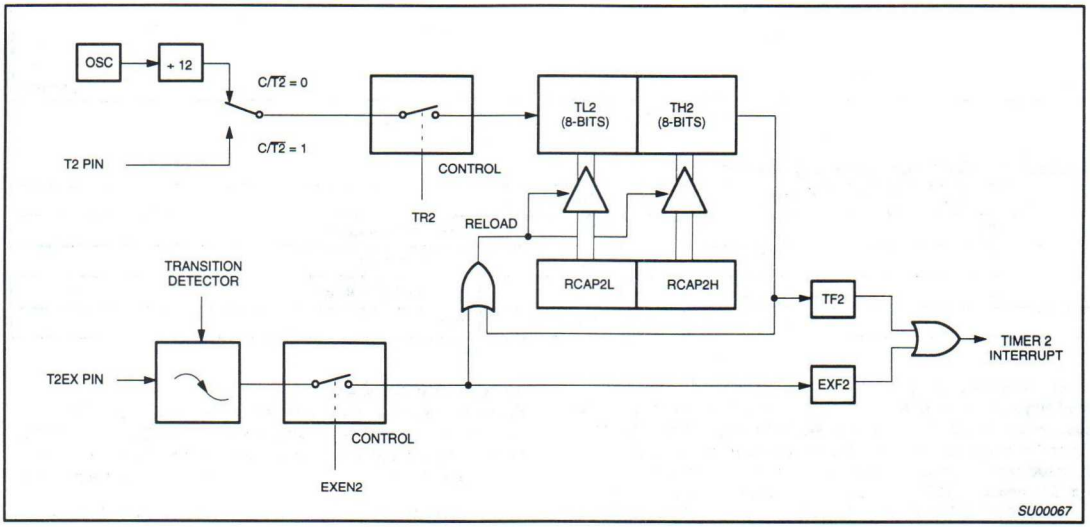



**Figure 3. Timer 2 in Auto-Reload Mode**

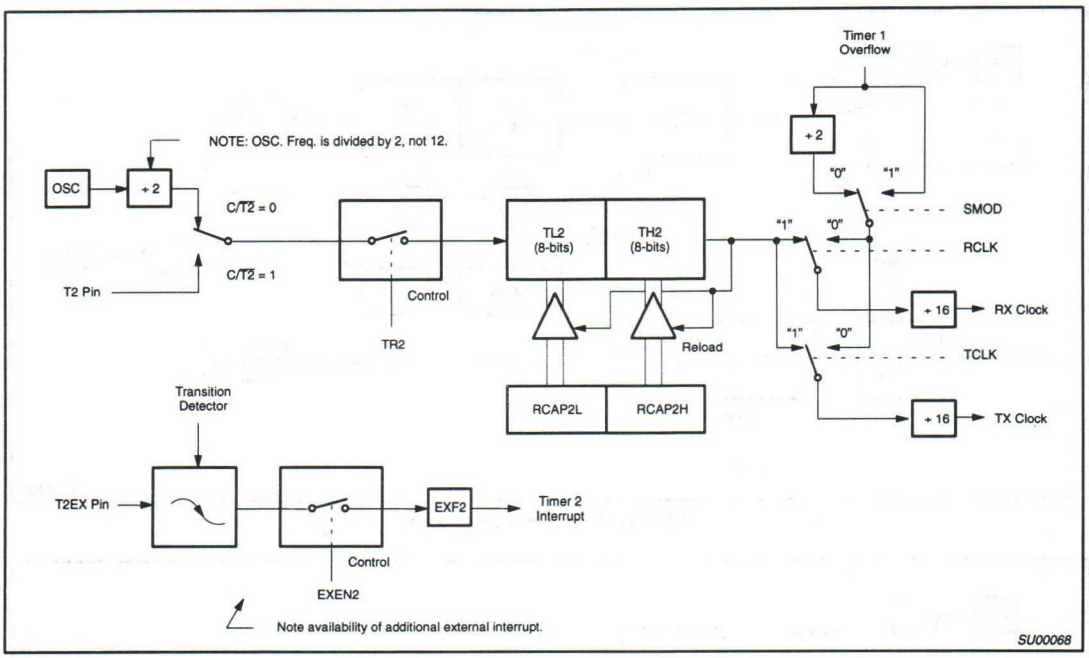



**Figure 4. Timer 2 in Baud Rate Generator Mode**

**Table 2. Timer 2 Operating Modes**

| RCLK + TCLK | CP/$\overline{RL2}$ | TR2 | MODE |
|---|---|---|---|
| 0 | 0 | 1 | 16-bit Auto-reload |
| 0 | 1 | 1 | 16-bit Capture |
| 1 | X | 1 | Baud rate generator |
| X | X | 0 | (off) |

Timer 2 as a baud rate generator is shown in Figure 4. This figure is valid only if RCLK + TCLK = 1 in T2CON. Note that a rollover in TH2 does not set TF2, and will not generate an interrupt. Therefore, the Timer 2 interrupt does not have to be disabled when Timer 2 is in the baud rate generator mode. Note too, that if EXEN2 is set, a 1-to-0 transition in T2EX will set EXF2 but will not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Thus when Timer 2 is in use as a baud rate generator, T2EX can be used as an extra external interrupt, if desired.

It should be noted that when Timer 2 is running (TR2 = 1) in "timer" function in the baud rate generator mode, one should not try to read or write TH2 or TL2. Under these conditions the timer is being incremented every state time, and the results of a read or write may not be accurate. The RCAP registers may be read, but should not be written to, because a write might overlap a reload and cause write and/or reload errors. Turn the timer off (clear TR2) before accessing the Timer 2 or RCAP registers, in this case.

## Timer/Counter 2 Set-up

Except for the baud rate generator mode, the values given for T2CON do not include the setting of the TR2 bit. Therefore, bit TR2 must be set, separately, to turn the timer on. See Table 3 for set-up of timer 2 as a timer. See Table 4 for set-up of timer 2 as a counter.

## Using Timer/Counter 2 to Generate Baud Rates

For this purpose, Timer 2 must be used in the baud rate generating mode. If Timer 2 is being clocked through pin T2 (P1.0) the baud rate is:

$$\text{Baud Rate} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

And if it is being clocked internally, the baud rate is:

$$\text{Baud Rate} = \frac{\text{Oscillator Frequency}}{32 \times [65536 - (\text{RCAP2H, RCAP2L})]}$$

To obtain the reload value for RCAP2H and RCA02L, the above equation can be rewritten as:

$$\text{RCAP2H, RCAP2L} = 65536 - \frac{\text{Oscillator Frequency}}{32 \times \text{Baud Rate}}$$

## Interrupts

The 80C32/87C52 has 6 interrupt sources. All except TF2 and EXF2 are identical sources to those in the 80C51.

The Interrupt Enable Register and the Interrupt Priority Register are modified to include the additional 80C32/87C52 interrupt sources. The operation of these registers is identical to the 80C51.

In the 80C32/87C52, the Timer 2 Interrupt is generated by the logical OR of TF2 and EXF2. Neither of these flags is cleared by hardware when the service routine is vectored to. In fact, the service routine may have to determine whether it was TF2 or EXF2 that generated the interrupt, and the bit will have to be cleared in software.

All of the bits that generate interrupts can be set or cleared by software, with the same result as though it has been set or cleared by hardware. That is, interrupts can be generated or pending interrupts can be canceled in software.

The interrupt vector addresses and the interrupt priority for requests in the same priority level are given in the following:

| Source | Vector Address | Priority Within Level |
|---|---|---|
| 1. IE0 | 0003H | (highest) |
| 2. TF0 | 000BH | |
| 3. IE1 | 0013H | |
| 4. TF1 | 001BH | |
| 5. RI + TI | 0023H | |
| 6. TF2 + EXF2 | 002BH | (lowest) |

Note that they are identical to those in the 80C51 except for the addition of the Timer 2 (TF1 and EXF2) interrupt at 002BH and at the lowest priority within a level.

**Table 3. Timer 2 as a Timer**

| MODE | T2CON | |
|---|---|---|
| | INTERNAL CONTROL (Note 1) | EXTERNAL CONTROL (Note 2) |
| 16-bit Auto-Reload | 00H | 08H |
| 16-bit Capture | 01H | 09H |
| Baud rate generator receive and transmit same baud rate | 34H | 36H |
| Receive only | 24H | 26H |
| Transmit only | 14H | 16H |

**Table 4. Timer 2 as a Counter**

| MODE | TMOD | |
|---|---|---|
| | INTERNAL CONTROL (Note 1) | EXTERNAL CONTROL (Note 2) |
| 16-bit | 02H | 0AH |
| Auto-Reload | 03H | 0BH |

**NOTES:**
1. Capture/reload occurs only on timer/counter overflow.
2. Capture/reload occurs on timer/counter overflow and a 1-to-0 transition on T2EX (P1.1) pin except when timer 2 is used in the baud rate generator mode.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the Logic Symbol, page 3-168.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-up reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles.

## IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. the control bits for the reduced power modes are in the special function register PCON.

## DESIGN CONSIDERATIONS

At power-on, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up.

Table 5 shows the state of I/O ports during low current operating modes.

As a precaution to coming out of an unexpected power down, $\overline{INT0}$ and $\overline{INT1}$ should be disabled prior to enterring power down.

**Table 5. External Pin Status During Idle and Power-Down Modes**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

**Electrical Deviations from Commercial Specifications for Extended Temperature Range (87C52)**
DC and AC parameters not included here are the same as in the commercial temperature range table.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$ | | –0.5 | $0.2V_{CC}$–0.15 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | | 0 | $0.2V_{CC}$–0.35 | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST | | $0.2V_{CC}$+1 | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage to XTAL1, RST | | $0.7V_{CC}$+0.1 | $V_{CC}$+0.5 | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3 | $V_{IN}$ = 0.45V | | –75 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3 | $V_{IN}$ = 2.0V | | –750 | µA |
| $I_{CC}$ | Power supply current:<br>Active mode<br>Idle mode<br>Power-down mode | $V_{CC}$ = 4.5–5.5V,<br>Frequency range =<br>3.5 to 16MHz | | <br>32<br>5<br>50 | <br>mA<br>mA<br>µA |

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | 0 to +70 or –40 to +85 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}$/$V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Maximum $I_{OL}$ per I/O pin | 15 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.5 | W |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V (87C52)
$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V (80C32)

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$[7] | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$[7] | | 0 | | $0.2V_{CC}$–0.3 | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST[7] | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST[7] | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3[9] | $I_{OL}$ = 1.6mA[2] | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$[9] | $I_{OL}$ = 3.2mA[2] | | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3, ALE, $\overline{PSEN}$[3] | $I_{OH}$ = –60µA,<br>$I_{OH}$ = –25µA<br>$I_{OH}$ = –10µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$ | | | V<br>V<br>V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode) | $I_{OH}$ = –800µA,<br>$I_{OH}$ = –300µA<br>$I_{OH}$ = –80µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$ | | | V<br>V<br>V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3[7] | $V_{IN}$ = 0.45V | | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3[7] | See note 4 | | | –650 | µA |
| $I_{LI}$ | Input leakage current, port 0 | $V_{IN} = V_{IL}$ or $V_{IH}$ | | | ±10 | µA |
| $I_{CC}$ | Power supply current:[7]<br>Active mode @ 16MHz[5]<br>Idle mode @ 16MHz<br>Power-down mode $T_{amb}$ = 0 to 70°C<br>$T_{amb}$ = –40 to +85°C | See note 6 | | <br>11.5<br>1.3<br>3 | <br>32<br>5<br>50<br>75 | <br>mA<br>mA<br>µA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | 50 | | 300 | kΩ |
| $C_{IO}$ | Pin capacitance[10] | | | | 15 | pF |

**NOTES:**

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
5. $I_{CCMAX}$ at other frequencies is given by: Active mode: $I_{CCMAX}$ = 1.5 × FREQ + 8.0: Idle mode: $I_{CCMAX}$ = 0.14 × FREQ +2.31, where FREQ is the external oscillator frequency in MHz. $I_{CCMAX}$ is given in mA. See Figure 12.
6. See Figures 13 through 16 for $I_{CC}$ test conditions.
7. These values apply only to $T_{amb}$ = 0°C to +70°C. For $T_{amb}$ = –40°C to +85°C, see table on previous page.
8. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
9. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 15mA (*NOTE: This is 85°C specification.)
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 67mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
10. This limit is for plastic packages. For ceramic packages, the maximum limit is 20pF.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V (87C52)[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 5 | Oscillator frequency<br>Speed versions : E | | | 3.5 | 16 | MHz |
| $t_{LHLL}$ | 5 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 5 | Address valid to ALE low | 22 | | $t_{CLCL}-40$ | | ns |
| $t_{LLAX}$ | 5 | Address hold after ALE low | 32 | | $t_{CLCL}-30$ | | ns |
| $t_{LLIV}$ | 5 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 5 | ALE low to $\overline{PSEN}$ low | 32 | | $t_{CLCL}-30$ | | ns |
| $t_{PLPH}$ | 5 | $\overline{PSEN}$ pulse width | 142 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 5 | $\overline{PSEN}$ low to valid instruction in | | 82 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 5 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 5 | Input instruction float after $\overline{PSEN}$ | | 37 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 5 | Address to valid instruction in | | 207 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 5 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 6, 7 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 6, 7 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 6, 7 | $\overline{RD}$ low to valid data in | | 147 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 6, 7 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 6, 7 | Data float after $\overline{RD}$ | | 65 | | $2t_{CLCL}-60$ | ns |
| $t_{LLDV}$ | 6, 7 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 6, 7 | Address to valid data in | | 397 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 6, 7 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 137 | 239 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 6, 7 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 122 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 6, 7 | Data valid to $\overline{WR}$ transition | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{WHQX}$ | 6, 7 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{QVWH}$ | 7 | Data valid to $\overline{WR}$ high | 287 | | $7t_{CLCL}-150$ | | ns |
| $t_{RLAZ}$ | 6, 7 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 6, 7 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 9 | High time | 20 | | 20 | $t_{CLCL}-t_{CLCX}$ | ns |
| $t_{CLCX}$ | 9 | Low time | 20 | | 20 | $t_{CLCL}-t_{CHCX}$ | ns |
| $t_{CLCH}$ | 9 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 9 | Fall time | | 20 | | 20 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 8 | Serial port clock cycle time | 750 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 8 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 8 | Output data hold after clock rising edge | 8 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 8 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 8 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}-133$ | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 80C32/52 to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.
4. See application note AN457 for external memory interface.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 24MHz CLOCK | | VARIABLE CLOCK | | 33MHz CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 5 | Oscillator frequency<br>Speed versions : I<br>: N | | | 3.5 | 24 | <br>3.5 | <br>33 | MHz |
| $t_{LHLL}$ | 5 | ALE pulse width | 43 | | $2t_{CLCL}$–40 | | 21 | | ns |
| $t_{AVLL}$ | 5 | Address valid to ALE low | 17 | | $t_{CLCL}$–25 | | 5 | | ns |
| $t_{LLAX}$ | 5 | Address hold after ALE low | 17 | | $t_{CLCL}$–25 | | 5 | | ns |
| $t_{LLIV}$ | 5 | ALE low to valid instruction in | | 102 | | $4t_{CLCL}$–65 | | 56 | ns |
| $t_{LLPL}$ | 5 | ALE low to $\overline{PSEN}$ low | 17 | | $t_{CLCL}$–25 | | 5 | | ns |
| $t_{PLPH}$ | 5 | $\overline{PSEN}$ pulse width | 80 | | $3t_{CLCL}$–45 | | | 46 | ns |
| $t_{PLIV}$ | 5 | $\overline{PSEN}$ low to valid instruction in | | 65 | | $3t_{CLCL}$–60 | | 31 | ns |
| $t_{PXIX}$ | 5 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 5 | Input instruction float after $\overline{PSEN}$ | | 17 | | $t_{CLCL}$–25 | | 5 | ns |
| $t_{AVIV}$ | 5 | Address to valid instruction in | | 128 | | $5t_{CLCL}$–80 | | 72 | ns |
| $t_{PLAZ}$ | 5 | $\overline{PSEN}$ low to address float | | 10 | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | | | |
| $t_{RLRH}$ | 6, 7 | $\overline{RD}$ pulse width | 150 | | $6t_{CLCL}$–100 | | 82 | | ns |
| $t_{WLWH}$ | 6, 7 | $\overline{WR}$ pulse width | 150 | | $6t_{CLCL}$–100 | | 82 | | ns |
| $t_{RLDV}$ | 6, 7 | $\overline{RD}$ low to valid data in | | 118 | | $5t_{CLCL}$–90 | | 62 | ns |
| $t_{RHDX}$ | 6, 7 | Data hold after $\overline{RD}$ | 0 | | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 6, 7 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}$–28 | | 33 | ns |
| $t_{LLDV}$ | 6, 7 | ALE low to valid data in | | 183 | | $8t_{CLCL}$–150 | | 92 | ns |
| $t_{AVDV}$ | 6, 7 | Address to valid data in | | 210 | | $9t_{CLCL}$–165 | | 108 | ns |
| $t_{LLWL}$ | 6, 7 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 75 | 175 | $3t_{CLCL}$–50 | $3t_{CLCL}$+50 | 41 | 141 | ns |
| $t_{AVWL}$ | 6, 7 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 92 | | $4t_{CLCL}$–75 | | 46 | | ns |
| $t_{QVWX}$ | 6, 7 | Data valid to $\overline{WR}$ transition | 12 | | $t_{CLCL}$–30 | | 0.3 | | ns |
| $t_{WHQX}$ | 6, 7 | Data hold after $\overline{WR}$ | 17 | | $t_{CLCL}$–25 | | 5 | | ns |
| $t_{QVWH}$ | 7 | Data valid to $\overline{WR}$ high | 162 | | $7t_{CLCL}$–130 | | 82 | | ns |
| $t_{RLAZ}$ | 6, 7 | $\overline{RD}$ low to address float | | 0 | | 0 | | 0 | ns |
| $t_{WHLH}$ | 6, 7 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 17 | 67 | $t_{CLCL}$–25 | $t_{CLCL}$+25 | 5 | 5 | ns |
| **External Clock** | | | | | | | | | |
| $t_{CHCX}$ | 9 | High time | 17 | | 17 | $t_{CLCL}$–$t_{CLCX}$ | | | ns |
| $t_{CLCX}$ | 9 | Low time | 17 | | 17 | $t_{CLCL}$–$t_{CHCX}$ | | | ns |
| $t_{CLCH}$ | 9 | Rise time | | 5 | | 5 | | | ns |
| $t_{CHCL}$ | 9 | Fall time | | 5 | | 5 | | | ns |
| **Shift Register** | | | | | | | | | |
| $t_{XLXL}$ | 8 | Serial port clock cycle time | 505 | | $12t_{CLCL}$ | | 363 | | ns |
| $t_{QVXH}$ | 8 | Output data setup to clock rising edge | 283 | | $10t_{CLCL}$–133 | | 170 | | ns |
| $t_{XHQX}$ | 8 | Output data hold after clock rising edge | 3 | | $2t_{CLCL}$–80 | | 19 | | ns |
| $t_{XHDX}$ | 8 | Input data hold after clock rising edge | 0 | | 0 | | 0 | | ns |
| $t_{XHDV}$ | 8 | Clock rising edge to input data valid | | 283 | | $10t_{CLCL}$–133 | | 170 | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 8XC52 to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.
4. Variable clock is specified for oscillator frequencies greater than 16MHz to 33MHz. For frequencies equal or less than 16MHz, see 16MHz "AC Electrial Characteristics", page 3-179.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{PSEN}$ low.

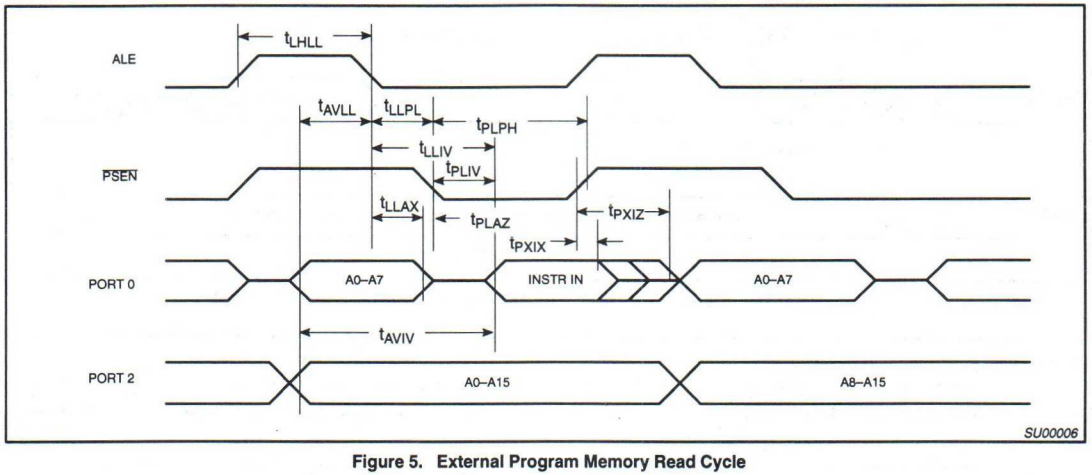



**Figure 5. External Program Memory Read Cycle**

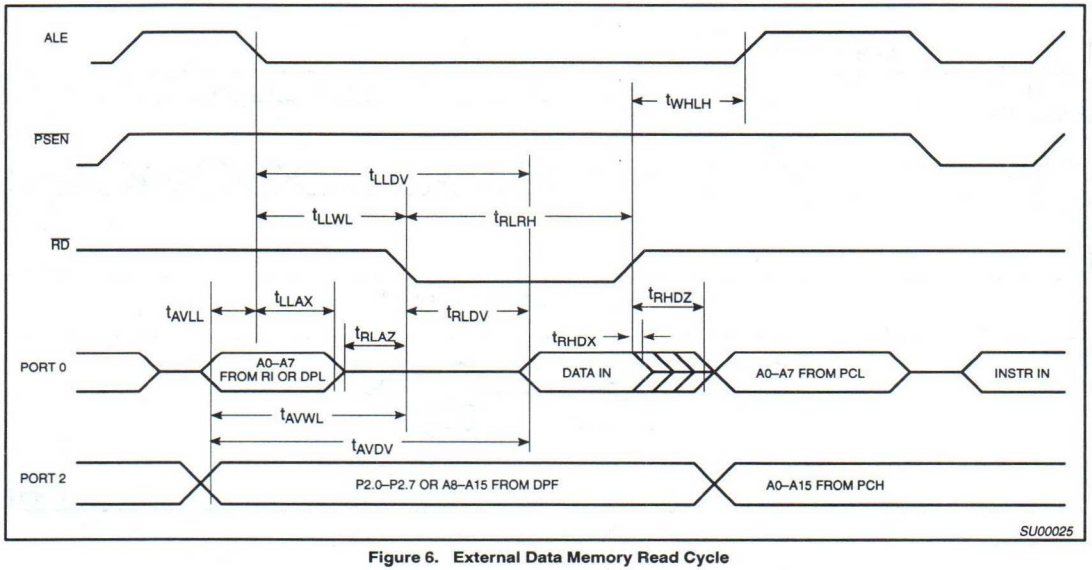



**Figure 6. External Data Memory Read Cycle**

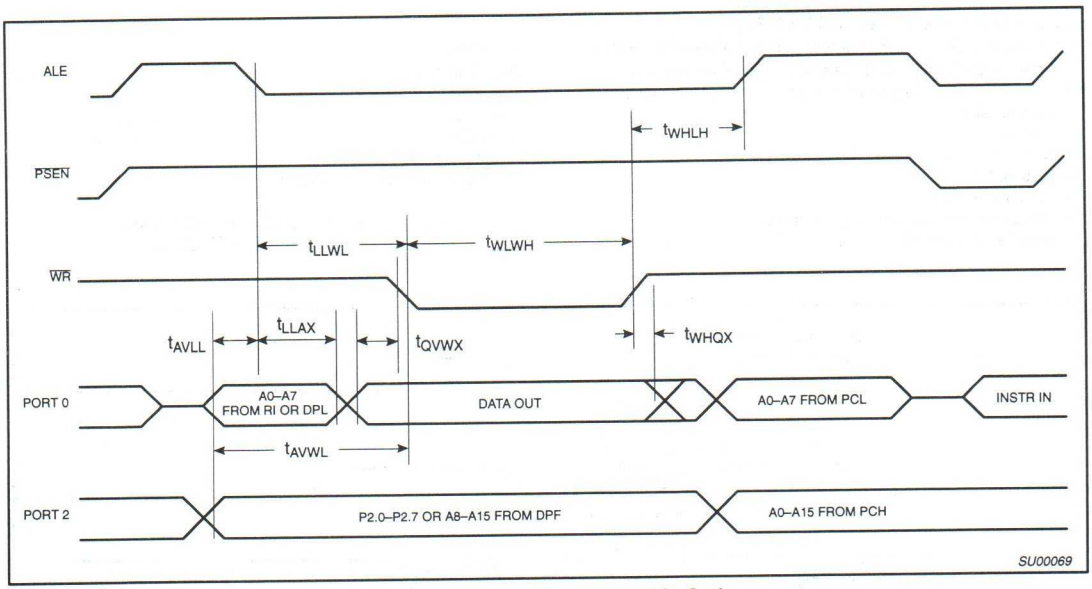



**Figure 7. External Data Memory Write Cycle**

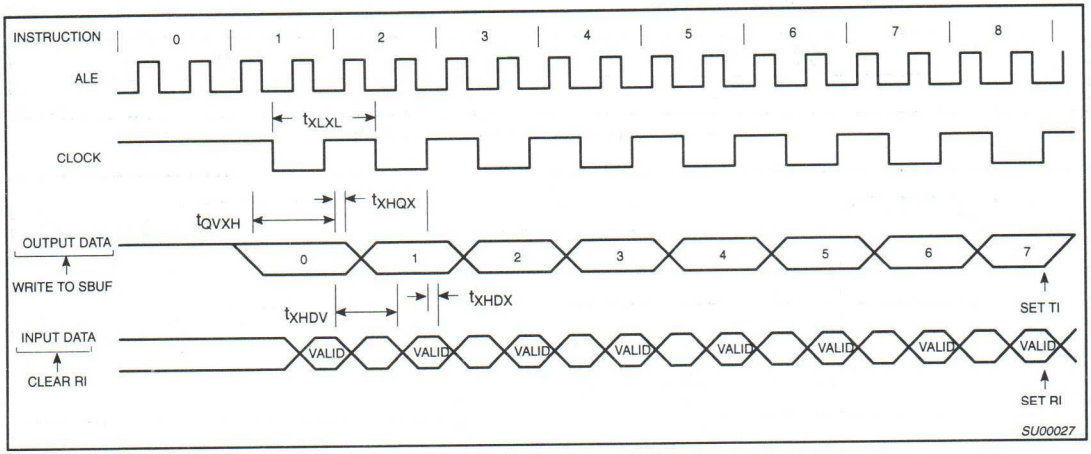



**Figure 8. Shift Register Mode Timing**

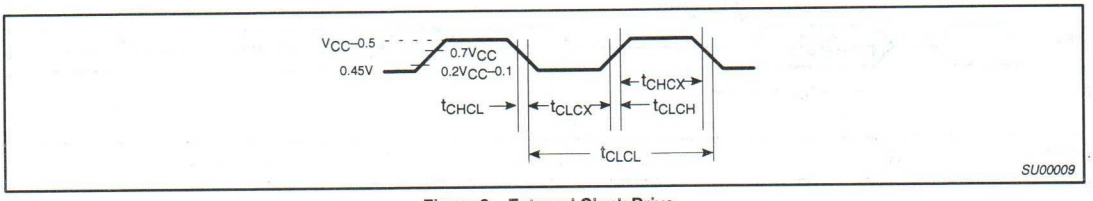
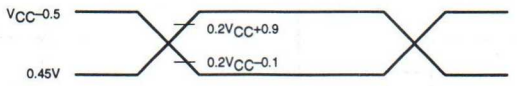


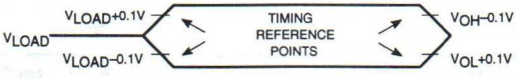

**Figure 9. External Clock Drive**




NOTE:
AC inputs during testing are driven at $V_{CC}$ –0.5 for a logic '1' and 0.45V for a logic '0'. Timing measurements are made at $V_{IH}$ min for a logic '1' and $V_{IL}$ max for a logic '0'.

SU00010

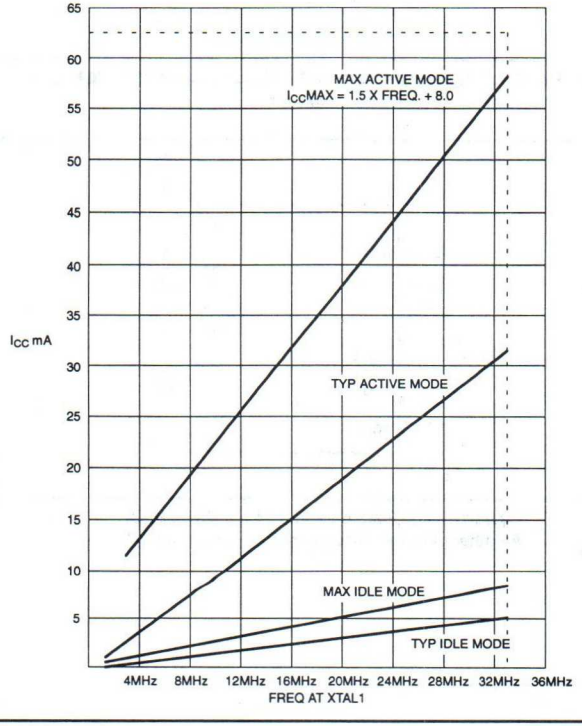
**Figure 10. AC Testing Input/Output**




NOTE:
For timing purposes, a port is no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded $V_{OH}/V_{OL}$ level occurs. $I_{OH}/I_{OL} \geq \pm 20$mA.

SU00011

**Figure 11. Float Waveform**




SU00070B

**Figure 12. $I_{CC}$ vs. FREQ**
**Valid only within frequency specifications of the device under test**

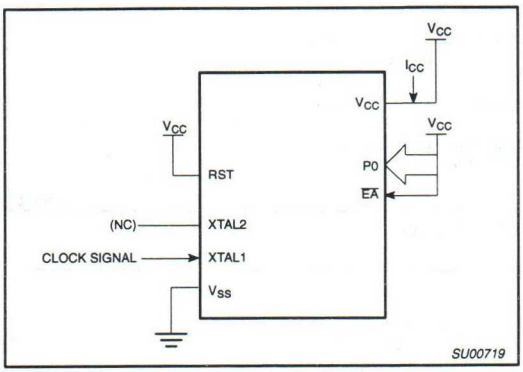



**Figure 13. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

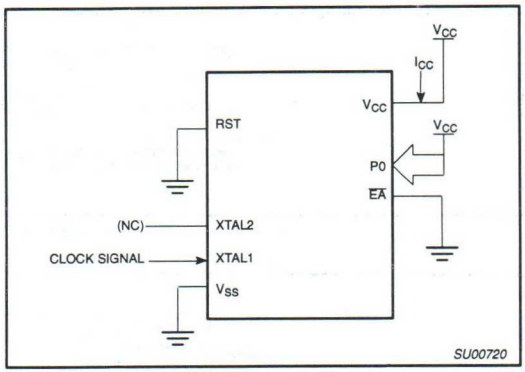



**Figure 14. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

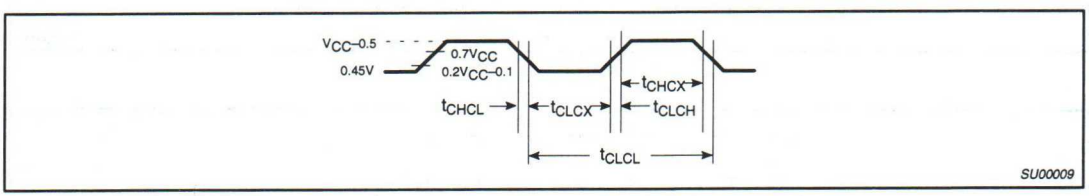



**Figure 15. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

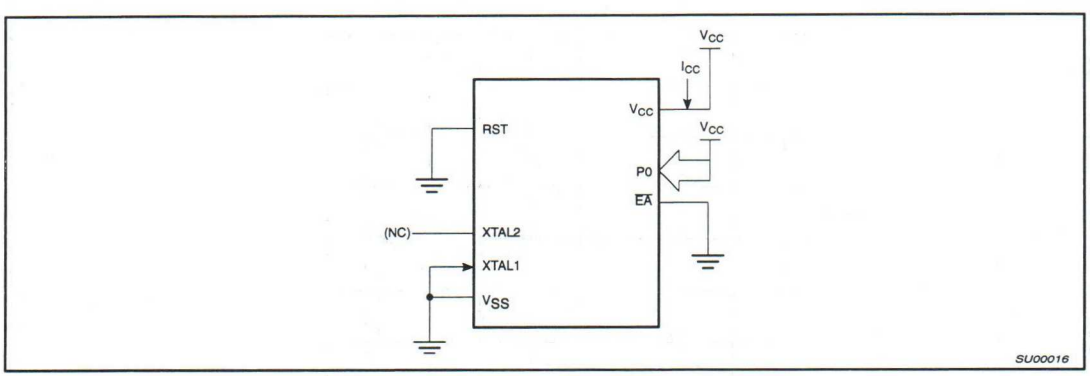



**Figure 16. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2V to 5.5V**

## EPROM CHARACTERISTICS

The 87C52 is programmed by using a modified Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C52 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C52 manufactured by Philips.

Table 6 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the security bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 17 and 18. Figure 19 shows the circuit configuration for normal program memory verification.

### Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 17. Note that the 87C52 is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1 and 2, as shown in Figure 17. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 3 specified in Table 6 are held at the 'Program Code Data' levels indicated in Table 6. The ALE/$\overline{PROG}$ is pulsed low 25 times as shown in Figure 18.

To program the encryption table, repeat the 25 pulse programming sequence for addresses 0 through 1FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the security bits, repeat the 25 pulse programming sequence using the 'Pgm Security Bit' levels. After one security bit is programmed, further programming of the code memory and encryption table is disabled. However, the other security bit can still be programmed.

Note that the $\overline{EA}$/$V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

#### Program Verification

If security bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 1 and 2 as shown in Figure 19. The other pins are held at the 'Verify Code Data' levels indicated in Table 6. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

#### Reading the Signature Bytes

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:
(030H) = 15H indicates manufactured by Philips
(031H) = 97H indicates 87C52

### Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 6, and which satisfies the timing specifications, is suitable.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-s/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

### Table 6. EPROM Programming Modes

| MODE | RST | $\overline{PSEN}$ | ALE/$\overline{PROG}$ | $\overline{EA}$/$V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm security bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm security bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V±10% during programming and verification.
4. *ALE/$\overline{PROG}$ receives 25 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

™Trademark phrase of Intel Corporation.

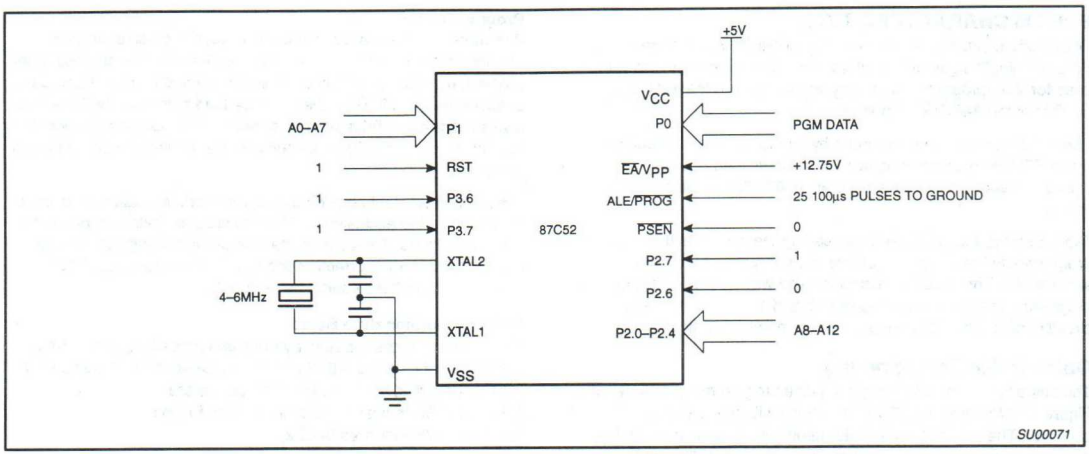



**Figure 17. Programming Configuration**

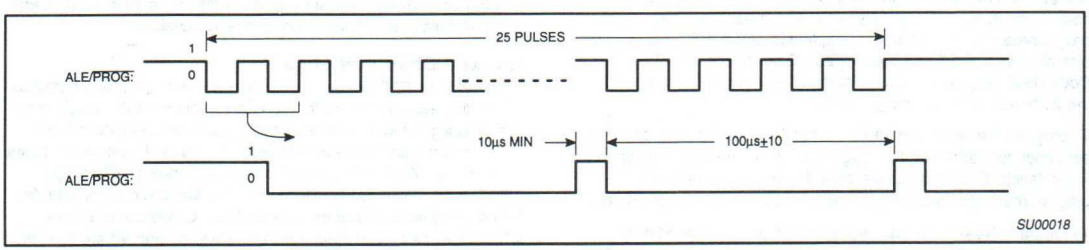



**Figure 18. PROG Waveform**

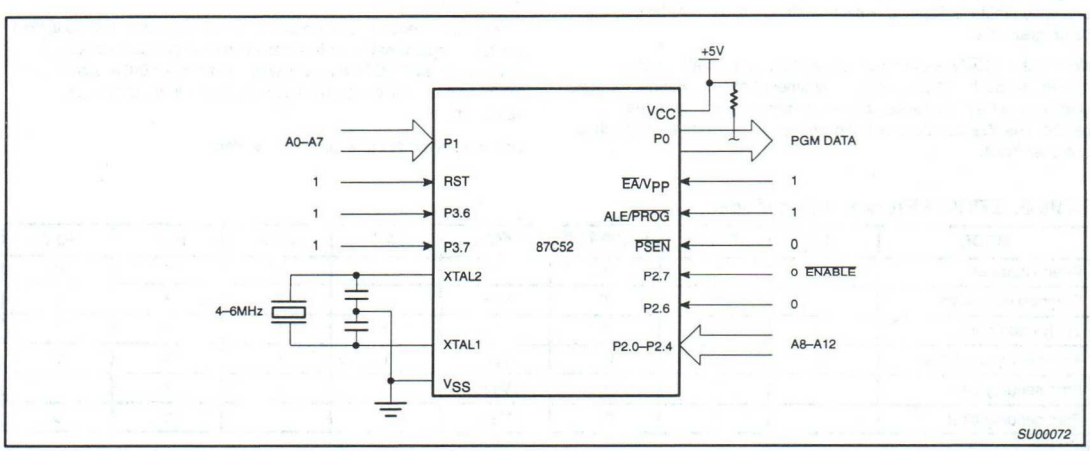
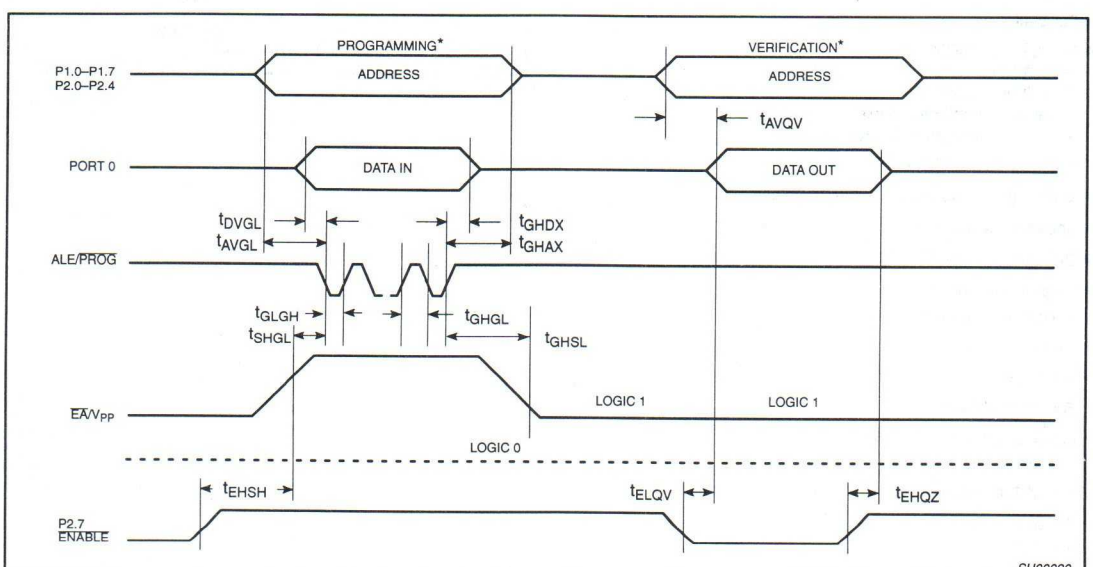



**Figure 19. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (See Figure 20)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




**NOTE:**

* FOR PROGRAMMING VERIFICATION SEE FIGURE 17.
FOR VERIFICATION CONDITIONS SEE FIGURE 19.

**Figure 20. EPROM Programming and Verification**

# CMOS single-chip 8-bit microcontrollers — 80C52/80C54/80C58

## DESCRIPTION

The 80C52/80C54/80C58 Single-Chip 8-Bit Microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 80C52/80C54/80C58 has the same instruction set as the 80C51.

This device provides architectural enhancements that make it applicable in a variety of applications for general control systems. The 80C52 contains 8k × 8 ROM memory, the 80C54 contains 16k × 8 ROM memory, and 80C58 contains 32k × 8 ROM memory, a volatile 256 × 8 read/write data memory, four 8-bit I/O ports, three 16-bit timer/event counters, a multi-source, four-priority-level, nested interrupt structure, an enhanced UART and on-chip oscillator and timing circuits. For systems that require extra capability, the 80C52/54/58 can be expanded using standard TTL compatible memories and logic.

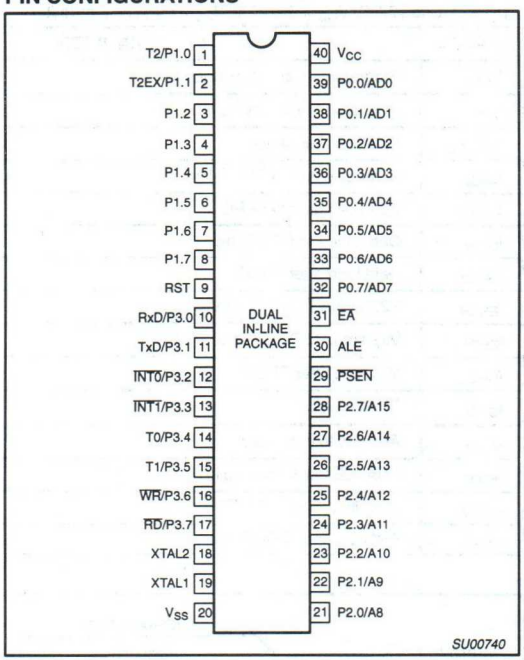
Its added features make it an even more powerful microcontroller for applications that require pulse width modulation, high-speed I/O and up/down counting capabilities such as motor control. It also has a more versatile serial channel that facilitates multiprocessor communications.

See 87C52/80C32 and 87C54/87C58 data sheets for EPROM and ROMless devices.

## FEATURES

- 80C51 central processing unit
- Full static operation
- 8k × 8 ROM: 80C52;
  16k × 8 ROM: 80C54;
  32k × 8 ROM: 80C58;
  all capable of addressing external memory to 64k bytes
  - Two level program security system
  - 64 byte encryption array
- 256 × 8 RAM, expandable externally to 64k bytes
- Speed range up to 33MHz
- Operating voltage 5V ±10%
- Three 16-bit timer/counters
  - T2 is an up/down counter
- 6 interrupt sources
- 4 level priority
- Four 8-bit I/O ports
- Full-duplex enhanced UART
  - Framing error detection
  - Automatic address recognition
- Power control modes
  - Idle mode
  - Power-down mode
- Once (On Circuit Emulation) Mode
- Five package styles
- Programmable clock out
- Low EMI (Inhibit ALE)
- Second DPTR register
- Asynchronous port reset

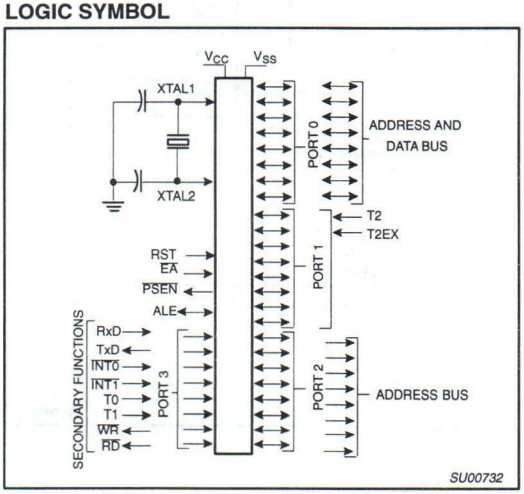
## PIN CONFIGURATIONS



## ORDERING INFORMATION

| ROM 8k × 8 | ROM 16k × 8 | ROM 32k × 8 | TEMPERATURE RANGE °C AND PACKAGE | FREQ MHz | DRAWING NUMBER |
|---|---|---|---|---|---|
| P80C52EBPN | P80C54EBPN | P80C58EBPN | 0 to +70, Plastic Dual In-line Package | 16 | SOT129-1 |
| P80C52EBAA | P80C54EBAA | P80C58EBAA | 0 to +70, Plastic Leaded Chip Carrier | 16 | SOT187-2 |
| P80C52EBBB | P80C54EBBB | P80C58EBBB | 0 to +70, Plastic Quad Flat Pack | 16 | SOT307-2 |
| P80C52EFPN | P80C54EFPN | P80C58EFPN | –40 to +85, Plastic Dual In-line Package | 16 | SOT129-1 |
| P80C52EFAA | P80C54EFAA | P80C58EFAA | –40 to +85, Plastic Leaded Chip Carrier | 16 | SOT187-2 |
| P80C52EFBB | P80C54EFBB | P80C58EFBB | –40 to +85, Plastic Quad Flat Pack | 16 | SOT307-2 |
| P80C52IBP N | P80C54IBP N | P80C58IBP N | 0 to +70, Plastic Dual In-line Package | 24 | SOT129-1 |
| P80C52IBA A | P80C54IBA A | P80C58IBA A | 0 to +70, Plastic Leaded Chip Carrier | 24 | SOT187-2 |
| P80C52IBB B | P80C54IBB B | P80C58IBB B | 0 to +70, Plastic Quad Flat Pack | 24 | SOT307-2 |
| P80C52IFP N | P80C54IFP N | P80C58IFP N | –40 to +85, Plastic Dual In-line Package | 24 | SOT129-1 |
| P80C52IFA A | P80C54IFA A | P80C58IFA A | –40 to +85, Plastic Leaded Chip Carrier | 24 | SOT187-2 |
| P80C52IFB B | P80C54IFB B | P80C58IFB B | –40 to +85, Plastic Quad Flat Pack | 24 | SOT307-2 |
| P80C52NBAA | P80C54NBAA | P80C58NBAA | 0 to +70, Plastic Leaded Chip Carrier | 33 | SOT187-2 |
| P80C52NBPN | P80C54NBPN | P80C58NBPN | 0 to +70, Plastic Dual In-line Package | 33 | SOT129-1 |
| P80C52NBBB | P80C54NBBB | P80C58NBBB | 0 to +70, Plastic Quad Flat Pack | 33 | SOT307-2 |
| P80C52NFAA | P80C54NFAA | P80C58NFAA | –40 to +85, Plastic Leaded Chip Carrier | 33 | SOT187-2 |
| P80C52NFPN | P80C54NFPN | P80C58NFPN | –40 to +85, Plastic Dual In-line Package | 33 | SOT129-1 |
| P80C52NFBB | P80C54NFBB | P80C58NFBB | –40 to +85, Plastic Quad Flat Pack | 33 | SOT307-2 |

## LOGIC SYMBOL



## BLOCK DIAGRAM

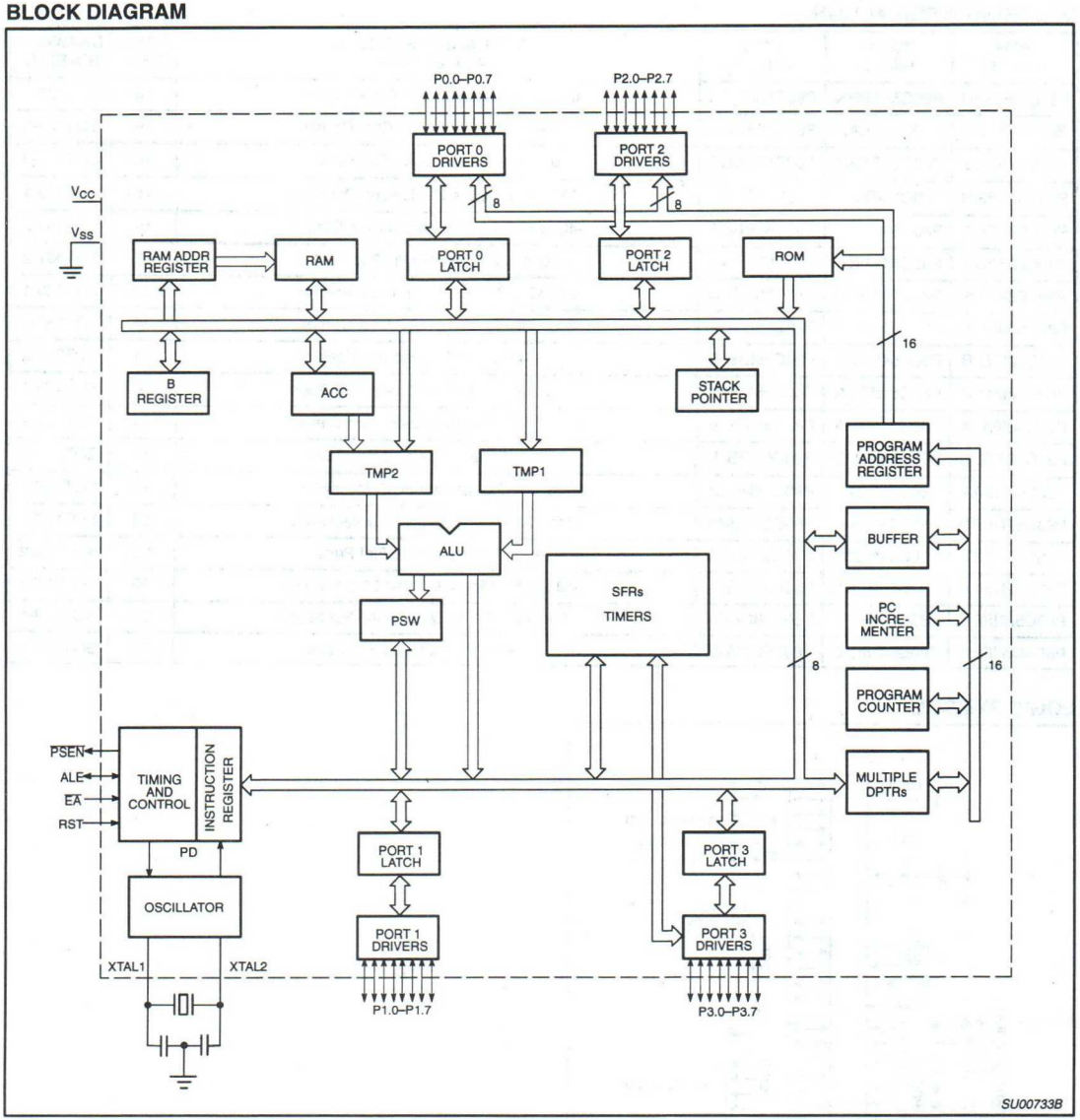
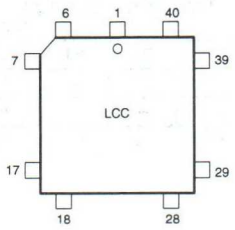


**Table 1. 80C52/80C54/80C58 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| AUXR# | Auxiliary | 8EH | – | – | – | – | – | – | – | AO | xxxxxxx0B |
| AUXR1# | Auxiliary 1 | A2H | – | – | – | – | – | – | – | DPS | xxxxxxx0B |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data Pointer (2 bytes) | | | | | | | | | | |
| DPH | Data Pointer High | 83H | | | | | | | | | 00H |
| DPL | Data Pointer Low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE* | Interrupt Enable | A8H | EA | EC | ET2 | ES | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP* | Interrupt Priority | B8H | – | – | PT2 | PS | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| IPH# | Interrupt Priority High | B7H | – | – | PT2H | PSH | PT1H | PX1H | PT0H | PX0H | x0000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | – | – | – | – | – | – | T2EX | T2 | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | AD15 | AD14 | AD13 | AD12 | AD11 | AD10 | AD9 | AD8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TxD | RxD | FFH |
| PCON#[1] | Power Control | 87H | SMOD1 | SMOD0 | – | POF[2] | GF1 | GF0 | PD | IDL | 00xx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program Status Word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| RACAP2H# | Timer 2 Capture High | CBH | | | | | | | | | 00H |
| RACAP2L# | Timer 2 Capture Low | CAH | | | | | | | | | 00H |
| SADDR# | Slave Address | A9H | | | | | | | | | 00H |
| SADEN# | Slave Address Mask | B9H | | | | | | | | | 00H |
| SBUF | Serial Data Buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial Control | 98H | SM0/FE | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack Pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer Control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| T2CON* | Timer 2 Control | C8H | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | $C/\overline{T2}$ | $CP/\overline{RL2}$ | 00H |
| T2MOD# | Timer 2 Mode Control | C9H | – | – | – | – | – | – | T2OE | DCEN | xxxxxx00B |
| TH0 | Timer High 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer High 1 | 8DH | | | | | | | | | 00H |
| TH2# | Timer High 2 | CDH | | | | | | | | | 00H |
| TL0 | Timer Low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer Low 1 | 8BH | | | | | | | | | 00H |
| TL2# | Timer Low 2 | CCH | | | | | | | | | 00H |
| TMOD | Timer Mode | 89H | GATE | $C/\overline{T}$ | M1 | M0 | GATE | $C/\overline{T}$ | M1 | M0 | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
– Reserved bits.
1. Reset value depends on reset source.
2. Bit will not be affected by Reset. POF is not present in 80C52.

## PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

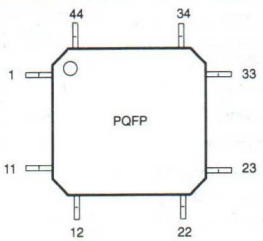



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | NC* | 16 | P3.4/T0 | 31 | P2.7/A15 |
| 2 | P1.0/T2 | 17 | P3.5/T1 | 32 | $\overline{PSEN}$ |
| 3 | P1.1/T2EX | 18 | P3.6/$\overline{WR}$ | 33 | ALE |
| 4 | P1.2 | 19 | P3.7/$\overline{RD}$ | 34 | NC* |
| 5 | P1.3 | 20 | XTAL2 | 35 | $\overline{EA}$ |
| 6 | P1.4 | 21 | XTAL1 | 36 | P0.7/AD7 |
| 7 | P1.5 | 22 | $V_{SS}$ | 37 | P0.6/AD6 |
| 8 | P1.6 | 23 | NC* | 38 | P0.5/AD5 |
| 9 | P1.7 | 24 | P2.0/A8 | 39 | P0.4/AD4 |
| 10 | RST | 25 | P2.1/A9 | 40 | P0.3/AD3 |
| 11 | P3.0/RxD | 26 | P2.2/A10 | 41 | P0.2/AD2 |
| 12 | NC* | 27 | P2.3/A11 | 42 | P0.1/AD1 |
| 13 | P3.1/TxD | 28 | P2.4/A12 | 43 | P0.0/AD0 |
| 14 | P3.2/$\overline{INT0}$ | 29 | P2.5/A13 | 44 | $V_{CC}$ |
| 15 | P3.3/$\overline{INT1}$ | 30 | P2.6/A14 | | |

* DO NOT CONNECT

SU00741A

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS




| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | P1.5 | 16 | $V_{SS}$ | 31 | P0.6/AD6 |
| 2 | P1.6 | 17 | NC* | 32 | P0.5/AD5 |
| 3 | P1.7 | 18 | P2.0/A8 | 33 | P0.4/AD4 |
| 4 | RST | 19 | P2.1/A9 | 34 | P0.3/AD3 |
| 5 | P3.0/RxD | 20 | P2.2/A10 | 35 | P0.2/AD2 |
| 6 | NC* | 21 | P2.3/A11 | 36 | P0.1/AD1 |
| 7 | P3.1/TxD | 22 | P2.4/A12 | 37 | P0.0/AD0 |
| 8 | P3.2/$\overline{INT0}$ | 23 | P2.5/A13 | 38 | $V_{CC}$ |
| 9 | P3.3/$\overline{INT1}$ | 24 | P2.6/A14 | 39 | NC* |
| 10 | P3.4/T0 | 25 | P2.7/A15 | 40 | P1.0/T2 |
| 11 | P3.5/T1 | 26 | $\overline{PSEN}$ | 41 | P1.1/T2EX |
| 12 | P3.6/$\overline{WR}$ | 27 | ALE | 42 | P1.2 |
| 13 | P3.7/$\overline{RD}$ | 28 | NC* | 43 | P1.3 |
| 14 | XTAL2 | 29 | $\overline{EA}$ | 44 | P1.4 |
| 15 | XTAL1 | 30 | P0.7/AD7 | | |

* DO NOT CONNECT

SU00742A

## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER DIP | PIN NUMBER LCC | PIN NUMBER QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification. External pull-ups are required during program verification. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also receives the low-order address byte during program memory verification. Alternate functions include: |
| | 1 | 2 | 40 | I/O | **T2 (P1.0):** Timer/Counter 2 external count input/Clockout |
| | 2 | 3 | 41 | I | **T2EX (P1.1):** Timer/Counter 2 Reload/Capture/Direction Control |
| | 3 | 4 | 42 | I | |
| | 4 | 5 | 43 | I/O | |
| | 5 | 6 | 44 | I/O | |
| | 6 | 7 | 1 | I/O | |
| | 7 | 8 | 2 | I/O | |
| | 8 | 9 | 3 | I/O | |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |

## PIN DESCRIPTIONS (Continued)

| MNEMONIC | PIN NUMBER DIP | PIN NUMBER LCC | PIN NUMBER QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{\text{INT0}}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{\text{INT1}}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{\text{WR}}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{\text{RD}}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{CC}$. |
| ALE | 30 | 33 | 27 | O | **Address Latch Enable:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. ALE can be disabled by setting SFR auxiliary.0. With this bit set, ALE will be active only during a MOVX instruction. |
| $\overline{\text{PSEN}}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the 80C52/80C54/80C58 is executing code from the external program memory, $\overline{\text{PSEN}}$ is activated twice each machine cycle, except that two $\overline{\text{PSEN}}$ activations are skipped during each access to external data memory. $\overline{\text{PSEN}}$ is not activated during fetches from internal program memory. |
| $\overline{\text{EA}}$ | 31 | 35 | 29 | I | **External Access Enable:** $\overline{\text{EA}}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H and 7FFFH. If $\overline{\text{EA}}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 7FFFH. If security bit 1 is programmed, $\overline{\text{EA}}$ will be internally latched on Reset. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTE:**
To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher than $V_{CC}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

## TIMER 2 OPERATION

### Timer 2

Timer 2 is a 16-bit Timer/Counter which can operate as either an event timer or an event counter, as selected by C/T2* in the special function register T2CON (see Figure 1). Timer 2 has three operating modes:Capture, Auto-reload (up or down counting) ,and Baud Rate Generator, which are selected by bits in the T2CON as shown in Table 2.

### Capture Mode

In the capture mode there are two options which are selected by bit EXEN2 in T2CON. If EXEN2=0, then timer 2 is a 16-bit timer or counter (as selected by C/T2* in T2CON) which, upon overflowing sets bit TF2, the timer 2 overflow bit. This bit can be used to generate an interrupt (by enabling the Timer 2 interrupt bit in the IE register/SFR table). If EXEN2= 1, Timer 2 operates as described above, but with the added feature that a 1- to -0 transition at external input T2EX causes the current value in the Timer 2 registers, TL2 and TH2, to be captured into registers RCAP2L and RCAP2H, respectively. In addition, the transition at T2EX causes bit EXF2 in T2CON to be set, and EXF2 like TF2 can generate an interrupt (which vectors to the same location as Timer 2 overflow interrupt. The Timer 2 interrupt service routine can interrogate TF2 and EXF2 to determine which event caused the interrupt). The capture mode is illustrated in Figure 2 (There is no reload value for TL2 and TH2 in this mode. Even when a capture event occurs from T2EX, the counter keeps on counting T2EX pin transitions or osc/12 pulses.).

### Auto-Reload Mode (Up or Down Counter)

In the 16-bit auto-reload mode, Timer 2 can be configured (as either a timer or counter (C/T2* in T2CON)) then programmed to count up or down. The counting direction is determined by bit DCEN(Down Counter Enable) which is located in the T2MOD register (see Figure 3). When reset is applied the DCEN=0 which means Timer 2 will default to counting up. If DCEN bit is set, Timer 2 can count up or down depending on the value of the T2EX pin.

Figure 4 shows Timer 2 which will count up automatically since DCEN=0. In this mode there are two options selected by bit EXEN2 in T2CON register. If EXEN2=0, then Timer 2 counts up to 0FFFFH and sets the TF2 (Overflow Flag) bit upon overflow. This causes the Timer 2 registers to be reloaded with the 16-bit value in RCAP2L and RCAP2H.

The values in RCAP2L and RCAP2H are preset by software means. If EXEN2=1, then a 16-bit reload can be triggered either by an overflow or by a 1-to-0 transition at input T2EX. This transition also sets the EXF2 bit. The Timer 2 interrupt, if enabled, can be generated when either TF2 or EXF2 are 1.

In Figure 5 DCEN=1 which enables Timer 2 to count up or down. This mode allows pin T2EX to control the direction of count. When a logic 1 is applied at pin T2EX Timer 2 will count up. Timer 2 will overflow at 0FFFFH and set the TF2 flag, which can then generate an interrupt, if the interrupt is enabled. This timer overflow also causes the 16–bit value in RCAP2L and RCAP2H to be reloaded into the timer registers TL2 and TH2.

When a logic 0 is applied at pin T2EX this causes Timer 2 to count down. The timer will underflow when TL2 and TH2 become equal to the value stored in RCAP2L and RCAP2H. Timer 2 underflow sets the TF2 flag and causes 0FFFFH to be reloaded into the timer registers TL2 and TH2.

The external flag EXF2 toggles when Timer 2 underflows or overflows. This EXF2 bit can be used as a 17th bit of resolution if needed. The EXF2 flag does not generate an interrupt in this mode of operation.

(MSB) (LSB)

| TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/$\overline{T2}$ | CP/$\overline{RL2}$ |
|---|---|---|---|---|---|---|---|

| Symbol | Position | Name and Significance |
|---|---|---|
| TF2 | T2CON.7 | Timer 2 overflow flag set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK or TCLK = 1. |
| EXF2 | T2CON.6 | Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software. EXF2 does not cause an interrupt in up/down counter mode (DCEN = 1). |
| RCLK | T2CON.5 | Receive clock flag. When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in modes 1 and 3. RCLK = 0 causes Timer 1 overflow to be used for the receive clock. |
| TCLK | T2CON.4 | Transmit clock flag. When set, causes the serial port to use Timer 2 overflow pulses for its transmit clock in modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock. |
| EXEN2 | T2CON.3 | Timer 2 external enable flag. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX. |
| TR2 | T2CON.2 | Start/stop control for Timer 2. A logic 1 starts the timer. |
| C/$\overline{T2}$ | T2CON.1 | Timer or counter select. (Timer 2)<br>0 = Internal timer (OSC/12)<br>1 = External event counter (falling edge triggered). |
| CP/$\overline{RL2}$ | T2CON.0 | Capture/Reload flag. When set, captures will occur on negative transitions at T2EX if EXEN2 = 1. When cleared, auto-reloads will occur either with Timer 2 overflows or negative transitions at T2EX when EXEN2 = 1. When either RCLK = 1 or TCLK = 1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow. |

SU00728

**Figure 1. Timer/Counter 2 (T2CON) Control Register**

**Table 2. Timer 2 Operating Modes**

| RCLK + TCLK | CP/$\overline{RL2}$ | TR2 | MODE |
|---|---|---|---|
| 0 | 0 | 1 | 16-bit Auto-reload |
| 0 | 1 | 1 | 16-bit Capture |
| 1 | X | 1 | Baud rate generator |
| X | X | 0 | (off) |

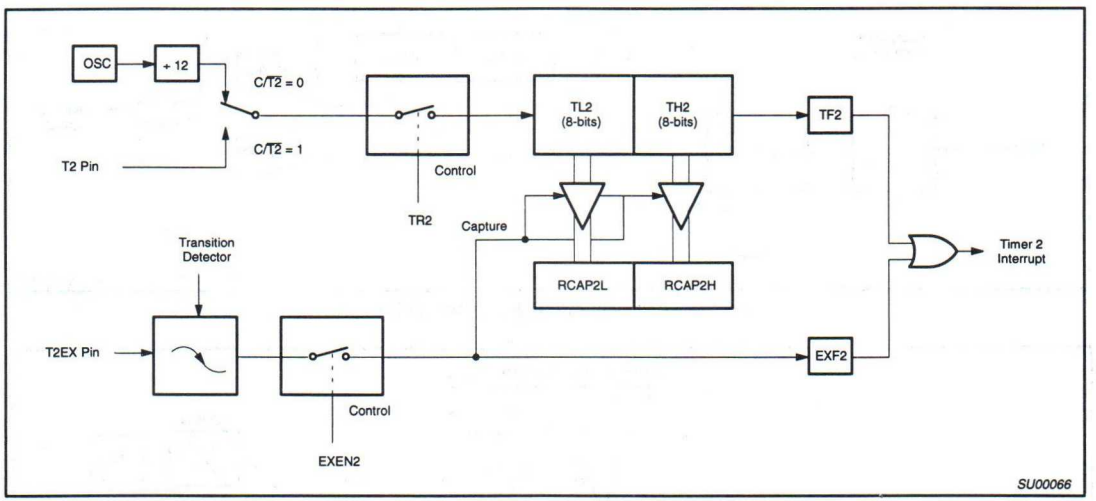



**Figure 2. Timer 2 in Capture Mode**

**T2MOD** Address = 0C9H Reset Value = XXXX XX00B

Not Bit Addressable

| — | — | — | — | — | — | T2OE | DCEN |
|---|---|---|---|---|---|---|---|
| Bit 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Symbol | Function |
|---|---|
| — | Not implemented, reserved for future use.* |
| T2OE | Timer 2 Output Enable bit. See details in Programmable Clock-Out. |
| DCEN | Down Count Enable bit. When set, this allows Timer 2 to be configured as an up/down counter. |

* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

SU00746

**Figure 3. Timer 2 Mode (T2MOD) Control Register**

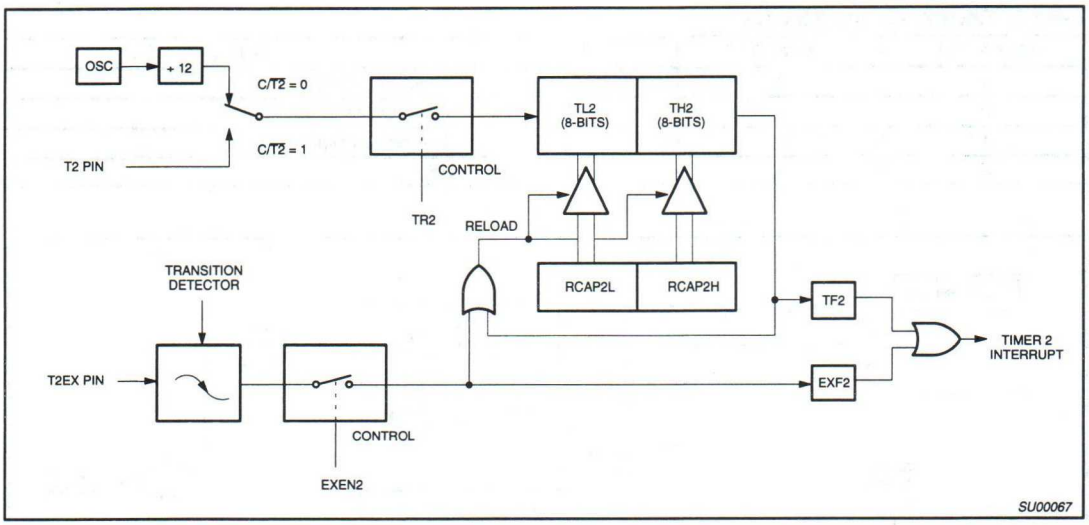



**Figure 4. Timer 2 in Auto-Reload Mode (DCEN = 0)**

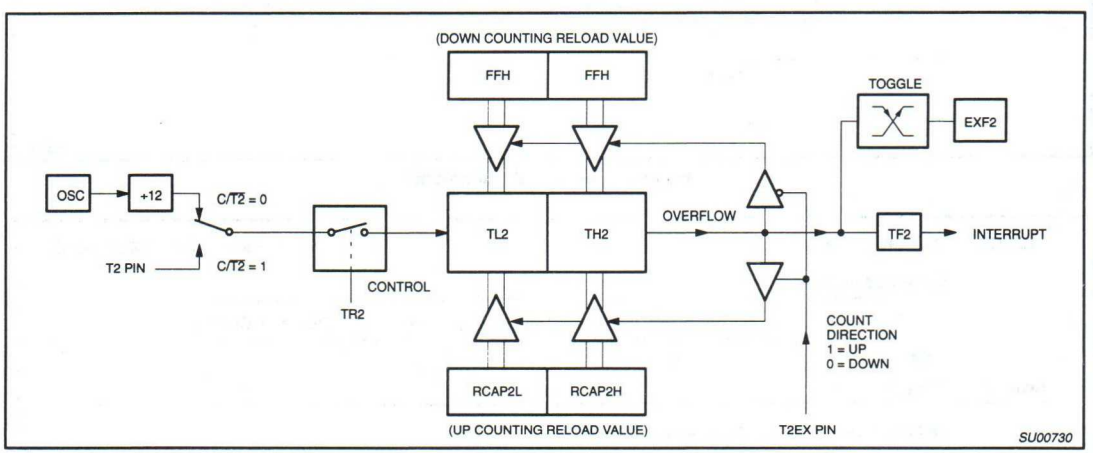



**Figure 5. Timer 2 Auto Reload Mode (DCEN = 1)**

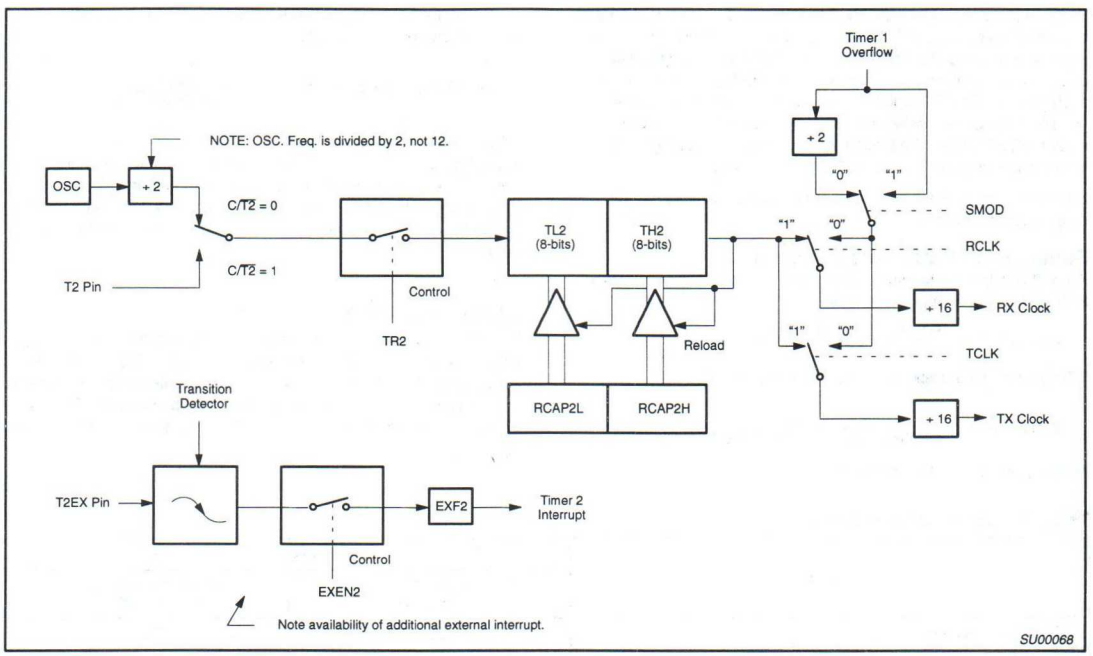



**Figure 6. Timer 2 in Baud Rate Generator Mode**

## Table 3. Timer 2 Generated Commonly Used Baud Rates

| Baud Rate | Osc Freq | Timer 2 | |
|---|---|---|---|
| | | RCAP2H | RCAP2L |
| 375K | 12MHz | FF | FF |
| 9.6K | 12MHz | FF | D9 |
| 2.8K | 12MHz | FF | B2 |
| 2.4K | 12MHz | FF | 64 |
| 1.2K | 12MHz | FE | C8 |
| 300 | 12MHz | FB | 1E |
| 110 | 12MHz | F2 | AF |
| 300 | 6MHz | FD | 8F |
| 110 | 6MHz | F9 | 57 |

## Baud Rate Generator Mode

Bits TCLK and/or RCLK in T2CON (Table 2) allow the serial port transmit and receive baud rates to be derived from either Timer 1 or Timer 2. When TCLK= 0, Timer 1 is used as the serial port transmit baud rate generator. When TCLK= 1, Timer 2 is used as the serial port transmit baud rate generator. RCLK has the same effect for the serial port receive baud rate. With these two bits, the serial port can have different receive and transmit baud rates – one generated by Timer 1, the other by Timer 2.

Figure 6 shows the Timer 2 in baud rate generation mode. The baud rate generation mode is like the auto-reload mode,in that a rollover in TH2 causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2H and RCAP2L, which are preset by software.

The baud rates in modes 1 and 3 are determined by Timer 2's overflow rate given below:

$$\text{Modes 1 and 3 Baud Rates} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

The timer can be configured for either "timer" or "counter" operation. In many applications, it is configured for "timer" operation (C/T2*=0). Timer operation is different for Timer 2 when it is being used as a baud rate generator.

Usually, as a timer it would increment every machine cycle (i.e., 1/12 the oscillator frequency). As a baud rate generator, it increments every state time (i.e., 1/2 the oscillator frequency). Thus the baud rate formula is as follows:

$$\text{Modes 1 and 3 Baud Rates} = \frac{\text{Oscillator Frequency}}{[32 \times [65536 - (\text{RCAP2H, RCAP2L})]]}$$

Where: (RCAP2H, RCAP2L)= The content of RCAP2H and RCAP2L taken as a 16-bit unsigned integer.

The Timer 2 as a baud rate generator mode shown in Figure 6, is valid only if RCLK and/or TCLK = 1 in T2CON register. Note that a rollover in TH2 does not set TF2, and will not generate an interrupt. Thus, the Timer 2 interrupt does not have to be disabled when Timer 2 is in the baud rate generator mode. Also if the EXEN2 (T2 external enable flag) is set, a 1-to-0 transition in T2EX (Timer/counter 2 trigger input) will set EXF2 (T2 external flag) but will not cause a reload from (RCAP2H, RCAP2L) to (TH2,TL2). Therefore when Timer 2 is in use as a baud rate generator, T2EX can be used as an additional external interrupt, if needed.

When Timer 2 is in the baud rate generator mode, one should not try to read or write TH2 and TL2. As a baud rate generator, Timer 2 is incremented every state time (osc/2) or asynchronously from pin T2; under these conditions, a read or write of TH2 or TL2 may not be accurate. The RCAP2 registers may be read, but should not be written to, because a write might overlap a reload and cause write and/or reload errors. The timer should be turned off (clear TR2) before accessing the Timer 2 or RCAP2 registers.

Table 3 shows commonly used baud rates and how they can be obtained from Timer 2.

## Summary Of Baud Rate Equations

Timer 2 is in baud rate generating mode. If Timer 2 is being clocked through pin T2(P1.0) the baud rate is:

$$\text{Baud Rate} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

If Timer 2 is being clocked internally, the baud rate is:

$$\text{Baud Rate} = \frac{f_{OSC}}{[32 \times [65536 - (\text{RCAP2H, RCAP2L})]]}$$

Where $f_{OSC}$= Oscillator Frequency

To obtain the reload value for RCAP2H and RCAP2L, the above equation can be rewritten as:

$$\text{RCAP2H, RCAP2L} = 65536 - \left(\frac{f_{OSC}}{32 \times \text{Baud Rate}}\right)$$

## Timer/Counter 2 Set-up

Except for the baud rate generator mode, the values given for T2CON do not include the setting of the TR2 bit. Therefore, bit TR2 must be set, separately, to turn the timer on. see Table 4 for set-up of Timer 2 as a timer. Also see Table 5 for set-up of Timer 2 as a counter.

## POWER OFF FLAG[3]

The Power Off Flag (POF) is set by on-chip circuitry when the $V_{CC}$ level on the 80C54/80C58 rises from 0 to 5V. The POF bit can be set or cleared by software allowing a user to determine if the reset is the result of a power-on or a warm start after powerdown. The $V_{CC}$ level must remain above 3V for the POF to remain unaffected by the $V_{CC}$ level.

**Table 4. Timer 2 as a Timer**

| MODE | T2CON | |
|---|---|---|
| | INTERNAL CONTROL (Note 1) | EXTERNAL CONTROL (Note 2) |
| 16-bit Auto-Reload | 00H | 08H |
| 16-bit Capture | 01H | 09H |
| Baud rate generator receive and transmit same baud rate | 34H | 36H |
| Receive only | 24H | 26H |
| Transmit only | 14H | 16H |

**Table 5. Timer 2 as a Counter**

| MODE | TMOD | |
|---|---|---|
| | INTERNAL CONTROL (Note 1) | EXTERNAL CONTROL (Note 2) |
| 16-bit | 02H | 0AH |
| Auto-Reload | 03H | 0BH |

**NOTES:**
1. Capture/reload occurs only on timer/counter overflow.
2. Capture/reload occurs on timer/counter overflow and a 1-to-0 transition on T2EX (P1.1) pin except when Timer 2 is used in the baud rate generator mode.
3. POF not present in 80C52.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

### Reset

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up. Ports 1, 2, and 3 will asynchronously be driven to their reset condition when a voltage above $V_{IH1}$ (min.) is applied to RESET.

### Idle Mode

In the idle mode (see Table 6), the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

### Power-Down Mode

To save even more power, a Power Down mode (see Table 6) can be invoked by software. In this mode, the oscillator is stopped and the instruction that invoked Power Down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the Power Down mode is terminated.

On the 80C52/54/58 either a hardware reset or external interrupt can be used to exit from Power Down. Reset redefines all the SFRs but does not change the on-chip RAM. An external interrupt allows both the SFRs and the on-chip RAM to retain their values.

To properly terminate Power Down the reset or external interrupt should not be executed before $V_{CC}$ is restored to its normal operating level and must be held active long enough for the oscillator to restart and stabilize (normally less than 10ms).

With an external interrupt, INT0 and INT1 must be enabled and configured as level-sensitive. Holding the pin low restarts the oscillator but bringing the pin back high completes the exit. Once the interrupt is serviced, the next instruction to be executed after RETI will be the one following the instruction that put the device into Power Down.

### Design Consideration

- When the idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal rest algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

### ONCE™ Mode

The ONCE ("On-Circuit Emulation") Mode facilitates testing and debugging of systems using the 80C52/54/58 without removing the device from the circuit. The ONCE Mode is invoked by:

1. Pull ALE low while the device is in reset and $\overline{PSEN}$ is high;
2. Hold ALE low as RST is deactivated.

While the device is in ONCE Mode, the Port 0 pins go into a float state, and the other port pins and ALE and $\overline{PSEN}$ are weakly pulled high. The oscillator circuit remains active. While the 80C52/54/58 is in this mode, an emulator or test CPU can be used to drive the circuit. Normal operation is restored when a normal reset is applied.

### Programmable Clock-Out

The 80C52/54/58 has a new feature. A 50% duty cycle clock can be programmed to come out on P1.0. This pin, besides being a regular I/O pin, has two alternate functions. It can be programmed:

1. to input the external clock for Timer/Counter 2, or
2. to output a 50% duty cycle clock ranging from 61Hz to 4MHz at a 16MHz operating frequency.

To configure the Timer/Counter 2 as a clock generator, bit C/T2 (in T2CON) must be cleared and bit T2OE in T2MOD must be set. Bit TR2 (T2CON.2) also must be set to start the timer.

The Clock-Out frequency depends on the oscillator frequency and the reload value of Timer 2 capture registers (RCAP2H, RCAP2L) as shown in this equation:

$$\frac{\text{Oscillator Frequency}}{4 \times (65536 - \text{RCAP2H,RCAP2L})}$$

In the Clock-Out mode Timer 2 roll-overs will not generate an interrupt. This is similar to when it is used as a baud-rate generator. It is possible to use Timer 2 as a baud-rate generator and a clock generator simultaneously. Note, however, that the baud-rate and the Clock-Out frequency will be the same.

**Table 6. External Pin Status During Idle and Power-Down Mode**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

## Enhanced UART

The UART operates in all of the usual modes that are described in the first section of *Data Handbook IC20, 80C51-Based 8-Bit Microcontrollers*. In addition the UART can perform framing error detect by looking for missing stop bits, and automatic address recognition. The 80C52/54/58 UART also fully supports multiprocessor communication as does the standard 80C51 UART.

When used for framing error detect the UART looks for missing stop bits in the communication. A missing bit will set the FE bit in the SCON register. The FE bit shares the SCON.7 bit with SM0 and the function of SCON.7 is determined by PCON.6 (SMOD0) (see Figure 7). If SMOD0 is set then SCON.7 functions as FE. SCON.7 functions as SM0 when SMOD0 is cleared. When used as FE SCON.7 can only be cleared by software. Refer to Figure 8.

### Automatic Address Recognition

Automatic Address Recognition is a feature which allows the UART to recognize certain addresses in the serial bit stream by using hardware to make the comparisons. This feature saves a great deal of software overhead by eliminating the need for the software to examine every serial address which passes by the serial port. This feature is enabled by setting the SM2 bit in SCON. In the 9 bit UART modes, mode 2 and mode 3, the Receive Interrupt flag (RI) will be automatically set when the received byte contains either the "Given" address or the "Broadcast" address. The 9 bit mode requires that the 9th information bit is a 1 to indicate that the received information is an address and not data. Automatic address recognition is shown in Figure 9.

The 8 bit mode is called Mode 1. In this mode the RI flag will be set if SM2 is enabled and the information received has a valid stop bit following the 8 address bits and the information is either a Given or Broadcast address.

Mode 0 is the Shift Register mode and SM2 is ignored.

Using the Automatic Address Recognition feature allows a master to selectively communicate with one or more slaves by invoking the Given slave address or addresses. All of the slaves may be contacted by using the Broadcast address. Two special Function Registers are used to define the slave's address, SADDR, and the address mask, SADEN. SADEN is used to define which bits in the SADDR are to b used and which bits are "don't care". The SADEN mask can be logically ANDed with the SADDR to create the "lGiven" address which the master will use for addressing each of the slaves. Use of the Given address allows multiple slaves to be recognized while excluding others. The following examples will help to show the versatility of this scheme:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1101 |
| | Given | = | 1100 00X0 |
| Slave 1 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1110 |
| | Given | = | 1100 000X |

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires a 0 in bit 0 and it ignores bit 1. Slave 1 requires a 0 in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010 since slave 1 requires a 0 in bit 1. A unique address for slave 1 would be 1100 0001 since a 1 in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 1100 0000.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1001 |
| | Given | = | 1100 0XX0 |
| Slave 1 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1010 |
| | Given | = | 1110 0X0X |
| Slave 2 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1100 |
| | Given | = | 1110 00XX |

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 and 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary t make bit 2 = 1 to exclude slave 2.

The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are teated as don't-cares. In most cases, interpreting the don't-cares as ones, the broadcast address will be FF hexadecimal.

Upon reset SADDR (SFR address 0A9H) and SADEN (SFR address 0B9H) are leaded with 0s. This produces a given address of all "don't cares" as well as a Broadcast address of all "don't cares". this effectively disables the Automatic Addressing mode and allows the microcontroller to use standard 80C51 type UART drivers which do not make use of this feature.

**SCON** Address = 98H Reset Value = 0000 0000B

Bit Addressable

| | SM0/FE | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

(SMOD0 = 0/1)*

| Symbol | Function |
|---|---|
| **FE** | Framing Error bit. This bit is set by the receiver when an invalid stop bit is detected. The FE bit is not cleared by valid frames but should be cleared by software. The SMOD0 bit must be set to enable access to the FE bit. |
| **SM0** | Serial Port Mode Bit 0, (SMOD0 must = 0 to access bit SM0) |
| **SM1** | Serial Port Mode Bit 1 |

| SM0 | SM1 | Mode | Description | Baud Rate** |
|---|---|---|---|---|
| 0 | 0 | 0 | shift register | $f_{OSC}/12$ |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{OSC}/64$ or $f_{OSC}/32$ |
| 1 | 1 | 3 | 9-bit UART | variable |

| | |
|---|---|
| **SM2** | Enables the Automatic Address Recognition feature in Modes 2 or 3. If SM2 = 1 then RI will not be set unless the received 9th data bit (RB8) is 1, indicating an address, and the received byte is a Given or Broadcast Address. In Mode 1, if SM2 = 1 then RI will not be activated unless a valid stop bit was received, and the received byte is a Given or Broadcast Address. In Mode 0, SM2 should be 0. |
| **REN** | Enables serial reception. Set by software to enable reception. Clear by software to disable reception. |
| **TB8** | The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired. |
| **RB8** | In modes 2 and 3, the 9th data bit that was received. In Mode 1, if SM2 = 0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used. |
| **TI** | Transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software. |
| **RI** | Receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software. |

**NOTE:**
*SMOD0 is located at PCON6.
**$f_{OSC}$ = oscillator frequency

SU00043

**Figure 7. SCON: Serial Port Control Register**

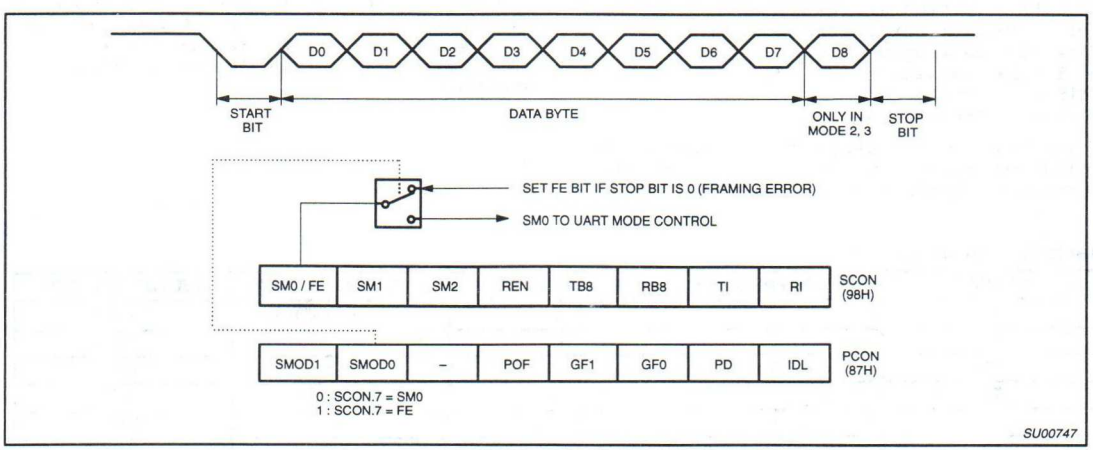



**Figure 8. UART Framing Error Detection**

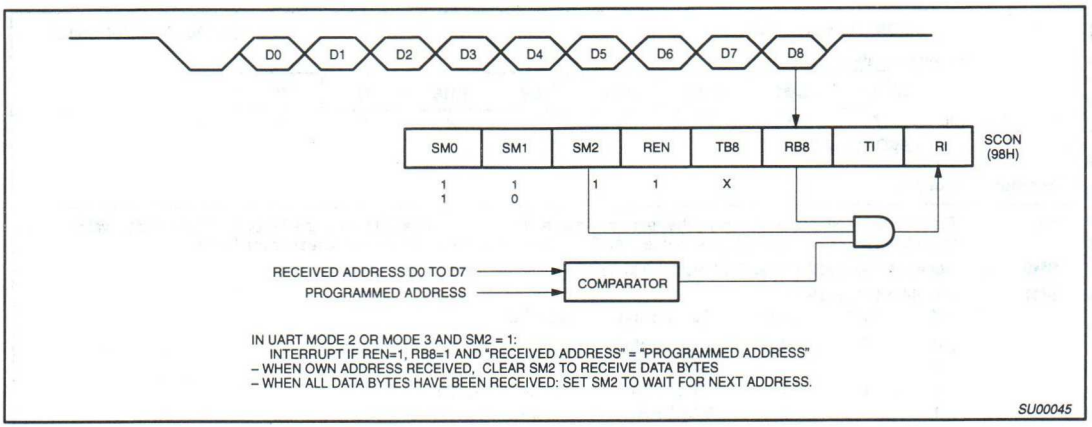



**Figure 9. UART Multiprocessor Communication, Automatic Address Recognition**

## Interrupt Priority Structure

The 80C52/54/58 has a 6-source four-level interrupt structure. There are 3 SFRs associated with the interrupts on the 80C52/54/58. They are the IE and IP. (See Figures 10 and 11.) In addition, there is the IPH (Interrupt Priority High) register that makes the four-level interrupt structure possible. The IPH is located at SFR address B7H. The structure of the IPH register and a description of its bits is shown below:

**IPH (Interrupt Priority High) (B7H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | – | PT2H | PSH | PT1H | PX1H | PT0H | PX0H |

IPH.0 PX0H External interrupt 0 priority high
IPH.1 PT0H Timer 0 interrupt priority high
IPH.2 PX1H External interrupt 1 priority high
IPH.3 PT1H Timer 1 interrupt priority high
IPH.4 PSH Serial Port interrupt high
IPH.5 PT2H Timer 2 interrupt priority high
IPH.6 — Not implemented
IPH.7 — Not implemented

The function of the IPH SFR is simple and when combined with the IP SFR determines the priority of each interrupt. The priority of each interrupt is determined as shown in the following table:

| PRIORITY BITS | | INTERRUPT PRIORITY LEVEL |
|---|---|---|
| IPH.x | IP.x | |
| 0 | 0 | Level 0 (lowest priority) |
| 0 | 1 | Level 1 |
| 1 | 0 | Level 2 |
| 1 | 1 | Level 3 (highest priority) |

The priority scheme for servicing the interrupts is the same as that for the 80C51, except there are four interrupt levels on the 80C52/54/58 rather than two as on the 80C51. An interrupt will be serviced as long as an interrupt of equal or higher priority is not already being serviced. If an interrupt of equal or higher level priority is being serviced, the new interrupt will wait until it is finished before being serviced. If a lower priority level interrupt is being serviced, it will be stopped and the new interrupt serviced. When the new interrupt is finished, the lower priority level interrupt that was stopped will be completed.

**Table 7. Interrupt Table**

| SOURCE | POLLING PRIORITY | REQUEST BITS | HARDWARE CLEAR? | VECTOR ADDRESS |
|---|---|---|---|---|
| X0 | 1 | IE0 | N (L)[1] Y (T)[2] | 03H |
| T0 | 2 | TP0 | Y | 0BH |
| X1 | 3 | IE1 | N (L) Y (T) | 13H |
| T1 | 4 | TF1 | Y | 1BH |
| SP | 5 | R1, TI | N | 23H |
| T2 | 6 | TF2, EXF2 | N | 2BH |
| PCA | 7 | CF, CCFn n = 0–4 | N | 33H |

**NOTES:**
1. L = Level activated
2. T = Transition activated

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| IE0 (A8H) | EA | — | ET2 | ES | ET1 | EX1 | ET0 | EX0 |

Enable Bit = 1 enables the interrupt.
Enable Bit = 0 disables it.

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IE.7 | EA | Global disable bit. If EA = 0, all interrupts are disabled. If EA = 1, each interrupt can be individually enabled or disabled by setting or clearing its enable bit. |
| IE.6 | — | |
| IE.5 | ET2 | Timer 2 interrupt enable bit. |
| IE.4 | ES | Serial Port interrupt enable bit. |
| IE.3 | ET1 | Timer 1 interrupt enable bit. |
| IE.2 | EX1 | External interrupt 1 enable bit. |
| IE.1 | ET0 | Timer 0 interrupt enable bit. |
| IE.0 | EX0 | External interrupt 0 enable bit. |

SU00743

**Figure 10. IE Registers**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| IP0 (B8H) | — | — | PT2 | PS | PT1 | PX1 | PT0 | PX0 |

Priority Bit = 1 assigns high priority
Priority Bit = 0 assigns low priority

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IP.7 | — | Not implemented, reserved for future use. |
| IP.6 | — | |
| IP.5 | PT2 | Timer 2 interrupt priority bit. |
| IP.4 | PS | Serial Port interrupt priority bit. |
| IP.3 | PT1 | Timer 1 interrupt priority bit. |
| IP.2 | PX1 | External interrupt 1 priority bit. |
| IP.1 | PT0 | Timer 0 interrupt priority bit. |
| IP.0 | PX0 | External interrupt 0 priority bit. |

SU00744

**Figure 11. IP Registers**

## Reduced EMI Mode

The AO bit (AUXR.0) in the AUXR register when set disables the ALE output.

## 80C52/80C54/80C58 Reduced EMI Mode

**AUXR (8EH)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | – | – | – | AO |

AO: Turns off ALE output.

## Dual Data Pointer Register (DPTR)

The dual DPTR structure (see Figure 12) is a way by which the 80C52/54/58 will specify the address of an external data memory location. There are two 16-bit DPTR registers that address the external memory, and a single bit called DPS = AUXR1/bit0 that allows the program code to switch between them.

- Register Name: AUXR1#
- SFR Address: A2H
- Reset Value: xxxxxxx0B

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | – | – | – | DPS |

Where:
DPS = AUXR1/bit0 = Switches between DPTR0 and DPTR1.

| Select Reg | DPS |
|---|---|
| DPTR0 | 0 |
| DPTR1 | 1 |

The DPS bit status whould be saved by software when switching between DPTR0 and DPTR1.

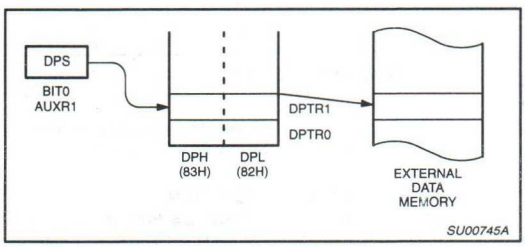



**Figure 12. DPTR Structure**

### DPTR Instructions

The instructions that refer to DPTR refer to the data pointer that is currently selected using the AUXR1/bit 0 register. The six instructions that use the DPTR are as follows:

| | |
|---|---|
| INC DPTR | Increments the data pointer by 1 |
| MOV DPTR, #data16 | Loads the DPTR with a 16-bit constant |
| MOV A, @ A+DPTR | Move code byte relative to DPTR to ACC |
| MOVX A, @ DPTR | Move external RAM (16-bit address) to ACC |
| MOVX @ DPTR , A | Move ACC to external RAM (16-bit address) |
| JMP @ A + DPTR | Jump indirect relative to DPTR |

The data pointer can be accessed on a byte-by-byte basis by specifying the Low or High byte in an instruction which accesses the SFRs. See application note AN458 for detailed operation

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | 0 to +70 or –40 to +85 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Maximum $I_{OL}$ per I/O pin | 15 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.5 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5.0V ±10%; $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $V_{IL}$ | Input low voltage | 4.5V < $V_{CC}$ < 5.5V | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IH}$ | Input high voltage (ports 0, 1, 2, 3, $\overline{EA}$) | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3[8] | $V_{CC}$ = 4.5V<br>$I_{OL}$ = 1.6mA[2] | | | 0.4 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$[8, 7] | $V_{CC}$ = 4.5V<br>$I_{OL}$ = 3.2mA[2] | | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3 [3] | $V_{CC}$ = 4.5V<br>$I_{OH}$ = –30µA | $V_{CC}$ – 0.7 | | | V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode), ALE[9], $\overline{PSEN}$[3] | $V_{CC}$ = 4.5V<br>$I_{OH}$ = –3.2mA | $V_{CC}$ – 0.7 | | | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3 | $V_{IN}$ = 0.4V | –1 | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3[6] | $V_{IN}$ = 2.0V<br>See note 4 | | | –650 | µA |
| $I_{LI}$ | Input leakage current, port 0 | 0.45 < $V_{IN}$ < $V_{CC}$ – 0.3 | | | ±10 | µA |
| $I_{CC}$ | Power supply current (see Figure 20):<br>Active mode @ 16MHz[5]<br>Idle mode @ 16MHz[5]<br>Power-down mode | See note 5<br><br><br>$T_{amb}$ = 0 to +70°C<br>$T_{amb}$ = –40 to +85°C | | <br><br><br>3 | <br>16<br>4<br>50<br>75 | <br>mA<br>mA<br>µA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | 40 | | 225 | kΩ |
| $C_{IO}$ | Pin capacitance[10] (except $\overline{EA}$) | | | | 15 | pF |

**NOTES:**

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the ($V_{CC}$–0.7) specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
5. See Figures 21 through 24 for $I_{CC}$ test conditions.
   Active Mode: $I_{CC} = 0.9 \times FREQ + 1.1$;
   Idle Mode: $I_{CC} = 0.18 \times FREQ + 1.0$; See Figure 20.
6. This value applies to $T_{amb}$ = 0°C to +70°C. For $T_{amb}$ = –40°C to +85°C, $I_{TL}$ = –750µA.
7. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
8. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 15mA (*NOTE: This is 85°C specification.)
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 71mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
9. ALE is tested to $V_{OH1}$, except when ALE is off then $V_{OH}$ is the voltage specification.
10. Pin capacitance is characterized but not tested. Pin capacitance is less than 25pF. Pin capacitance of ceramic package is less than 15pF (except $\overline{EA}$ it is 25pF).

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 13 | Oscillator frequency<br>Speed versions : E | | | 3.5 | 16 | MHz |
| $t_{LHLL}$ | 13 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 13 | Address valid to ALE low | 22 | | $t_{CLCL}-40$ | | ns |
| $t_{LLAX}$ | 13 | Address hold after ALE low | 32 | | $t_{CLCL}-30$ | | ns |
| $t_{LLIV}$ | 13 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 13 | ALE low to $\overline{PSEN}$ low | 32 | | $t_{CLCL}-30$ | | ns |
| $t_{PLPH}$ | 13 | $\overline{PSEN}$ pulse width | 142 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 13 | $\overline{PSEN}$ low to valid instruction in[4] | | 82 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 13 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 13 | Input instruction float after $\overline{PSEN}$ | | 37 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 13 | Address to valid instruction in[4] | | 207 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 13 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 14, 15 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 14, 15 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 14, 15 | $\overline{RD}$ low to valid data in | | 147 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 14, 15 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 14, 15 | Data float after $\overline{RD}$ | | 65 | | $2t_{CLCL}-60$ | ns |
| $t_{LLDV}$ | 14, 15 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 14, 15 | Address to valid data in | | 397 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 14, 15 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 137 | 239 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 14, 15 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 122 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 14, 15 | Data valid to $\overline{WR}$ transition | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{WHQX}$ | 14, 15 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{QVWH}$ | 15 | Data valid to $\overline{WR}$ high | 287 | | $7t_{CLCL}-150$ | | ns |
| $t_{RLAZ}$ | 14, 15 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 14, 15 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 17 | High time | 20 | | 20 | $t_{CLCL}-t_{CLCX}$ | ns |
| $t_{CLCX}$ | 17 | Low time | 20 | | 20 | $t_{CLCL}-t_{CHCX}$ | ns |
| $t_{CLCH}$ | 17 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 17 | Fall time | | 20 | | 20 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 16 | Serial port clock cycle time | 750 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 16 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 16 | Output data hold after clock rising edge | 8 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 16 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 16 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}-133$ | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 80C52/54/58 to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.
4. See application note AN457 for external memory interfacing.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 24MHz CLOCK | | VARIABLE CLOCK[4] | | 33MHz CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 13 | Oscillator frequency<br>Speed versions : I (24MHz)<br>: N (33MHz) | 3.5 | 24 | 3.5 | 33 | 3.5 | 33 | MHz |
| $t_{LHLL}$ | 13 | ALE pulse width | 43 | | $2t_{CLCL}-40$ | | 21 | | ns |
| $t_{AVLL}$ | 13 | Address valid to ALE low | 17 | | $t_{CLCL}-25$ | | 5 | | ns |
| $t_{LLAX}$ | 13 | Address hold after ALE low | 17 | | $t_{CLCL}-25$ | | | | ns |
| $t_{LLIV}$ | 13 | ALE low to valid instruction in | | 102 | | $4t_{CLCL}-65$ | | 55 | ns |
| $t_{LLPL}$ | 13 | ALE low to $\overline{PSEN}$ low | 17 | | $t_{CLCL}-25$ | | 5 | | ns |
| $t_{PLPH}$ | 13 | $\overline{PSEN}$ pulse width | 80 | | $3t_{CLCL}-45$ | | 45 | | ns |
| $t_{PLIV}$ | 13 | $\overline{PSEN}$ low to valid instruction in | | 65 | | $3t_{CLCL}-60$ | | 30 | ns |
| $t_{PXIX}$ | 13 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 13 | Input instruction float after $\overline{PSEN}$ | | 17 | | $t_{CLCL}-25$ | | 5 | ns |
| $t_{AVIV}$ | 13 | Address to valid instruction in | | 128 | | $5t_{CLCL}-80$ | | 70 | ns |
| $t_{PLAZ}$ | 13 | $\overline{PSEN}$ low to address float | | 10 | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | | | |
| $t_{RLRH}$ | 14, 15 | $\overline{RD}$ pulse width | 150 | | $6t_{CLCL}-100$ | | 82 | | ns |
| $t_{WLWH}$ | 14, 15 | $\overline{WR}$ pulse width | 150 | | $6t_{CLCL}-100$ | | 82 | | ns |
| $t_{RLDV}$ | 14, 15 | $\overline{RD}$ low to valid data in | | 118 | | $5t_{CLCL}-90$ | | 60 | ns |
| $t_{RHDX}$ | 14, 15 | Data hold after $\overline{RD}$ | 0 | | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 14, 15 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-28$ | | 32 | ns |
| $t_{LLDV}$ | 14, 15 | ALE low to valid data in | | 183 | | $8t_{CLCL}-150$ | | 90 | ns |
| $t_{AVDV}$ | 14, 15 | Address to valid data in | | 210 | | $9t_{CLCL}-165$ | | 105 | ns |
| $t_{LLWL}$ | 14, 15 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 75 | 175 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | 40 | 140 | ns |
| $t_{AVWL}$ | 14, 15 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 92 | | $4t_{CLCL}-75$ | | 45 | | ns |
| $t_{QVWX}$ | 14, 15 | Data valid to $\overline{WR}$ transition | 12 | | $t_{CLCL}-30$ | | 0 | | ns |
| $t_{WHQX}$ | 14, 15 | Data hold after $\overline{WR}$ | 17 | | $t_{CLCL}-25$ | | 5 | | ns |
| $t_{QVWH}$ | 15 | Data valid to $\overline{WR}$ high | 162 | | $7t_{CLCL}-130$ | | 80 | | ns |
| $t_{RLAZ}$ | 14, 15 | $\overline{RD}$ low to address float | | 0 | | 0 | | 0 | ns |
| $t_{WHLH}$ | 14, 15 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 17 | 67 | $t_{CLCL}-25$ | $t_{CLCL}+25$ | 5 | 55 | ns |
| **External Clock** | | | | | | | | | |
| $t_{CHCX}$ | 17 | High time | 17 | | 17 | $t_{CLCL}-t_{CLCX}$ | | | ns |
| $t_{CLCX}$ | 17 | Low time | 17 | | 17 | $t_{CLCL}-t_{CHCX}$ | | | ns |
| $t_{CLCH}$ | 17 | Rise time | | 5 | | 5 | | | ns |
| $t_{CHCL}$ | 17 | Fall time | | 5 | | 5 | | | ns |
| **Shift Register** | | | | | | | | | |
| $t_{XLXL}$ | 16 | Serial port clock cycle time | 505 | | $12t_{CLCL}$ | | 360 | | ns |
| $t_{QVXH}$ | 16 | Output data setup to clock rising edge | 283 | | $10t_{CLCL}-133$ | | 167 | | ns |
| $t_{XHQX}$ | 16 | Output data hold after clock rising edge | 3 | | $2t_{CLCL}-80$ | | | | ns |
| $t_{XHDX}$ | 16 | Input data hold after clock rising edge | 0 | | 0 | | 0 | | ns |
| $t_{XHDV}$ | 16 | Clock rising edge to input data valid | | 283 | | $10t_{CLCL}-133$ | | 167 | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 80C52/54/58 to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.
4. Variable clock is specified for oscillator frequencies greater than 16MHz to 33MHz. For frequencies equal or less than 16MHz, see 16MHz "AC Electrial Characteristics", page 3-206.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ =Time for ALE low to $\overline{PSEN}$ low.

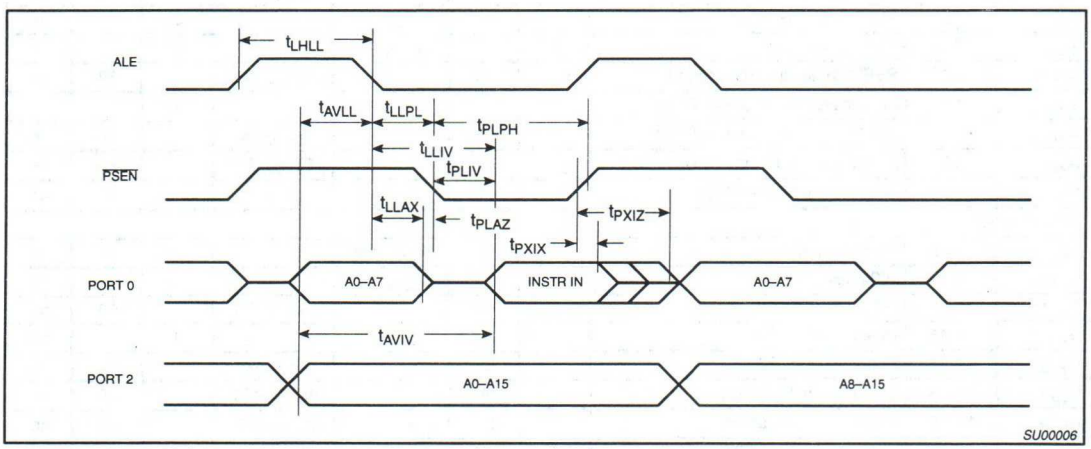



**Figure 13. External Program Memory Read Cycle**

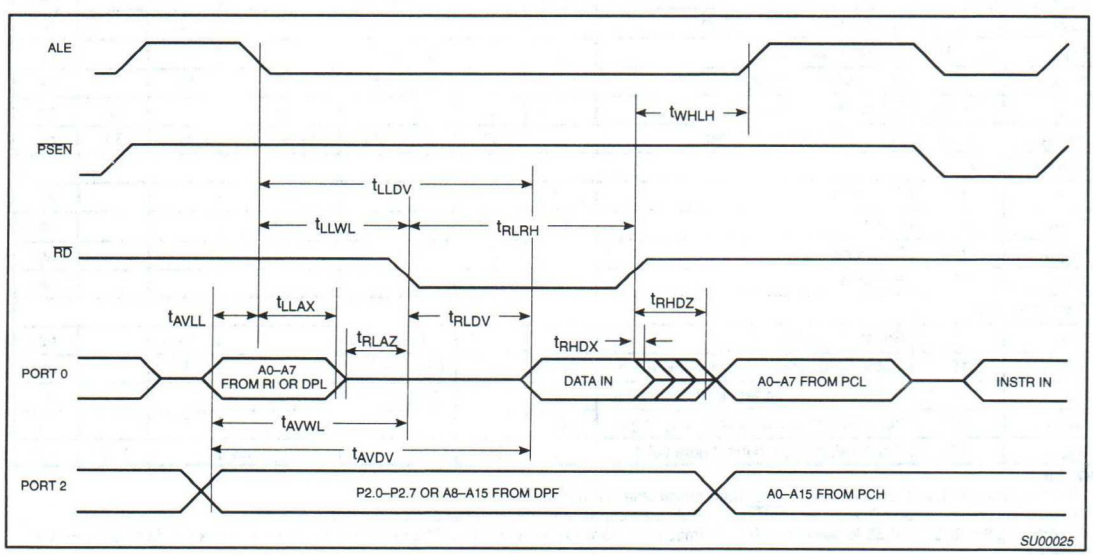



**Figure 14. External Data Memory Read Cycle**

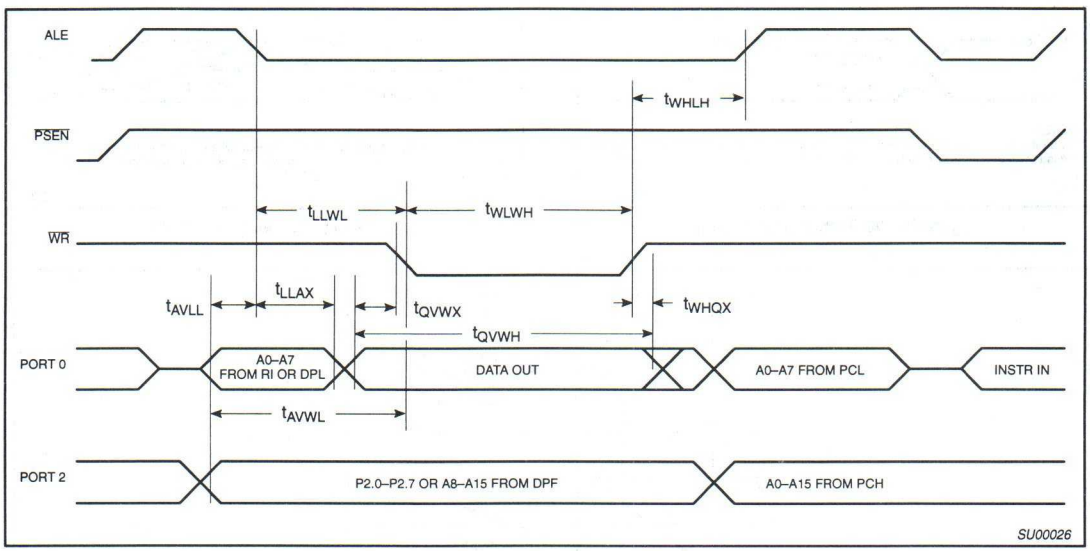



**Figure 15. External Data Memory Write Cycle**

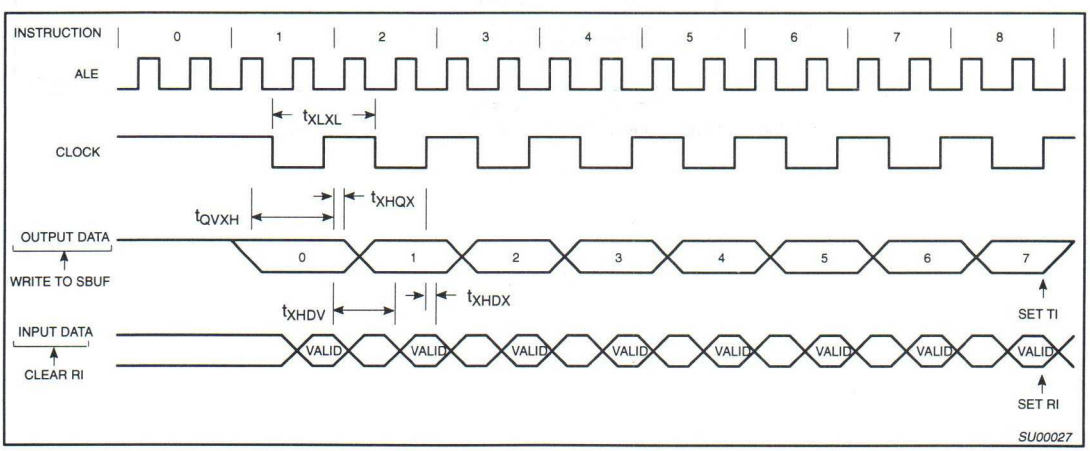



**Figure 16. Shift Register Mode Timing**

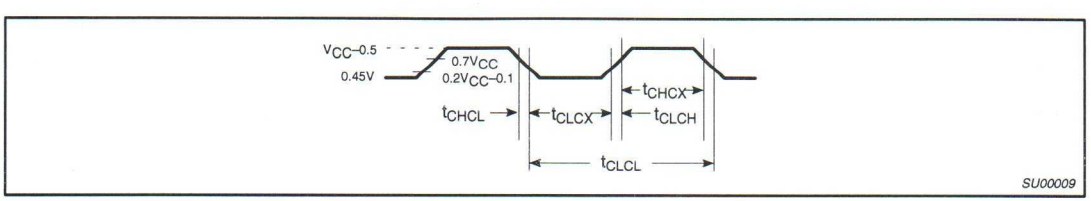



**Figure 17. External Clock Drive**

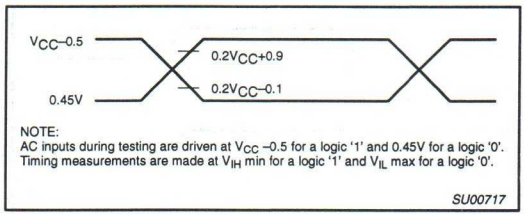



**Figure 18. AC Testing Input/Output**

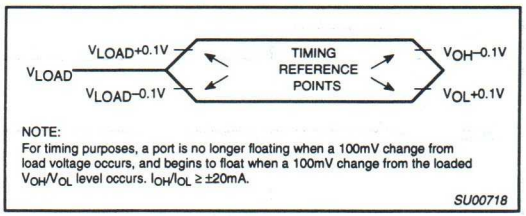



**Figure 19. Float Waveform**

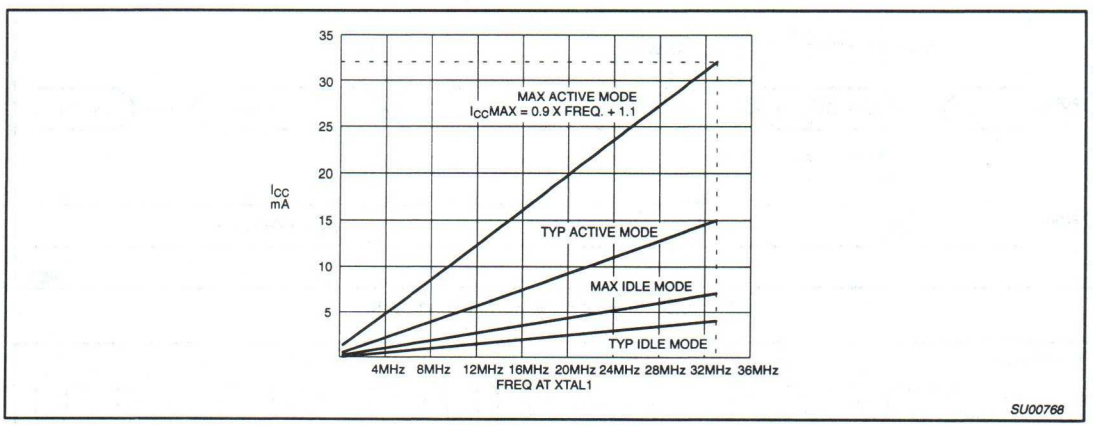



**Figure 20. $I_{CC}$ vs. FREQ**
**Valid only within frequency specifications of the device under test**

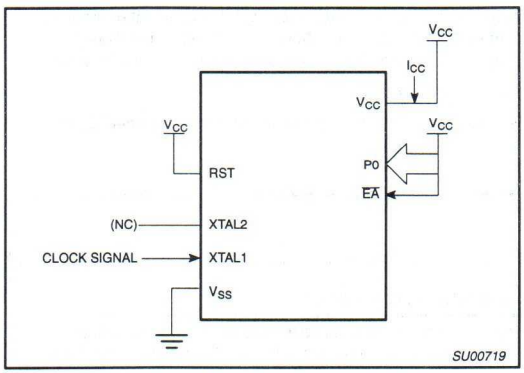



**Figure 21. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

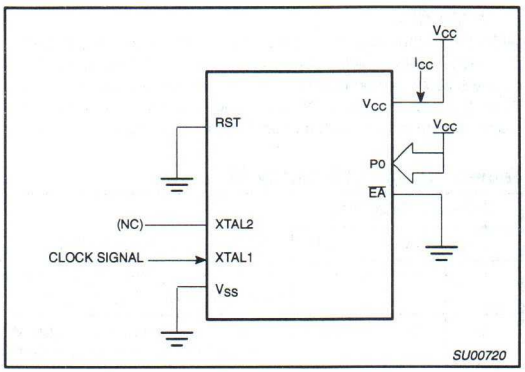



**Figure 22. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

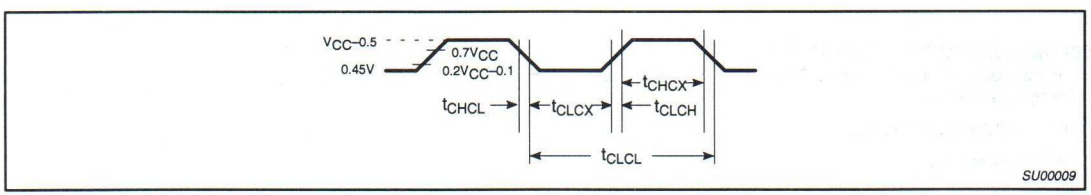



**Figure 23. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

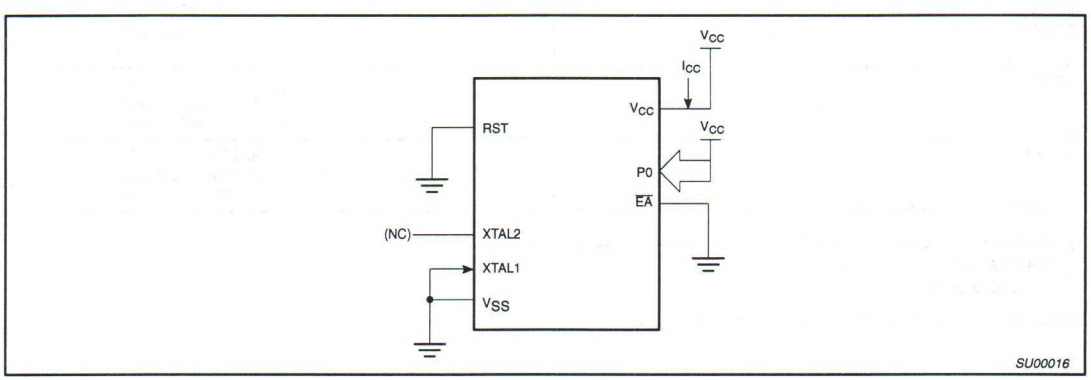



**Figure 24. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2V to 5.5V**

### Security Bits

With none of the security bits programmed the code in the program memory can be verified. If the encryption table is programmed, the code will be encrypted when verified. When only security bit 1 (see Table 8) is programmed, MOVC instructions executed from external program memory are disabled from fetching code bytes from the internal memory, $\overline{EA}$ is latched on Reset and all further programming of the EPROM is disabled. When security bits 1 and 2 are programmed, in addition to the above, verify mode is disabled.

### Encryption Array

64 bytes of encryption array are initially unprogrammed (all 1s).

**Table 8. Program Security Bits**

| PROGRAM LOCK BITS[1,2] | | | |
|---|---|---|---|
| | SB1 | SB2 | PROTECTION DESCRIPTION |
| 1 | U | U | No Program Security features enabled.<br>(Code verify will still be encrypted by the Encryption Array if programmed.) |
| 2 | P | U | MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory, $\overline{EA}$ is sampled and latched on Reset, and further programming of the EPROM is disabled. |

**NOTES:**
1. P – programmed. U – unprogrammed.
2. Any other combination of the security bits is not defined.

## 80C52 ROM CODE SUBMISSION

When submitting ROM code for the 80C52, the following must be specified:
1. 8k byte user ROM data
2. 64 byte ROM encryption key
3. ROM security bits.

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 1FFFH | DATA | 7:0 | User ROM Data |
| 2000H to 201FH | KEY | 7:0 | ROM Encryption Key<br>FFH = no encryption |
| 2020H | SEC | 0 | ROM Security Bit 1<br>0 = enable security<br>1 = disable security |
| 2020H | SEC | 1 | ROM Security Bit 2<br>0 = enable security<br>1 = disable security |

**Security Bit 1:** When programmed, this bit has two effects on masked ROM parts:
1. External MOVC is disabled, and
2. $\overline{EA}$ is latched on Reset.

**Security Bit 2:** When programmed, this bit inhibits Verify User ROM.

If the ROM Code file does not include the options, the following information must be included with the ROM code.

For each of the following, check the appropriate box, and send to Philips along with the code:

Security Bit #1: ☐ Enabled ☐ Disabled

Security Bit #2: ☐ Enabled ☐ Disabled

Encryption: ☐ No ☐ Yes If Yes, must send key file.

## 80C54 ROM CODE SUBMISSION

When submitting ROM code for the 80C54, the following must be specified:

1. 16k byte user ROM data
2. 64 byte ROM encryption key
3. ROM security bits.

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 3FFFH | DATA | 7:0 | User ROM Data |
| 4000H to 401FH | KEY | 7:0 | ROM Encryption Key<br>FFH = no encryption |
| 4020H | SEC | 0 | ROM Security Bit 1<br>0 = enable security<br>1 = disable security |
| 4020H | SEC | 1 | ROM Security Bit 2<br>0 = enable security<br>1 = disable security |

**Security Bit 1:** When programmed, this bit has two effects on masked ROM parts:

1. External MOVC is disabled, and
2. $\overline{EA}$ is latched on Reset.

**Security Bit 2:** When programmed, this bit inhibits Verify User ROM.

If the ROM Code file does not include the options, the following information must be included with the ROM code.

For each of the following, check the appropriate box, and send to Philips along with the code:

Security Bit #1: ☐ Enabled ☐ Disabled

Security Bit #2: ☐ Enabled ☐ Disabled

Encryption: ☐ No ☐ Yes If Yes, must send key file.

## 80C58 ROM CODE SUBMISSION

When submitting ROM code for the 80C58, the following must be specified:

1. 32k byte user ROM data
2. 64 byte ROM encryption key
3. ROM security bits.

If submitting a file, the format is as follows:

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 7FFFH | DATA | 7:0 | User ROM Data |
| 8000H to 801FH | KEY | 7:0 | ROM Encryption Key<br>FFH = no encryption |
| 8020H | SEC | 0 | ROM Security Bit 1<br>0 = enable security<br>1 = disable security |
| 8020H | SEC | 1 | ROM Security Bit 2<br>0 = enable security<br>1 = disable security |

**Security Bit 1:** When programmed, this bit has two effects on masked ROM parts:

1. External MOVC is disabled, and
2. $\overline{EA}$ is latched on Reset.

**Security Bit 2:** When programmed, this bit inhibits Verify User ROM.

If the ROM code file does not include the options, the following information must be included with the ROM code.

For each of the following check the appropriate box and send to Philips along with the code:

Security Bit #1: ☐ Enabled ☐ Disabled

Security Bit #2: ☐ Enabled ☐ Disabled

Encryption: ☐ No ☐ Yes If Yes, must send key file.

# CMOS single-chip 8-bit microcontrollers 87C54/87C58

## DESCRIPTION

The 87C54/87C58 Single-Chip 8-Bit Microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 87C54/87C58 has the same instruction set as the 80C51.

This device provides architectural enhancements that make it applicable in a variety of applications for general control systems. The 87C58 contains 32k × 8 EPROM memory, and the 87C54 contains 16k × 8 EPROM memory, a volatile 256 × 8 read/write data memory, four 8-bit I/O ports, three 16-bit timer/event counters, a multi-source, two-priority-level, nested interrupt structure, an enhanced UART and on-chip oscillator and timing circuits. For systems that require extra capability, the 87C54/87C58 can be expanded using standard TTL compatible memories and logic.

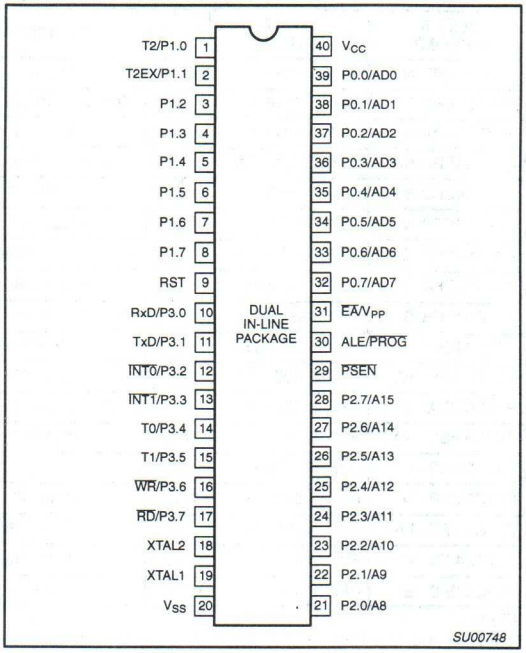
Its added features make it an even more powerful microcontroller for applications that require pulse width modulation, high-speed I/O and up/down counting capabilities such as motor control. It also has a more versatile serial channel that facilitates multiprocessor communications.

See 80C52/54/58 datasheet for ROM device specification.

## FEATURES

- 80C51 central processing unit
- 16k × 8 EPROM expandable externally to 64k bytes (87C54)
- 16k × 8 EPROM (87C54) and
  32k × 8 EPROM expandable externally to 64k bytes (87C58)
  - Improved Quick Pulse programming algorithm
  - Two level program security system
  - 32 byte encryption array
- 256 × 8 RAM, expandable externally to 64k bytes
- Three 16-bit timer/counters
  - T2 is an up/down counter
- Four 8-bit I/O ports
- Full-duplex enhanced UART
  - Framing error detection
  - Automatic address recognition
- Power control modes
  - Idle mode
  - Power-down mode
- Once (On Circuit Emulation) Mode
- Five package styles
- OTP package available
- Programmable clock out
- 6 interrupt sources
- 2 level priority

## PIN CONFIGURATIONS

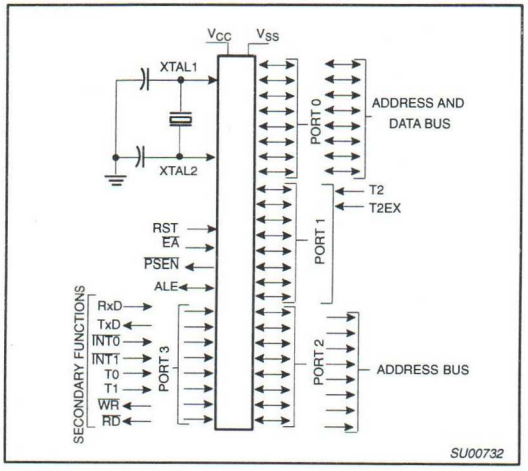


## ORDERING INFORMATION

| 16k × 8 EPROM[1] | 32k × 8 EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQUENCY | DRAWING NUMBER |
|---|---|---|---|---|---|
| P87C54EBP N | P87C58EBP N | OTP | 0 to +70, 40-Pin Plastic Dual In-line Package | 16MHz | SOT129-1 |
| P87C54EBF FA | P87C58EBF FA | UV | 0 to +70, 40-Pin Ceramic Dual In-line Package w/Window | 16MHz | 0590B |
| P87C54EBA A | P87C58EBA A | OTP | 0 to +70, 44-Pin Plastic Leaded Chip Carrier | 16MHz | SOT187-2 |
| P87C54EBL KA | P87C58EBL KA | UV | 0 to +70, 44-Pin Ceramic Leaded Chip Carrier w/Window | 16MHz | 1472A |
| P87C54EBB B | P87C58EBB B | OTP | 0 to +70, 44-Pin Plastic Quad Flat Pack | 16MHz | SOT307-2 |
| P87C54EFP N | P87C58EFP N | OTP | –40 to +85, 40-Pin Plastic Dual In-line Package | 16MHz | SOT129-1 |
| P87C54EFF FA | P87C58EFF FA | UV | –40 to +85, 40-Pin Ceramic Dual In-line Package w/Window | 16MHz | 0590B |
| P87C54EFA A | P87C58EFA A | OTP | –40 to +85, 44-Pin Plastic Leaded Chip Carrier | 16MHz | SOT187-2 |
| P87C54EFB B | P87C58EFB B | OTP | –40 to +85, 44-Pin Plastic Quad Flat Pack | 16MHz | SOT307-2 |
| P87C54IBP N | P87C58IBP N | OTP | 0 to +70, 40-Pin Plastic Dual In-line Package | 24MHz | SOT129-1 |
| P87C54IBF FA | P87C58IBF FA | UV | 0 to +70, 40-Pin Ceramic Dual In-line Package w/Window | 24MHz | 0590B |
| P87C54IBA A | P87C58IBA A | OTP | 0 to +70, 44-Pin Plastic Leaded Chip Carrier | 24MHz | SOT187-2 |
| P87C54IBL KA | P87C58IBL KA | UV | 0 to +70, 44-Pin Ceramic Leaded Chip Carrier w/Window | 24MHz | 1472A |
| P87C54IBB B | P87C58IBB B | OTP | 0 to +70, 44-Pin Plastic Quad Flat Pack | 24MHz | SOT307-2 |
| P87C54IFP N | P87C58IFP N | OTP | –40 to +85, 40-Pin Plastic Dual In-line Package | 24MHz | SOT129-1 |
| P87C54IFF FA | P87C58IFF FA | UV | –40 to +85, 40-Pin Ceramic Dual In-line Package w/Window | 24MHz | 0590B |
| P87C54IFA A | P87C58IFA A | OTP | –40 to +85, 44-Pin Plastic Leaded Chip Carrier | 24MHz | SOT187-2 |
| P87C54IFB B | P87C58IFB B | OTP | –40 to +85, 44-Pin Plastic Quad Flat Pack | 24MHz | SOT307-2 |

**NOTE:**
1. OTP = One Time Programmable EPROM. UV = Erasable EPROM.

## LOGIC SYMBOL



## BLOCK DIAGRAM

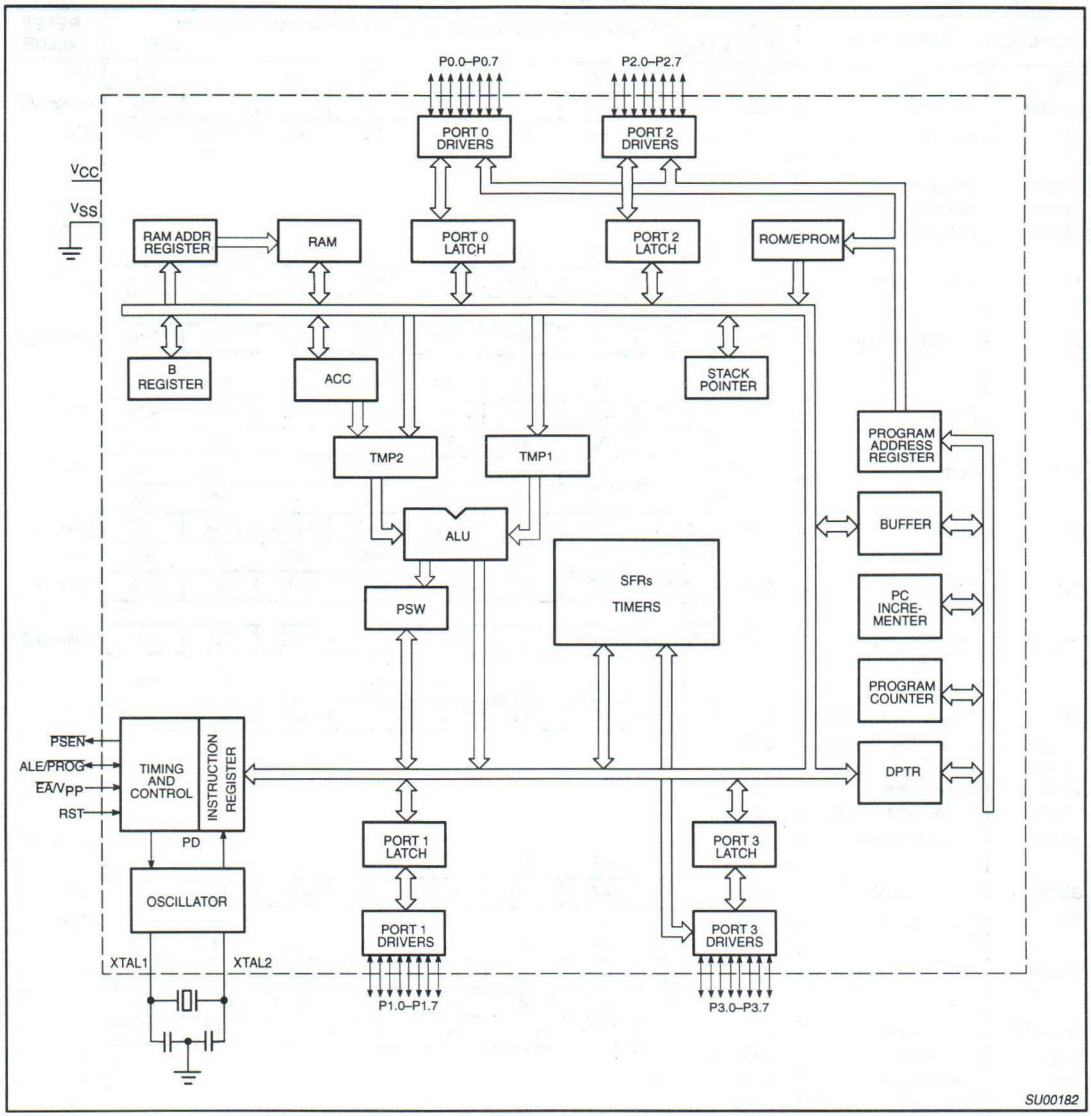
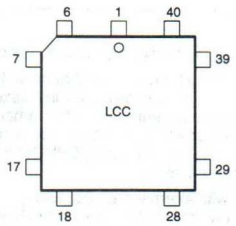


**Table 1. 87C54/87C58 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| AUXR# | Auxiliary | 8EH | – | – | – | – | – | – | – | AO | xxxxxxx0B |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data Pointer (2 bytes) | | | | | | | | | | |
| DPH | Data Pointer High | 83H | | | | | | | | | 00H |
| DPL | Data Pointer Low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE* | Interrupt Enable | A8H | EA | – | ET2 | ES | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP* | Interrupt Priority | B8H | – | – | PT2 | PS | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | – | – | – | – | – | – | T2EX | T2 | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | AD15 | AD14 | AD13 | AD12 | AD11 | AD10 | AD9 | AD8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TxD | RxD | FFH |
| PCON# | Power Control | 87H | SMOD1 | SMOD0 | – | POF[1] | GF1 | GF0 | PD | IDL | 00xxxx00B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program Status Word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| RCAP2H# | Timer 2 Capture High | CBH | | | | | | | | | 00H |
| RCAP2L# | Timer 2 Capture Low | CAH | | | | | | | | | 00H |
| SADDR# | Slave Address | A9H | | | | | | | | | 00H |
| SADEN# | Slave Address Mask | B9H | | | | | | | | | 00H |
| SBUF | Serial Data Buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial Control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack Pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer Control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| T2CON#* | Timer 2 Control | C8H | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | $C/\overline{T2}$ | $CP/\overline{RL2}$ | 00H |
| TH0 | Timer High 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer High 1 | 8DH | | | | | | | | | 00H |
| TH2# | Timer High 2 | CDH | | | | | | | | | 00H |
| TL0 | Timer Low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer Low 1 | 8BH | | | | | | | | | 00H |
| TL2# | Timer Low 2 | CCH | | | | | | | | | 00H |
| | | | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 | |
| TMOD | Timer Mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |
| T2MOD#* | Timer 2 Mode Control | C9H | – | – | – | – | – | – | T2OE | DCEN | xxxxxx00B |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
1. Reset value depends on reset source.

## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

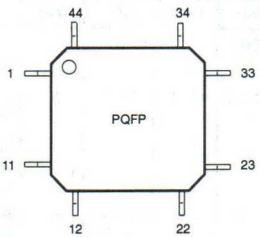



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | NC* | 16 | T0/P3.4 | 31 | P2.7/A15 |
| 2 | T2/P1.0 | 17 | T1/P3.5 | 32 | $\overline{PSEN}$ |
| 3 | T2EX/P1.1 | 18 | $\overline{WR}$/P3.6 | 33 | ALE/$\overline{PROG}$ |
| 4 | P1.2 | 19 | $\overline{RD}$/P3.7 | 34 | NC* |
| 5 | P1.3 | 20 | XTAL2 | 35 | $\overline{EA}$/$V_{PP}$ |
| 6 | P1.4 | 21 | XTAL1 | 36 | P0.7/AD7 |
| 7 | P1.5 | 22 | $V_{SS}$ | 37 | P0.6/AD6 |
| 8 | P1.6 | 23 | NC* | 38 | P0.5/AD5 |
| 9 | P1.7 | 24 | P2.0/A8 | 39 | P0.4/AD4 |
| 10 | RST | 25 | P2.1/A9 | 40 | P0.3/AD3 |
| 11 | RxD/P3.0 | 26 | P2.2/A10 | 41 | P0.2/AD2 |
| 12 | NC* | 27 | P2.3/A11 | 42 | P0.1/AD1 |
| 13 | TxD/P3.1 | 28 | P2.4/A12 | 43 | P0.0/AD0 |
| 14 | $\overline{INT0}$/P3.2 | 29 | P2.5/A13 | 44 | $V_{CC}$ |
| 15 | $\overline{INT1}$/P3.3 | 30 | P2.6/A14 | | |

* DO NOT CONNECT

*SU00061*

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS




| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | P1.5 | 16 | $V_{SS}$ | 31 | P0.6/AD6 |
| 2 | P1.6 | 17 | NC* | 32 | P0.5/AD5 |
| 3 | P1.7 | 18 | P2.0/A8 | 33 | P0.4/AD4 |
| 4 | RST | 19 | P2.1/A9 | 34 | P0.3/AD3 |
| 5 | RxD/P3.0 | 20 | P2.2/A10 | 35 | P0.2/AD2 |
| 6 | NC* | 21 | P2.3/A11 | 36 | P0.1/AD1 |
| 7 | TxD/P3.1 | 22 | P2.4/A12 | 37 | P0.0/AD0 |
| 8 | $\overline{INT0}$/P3.2 | 23 | P2.5/A13 | 38 | $V_{CC}$ |
| 9 | $\overline{INT1}$/P3.3 | 24 | P2.6/A14 | 39 | NC* |
| 10 | T0/P3.4 | 25 | P2.7/A15 | 40 | T2/P1.0 |
| 11 | T1/P3.5 | 26 | $\overline{PSEN}$ | 41 | T2EX/P1.1 |
| 12 | $\overline{WR}$/P3.6 | 27 | ALE/$\overline{PROG}$ | 42 | P1.2 |
| 13 | $\overline{RD}$/P3.7 | 28 | NC* | 43 | P1.3 |
| 14 | XTAL2 | 29 | $\overline{EA}$/$V_{PP}$ | 44 | P1.4 |
| 15 | XTAL1 | 30 | P0.7/AD7 | | |

* DO NOT CONNECT

*SU00062*

## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER DIP | PIN NUMBER LCC | PIN NUMBER QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification and receives code bytes during EPROM programming. External pull-ups are required during program verification. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups, except P1.6 and P1.7 which are open drain. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also receives the low-order address byte during program memory verification. Alternate functions include: |
| | 1 | 2 | 40 | I | **T2 (P1.0):** Timer/Counter 2 external count input/Clockout |
| | 2 | 3 | 41 | I | **T2EX (P1.1):** Timer/Counter 2 Reload/Capture/Direction Control |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. Some Port 2 pins receive the high order address bits during EPROM programming and verification. |
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{CC}$. |
| ALE/$\overline{PROG}$ | 30 | 33 | 27 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. ALE can be disabled by setting SFR auxiliary.0. With this bit set, ALE will be active only during a MOVX instruction. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the 8XC58 is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}$/$V_{PP}$ | 31 | 35 | 29 | I | **External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H and 7FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 7FFFH. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. If security bit 1 is programmed, $\overline{EA}$ will be internally latched on Reset. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTE:**
To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher than $V_{CC}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

## TIMER 2

This is a 16-bit up or down counter, which can be operated as either a timer or event counter. It can be operated in one of three different modes (autoreload, capture or as the baud rate generator for the UART).

In the autoreload mode the Timer can be set to count up or down by setting or clearing the bit DCEN in the T2CON Special Function Register. The SFR's RCAP2H and RCAP2L are used to reload the Timer upon overflow or a 1-to-0 transition on the T2EX input (P1.1).

In the Capture mode Timer 2 can either set TF2 and generate an interrupt or capture its value. To capture Timer 2 in response to a 1-to-0 transition on the T2EX input, the EXEN2 bit in the T2CON must be set. Timer 2 is then captured in SFR's RCAP2H and RCAP2L.

As the baud rate generator, Timer 2 is selected by setting TCLK and/or RCLK in T2CON. As the baud rate generator Timer 2 is incremented at $^1/_2$ the oscillator frequency.

## POWER OFF FLAG

The Power Off Flag (POF) is set by on-chip circuitry when the $V_{CC}$ level on the 8XC58 rises from 0 to 5V. The POF bit can be set or cleared by software allowing a user to determine if the reset is the result of a power-on or a warm start after powerdown. The $V_{CC}$ level must remain above 3V for the POF to remain unaffected by the $V_{CC}$ level.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

### Reset

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up. Ports 1, 2, and 3 will asynchronously be driven to their reset condition when a voltage above $V_{IH1}$ is applied to RESET.

### Idle Mode

In the idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

### Power-Down Mode

To save even more power, a Power Down mode can be invoked by software. In this mode, the oscillator is stopped and the instruction that invoked Power Down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the Power Down mode is terminated.

On the 8XC58 either a hardware reset or external interrupt can use an exit from Power Down. Reset redefines all the SFRs but does not change the on-chip RAM. An external interrupt allows both the SFRs and the on-chip RAM to retain their values.

To properly terminate Power Down the reset or external interrupt should not be executed before $V_{CC}$ is restored to its normal operating level and must be held active long enough for the oscillator to restart and stabilize (normally less than 10ms).

With an external interrupt, INT0 and INT1 must be enabled and configured as level-sensitive. Holding the pin low restarts the oscillator but bringing the pin back high completes the exit. Once the interrupt is serviced, the next instruction to be executed after RETI will be the one following the instruction that put the device into Power Down.

### Design Consideration

- When the idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal rest algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.
- The windowed parts must be covered with an opaque label to assure proper chip operation.

### ONCE™ Mode

The ONCE ("On-Circuit Emulation") Mode facilitates testing and debugging of systems using the 8XC58 without the 8XC58 having to be removed from the circuit. The ONCE Mode is invoked by:

1. Pull ALE low while the device is in reset and $\overline{PSEN}$ is high;
2. Hold ALE low as RST is deactivated.

While the device is in ONCE Mode, the Port 0 pins go into a float state, and the other port pins and ALE and $\overline{PSEN}$ are weakly pulled high. The oscillator circuit remains active. While the 8XC58 is in this mode, an emulator or test CPU can be used to drive the circuit. Normal operation is restored when a normal reset is applied.

**Table 2. External Pin Status During Idle and Power-Down Mode**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

## Programmable Clock-Out

The 87C54/87C58 has a new feature. A 50% duty cycle clock can be programmed to come out on P1.0. This pin, besides being a regular I/O pin, has two alternate functions. It can be programmed (1) to input the external clock for Timer/Counter 2 or (2) to output a 50% duty cycle clock ranging from 61Hz to 4MHz at a 16MHz operating frequency.

To configure the Timer/Counter 2 as a clock generator, bit C/T2 (in T2CON) must be cleared and bit T20E in T2MOD must be set. Bit TR2 (T2CON.2) also must be set to start the timer.

The Clock-Out frequency depends on the oscillator frequency and the reload value of Timer 2 capture registers (RCAP2H, RCAP2L) as shown in this equation:

$$\frac{\text{OscillatorFrequency}}{4 \times (65536 - \text{RCAP2H, RCAP2L})}$$

In the Clock-Out mode Timer 2 roll-overs will not generate an interrupt. This is similar to when it is used as a baud-rate generator. It is possible to use Timer 2 as a baud-rate generator and a clock generator simultaneously. Note, however, that the baud-rate and the Clock-Out frequency will be the same.

## Enhanced UART

The UART operates in all of the usual modes that are described in the first section of this book for the 80C51. In addition the UART can perform framing error detect by looking for missing stop bits, and automatic address recognition. The 87C54/87C58 UART also fully supports multiprocessor communication as does the standard 80C51 UART.

When used for framing error detect the UART looks for missing stop bits in the communication. A missing bit will set the FE bit in the SCON register. The FE bit shares the SCON.7 bit with SM0 and the function of SCON.7 is determined by PCON.6 (SMOD0) (see Figure 1). If SMOD0 is set then SCON.7 functions as FE. SCON.7 functions as SM0 when SMOD0 is cleared. When used as FE SCON.7 can only be cleared by software. Refer to Figure 2.

### Automatic Address Recognition

Automatic Address Recognition is a feature which allows the UART to recognize certain addresses in the serial bit stream by using hardware to make the comparisons. This feature saves a great deal of software overhead by eliminating the need for the software to examine every serial address which passes by the serial port. This feature is enabled by setting the SM2 bit in SCON. In the 9 bit UART modes, mode 2 and mode 3, the Receive Interrupt flag (RI) will be automatically set when the received byte contains either the "Given" address or the "Broadcast" address. The 9 bit mode requires that the 9th information bit is a 1 to indicate that the received information is an address and not data. Automatic address recognition is shown in Figure 3.

The 8 bit mode is called Mode 1. In this mode the RI flag will be set if SM2 is enabled and the information received has a valid stop bit following the 8 address bits and the information is either a Given or Broadcast address.

Mode 0 is the Shift Register mode and SM2 is ignored.

Using the Automatic Address Recognition feature allows a master to selectively communicate with one or more slaves by invoking the Given slave address or addresses. All of the slaves may be contacted by using the Broadcast address. Two special Function Registers are used to define the slave's address, SADDR, and the address mask, SADEN. SADEN is used to define which bits in the SADDR are to b used and which bits are "don't care". The SADEN mask can be logically ANDed with the SADDR to create the "lGiven" address which the master will use for addressing each of the slaves. Use of the Given address allows multiple slaves to be recognized while excluding others. The following examples will help to show the versatility of this scheme:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1101 |
| | Given | = | 1100 00X0 |
| Slave 1 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1110 |
| | Given | = | 1100 000X |

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires a 0 in bit 0 and it ignores bit 1. Slave 1 requires a 0 in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010 since slave 1 requires a 0 in bit 1. A unique address for slave 1 would be 1100 0001 since a 1 in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 1100 0000.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1001 |
| | Given | = | 1100 0XX0 |
| Slave 1 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1010 |
| | Given | = | 1110 0X0X |
| Slave 2 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1100 |
| | Given | = | 1110 00XX |

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 and 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary t make bit 2 = 1 to exclude slave 2.

The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are teated as don't-cares. In most cases, interpreting the don't-cares as ones, the broadcast address will be FF hexadecimal.

Upon reset SADDR (SFR address 0A9H) and SADEN (SFR address 0B9H) are loaded with 0s. This produces a given address of all "don't cares" as well as a Broadcast address of all "don't cares". this effectively disables the Automatic Addressing mode and allows the microcontroller to use standard 80C51 type UART drivers which do not make use of this feature.

## Reduced EMI Mode

The AO bit (AUXR.0) in the AUXR register, when set, disables the ALE output.

**8XC58 Reduced EMI Mode**

**AUXR (0X8E)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | – | – | – | AO |

AO: Turns off ALE output.

## Interrupt Priority Structure

The 87C54/87C58 has a 6-source two-level interrupt structure. There are 3 SFRs associated with the interrupts. They are the IE and IP which are identical in function to those on the 80C51.

The priority scheme for servicing the interrupts is the same as that for the 80C51. An interrupt will be serviced as long as an interrupt of equal or higher priority is not already being serviced. If an interrupt of equal or higher level priority is being serviced, the new interrupt will wait until it is finished before being serviced. If a lower priority level interrupt is being serviced, it will be stopped and the new interrupt serviced. When the new interrupt is finished, the lower priority level interrupt that was stopped will be completed.

**Table 3. Interrupt Table**

| SOURCE | POLLING PRIORITY | REQUEST BITS | HARDWARE CLEAR? | VECTOR ADDRESS |
|---|---|---|---|---|
| X0 | 1 | IE0 | N (L) Y (T) | 03H |
| T0 | 2 | TP0 | Y | 0B |
| X1 | 3 | IE1 | N (L) Y (T) | 13 |
| T1 | 4 | TF1 | Y | 1B |
| SP | 5 | R1, TI | N | 23 |
| T2 | 6 | TF2, EXF2 | N | 2B |

**SCON** Address = 98H — Reset Value = 0000 0000B

Bit Addressable

| | SM0/FE | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

(SMOD0 = 0/1)*

| Symbol | Function |
|---|---|
| **FE** | Framing Error bit. This bit is set by the receiver when an invalid stop bit is detected. The FE bit is not cleared by valid frames but should be cleared by software. The SMOD0 bit must be set to enable access to the FE bit. |
| **SM0** | Serial Port Mode Bit 0, (SMOD0 must = 0 to access bit SM0) |
| **SM1** | Serial Port Mode Bit 1 |

| SM0 | SM1 | Mode | Description | Baud Rate** |
|---|---|---|---|---|
| 0 | 0 | 0 | shift register | $f_{OSC}/12$ |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{OSC}/64$ or $f_{OSC}/32$ |
| 1 | 1 | 3 | 9-bit UART | variable |

| | |
|---|---|
| **SM2** | Enables the Automatic Address Recognition feature in Modes 2 or 3. If SM2 = 1 then RI will not be set unless the received 9th data bit (RB8) is 1, indicating an address, and the received byte is a Given or Broadcast Address. In Mode 1, if SM2 = 1 then RI will not be activated unless a valid stop bit was received, and the received byte is a Given or Broadcast Address. In Mode 0, SM2 should be 0. |
| **REN** | Enables serial reception. Set by software to enable reception. Clear by software to disable reception. |
| **TB8** | The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired. |
| **RB8** | In modes 2 and 3, the 9th data bit that was received. In Mode 1, if SM2 = 0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used. |
| **TI** | Transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software. |
| **RI** | Receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software. |

**NOTE:**
*SMOD0 is located at PCON6.
**$f_{OSC}$ = oscillator frequency

SU00043

**Figure 1. SCON: Serial Port Control Register**

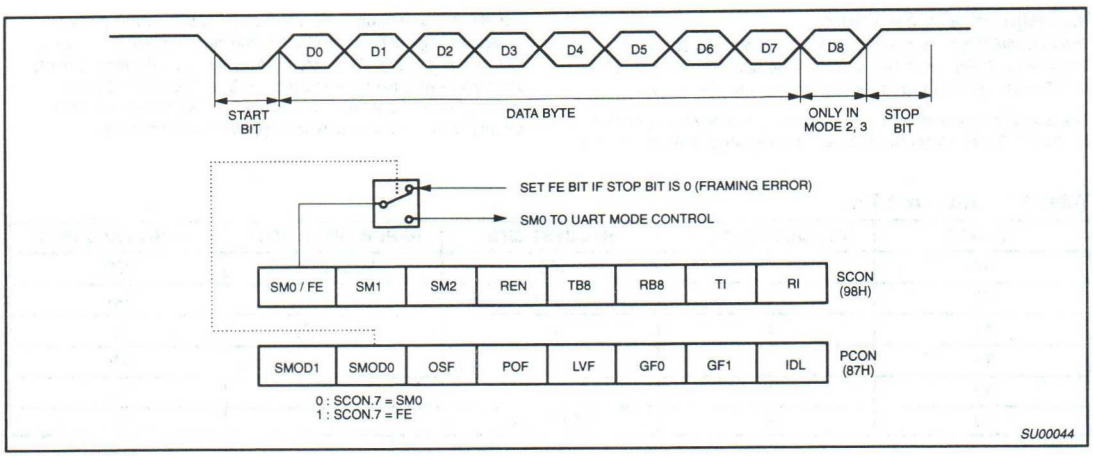



**Figure 2. UART Framing Error Detection**

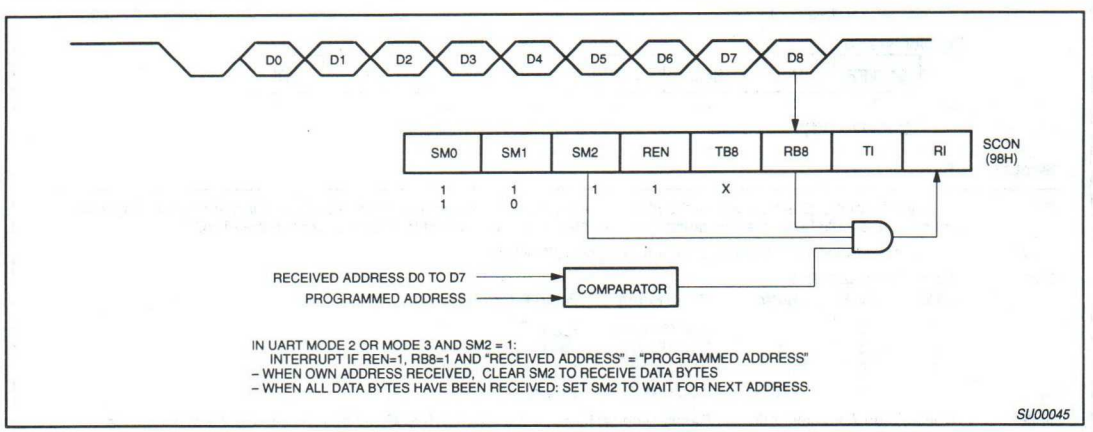



**Figure 3. UART Multiprocessor Communication, Automatic Address Recognition**

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | 0 to +70 or –40 to +85 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Maximum $I_{OL}$ per I/O pin | 15 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.5 | W |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$ | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | | 0 | | $0.2V_{CC}$–0.3 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3[7] | $I_{OL}$ = 1.6mA[2] | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$[7] | $I_{OL}$ = 3.2mA[2] | | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3 [3] | $I_{OH}$ = –30µA | $V_{CC}$ – 0.7 | | | V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode), ALE[8], $\overline{PSEN}$[3] | $I_{OH}$ = –3.2mA | $V_{CC}$ – 0.7 | | | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3 | $V_{IN}$ = 0.4V | | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3[5] | See note 4 | | | –650 | µA |
| $I_{LI}$ | Input leakage current, port 0 | 0.45 $V_{IN}$ < $V_{CC}$ – 0.3 | | | ±10 | µA |
| $I_{CC}$ | Power supply current (See Figure 11): | See note 10 | | | | |
| | Active mode @ 16MHz | | | 15 | 32 | mA |
| | Idle mode @ 16MHz | | | 3 | 5 | mA |
| | Power-down mode $T_{amb}$ = 0 to +70°C | | | 10 | 75 | µA |
| | Power-down mode $T_{amb}$ = –40 to +85°C | | | | 100 | µA |
| $R_{RST}$ | Internal reset pull-down resistor | | 40 | | 225 | kΩ |
| $C_{IO}$ | Pin capacitance[9] (except $\overline{EA}$) | | | | 15 | pF |

**NOTES:**

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
5. This value applies to $T_{amb}$ = 0°C to +70°C. For $T_{amb}$ = –40°C to 85°C, $I_{TL}$ = –750µA.
6. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
7. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 15mA
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 71mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
8. ALE is tested to $V_{OH1}$, except when ALE is off then $V_{OH}$ is the voltage specification.
9. Pin capacitance is characterized but not tested. Pin capacitance is less than 25pF. Pin capacitance of ceramic package is less than 15pF (except $\overline{EA}$ it is 25pF).
10. See Figures 12 through 15 for $I_{CC}$ test condition.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 4 | Oscillator frequency<br>Speed versions : E | | | 3.5 | 16 | MHz |
| $t_{LHLL}$ | 4 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 4 | Address valid to ALE low | 22 | | $t_{CLCL}-40$ | | ns |
| $t_{LLAX}$ | 4 | Address hold after ALE low | 32 | | $t_{CLCL}-30$ | | ns |
| $t_{LLIV}$ | 4 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 4 | ALE low to $\overline{PSEN}$ low | 32 | | $t_{CLCL}-30$ | | ns |
| $t_{PLPH}$ | 4 | $\overline{PSEN}$ pulse width | 142 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 4 | $\overline{PSEN}$ low to valid instruction in | | 82 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 4 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 4 | Input instruction float after $\overline{PSEN}$ | | 37 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 4 | Address to valid instruction in | | 207 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 4 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 5, 6 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 5, 6 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 5, 6 | $\overline{RD}$ low to valid data in | | 147 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 5, 6 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 5, 6 | Data float after $\overline{RD}$ | | 65 | | $2t_{CLCL}-60$ | ns |
| $t_{LLDV}$ | 5, 6 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 5, 6 | Address to valid data in | | 397 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 5, 6 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 137 | 239 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 5, 6 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 122 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 5, 6 | Data valid to $\overline{WR}$ transition | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{WHQX}$ | 5, 6 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{QVWH}$ | 6 | Data valid to $\overline{WR}$ high | 287 | | $7t_{CLCL}-150$ | | ns |
| $t_{RLAZ}$ | 5, 6 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 5, 6 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 8 | High time | 20 | | 20 | $t_{CLCL}+t_{CLCX}$ | ns |
| $t_{CLCX}$ | 8 | Low time | 20 | | 20 | $t_{CLCL}+t_{CHCX}$ | ns |
| $t_{CLCH}$ | 8 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 8 | Fall time | | 20 | | 20 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 7 | Serial port clock cycle time | 750 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 7 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 7 | Output data hold after clock rising edge | 8 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 7 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 7 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}-133$ | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 8XC58 to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 24MHz CLOCK | | VARIABLE CLOCK[4] | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 4 | Oscillator frequency<br>Speed versions : I | | | 3.5 | 24 | MHz |
| $t_{LHLL}$ | 4 | ALE pulse width | 43 | | $2t_{CLCL}$–40 | | ns |
| $t_{AVLL}$ | 4 | Address valid to ALE low | 17 | | $t_{CLCL}$–25 | | ns |
| $t_{LLAX}$ | 4 | Address hold after ALE low | 17 | | $t_{CLCL}$–25 | | ns |
| $t_{LLIV}$ | 4 | ALE low to valid instruction in | | 102 | | $4t_{CLCL}$–65 | ns |
| $t_{LLPL}$ | 4 | ALE low to $\overline{PSEN}$ low | 17 | | $t_{CLCL}$–25 | | ns |
| $t_{PLPH}$ | 4 | $\overline{PSEN}$ pulse width | 80 | | $3t_{CLCL}$–45 | | ns |
| $t_{PLIV}$ | 4 | $\overline{PSEN}$ low to valid instruction in | | 65 | | $3t_{CLCL}$–60 | ns |
| $t_{PXIX}$ | 4 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 4 | Input instruction float after $\overline{PSEN}$ | | 17 | | $t_{CLCL}$–25 | ns |
| $t_{AVIV}$ | 4 | Address to valid instruction in | | 128 | | $5t_{CLCL}$–80 | ns |
| $t_{PLAZ}$ | 4 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 5, 6 | $\overline{RD}$ pulse width | 150 | | $6t_{CLCL}$–100 | | ns |
| $t_{WLWH}$ | 5, 6 | $\overline{WR}$ pulse width | 150 | | $6t_{CLCL}$–100 | | ns |
| $t_{RLDV}$ | 5, 6 | $\overline{RD}$ low to valid data in | | 118 | | $5t_{CLCL}$–90 | ns |
| $t_{RHDX}$ | 5, 6 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 5, 6 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}$–28 | ns |
| $t_{LLDV}$ | 5, 6 | ALE low to valid data in | | 183 | | $8t_{CLCL}$–150 | ns |
| $t_{AVDV}$ | 5, 6 | Address to valid data in | | 210 | | $9t_{CLCL}$–165 | ns |
| $t_{LLWL}$ | 5, 6 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 75 | 175 | $3t_{CLCL}$–50 | $3t_{CLCL}$+50 | ns |
| $t_{AVWL}$ | 5, 6 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 92 | | $4t_{CLCL}$–75 | | ns |
| $t_{QVWX}$ | 5, 6 | Data valid to $\overline{WR}$ transition | 12 | | $t_{CLCL}$–30 | | ns |
| $t_{WHQX}$ | 5, 6 | Data hold after $\overline{WR}$ | 17 | | $t_{CLCL}$–25 | | ns |
| $t_{QVWH}$ | 6 | Data valid to $\overline{WR}$ high | 162 | | $7t_{CLCL}$–130 | | ns |
| $t_{RLAZ}$ | 5, 6 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 5, 6 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 17 | 67 | $t_{CLCL}$–25 | $t_{CLCL}$+25 | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 8 | High time | 17 | | 17 | $t_{CLCL}$–$t_{CLCX}$ | ns |
| $t_{CLCX}$ | 8 | Low time | 17 | | 17 | $t_{CLCL}$–$t_{CHCX}$ | ns |
| $t_{CLCH}$ | 8 | Rise time | | 5 | | 5 | ns |
| $t_{CHCL}$ | 8 | Fall time | | 5 | | 5 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 7 | Serial port clock cycle time | 505 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 7 | Output data setup to clock rising edge | 283 | | $10t_{CLCL}$–133 | | ns |
| $t_{XHQX}$ | 7 | Output data hold after clock rising edge | 3 | | $2t_{CLCL}$–80 | | ns |
| $t_{XHDX}$ | 7 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 7 | Clock rising edge to input data valid | | 283 | | $10t_{CLCL}$–133 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 87C58 to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.
4. Variable clock is specified for oscillator frequencies greater than 16MHz to 24MHz. For frequencies equal or less than 16MHz, see 16MHz "AC Electrial Characteristics", page 3-226.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ =Time for ALE low to $\overline{PSEN}$ low.

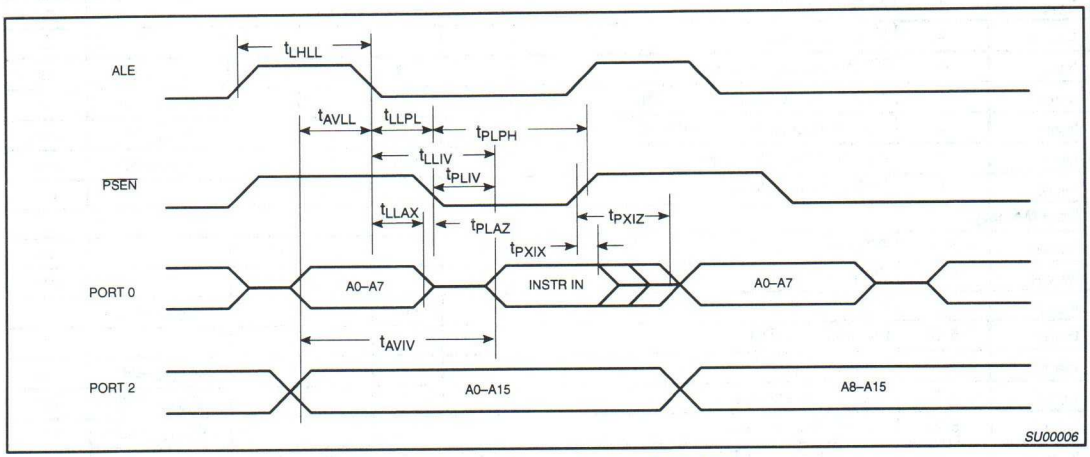



**Figure 4. External Program Memory Read Cycle**

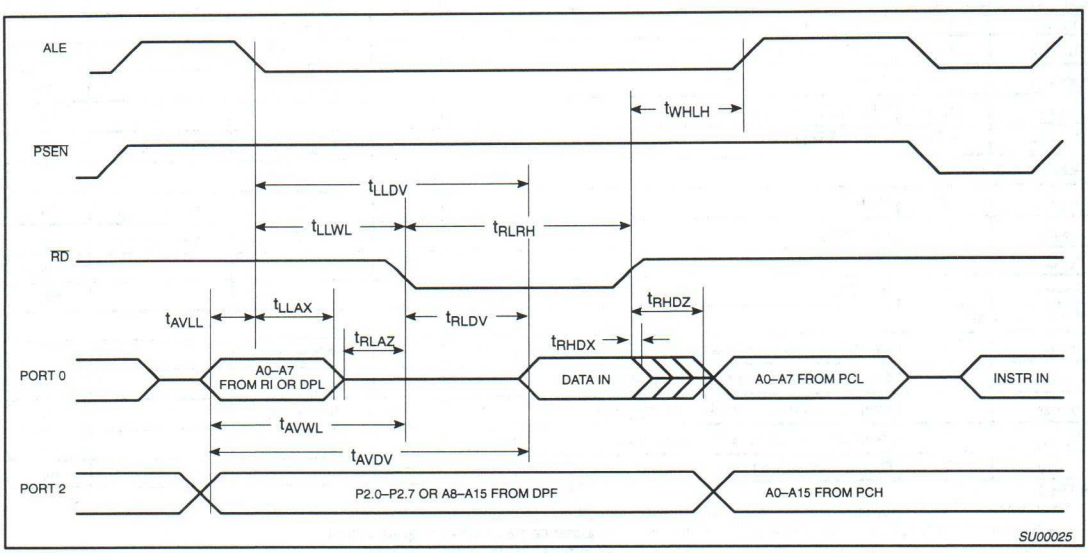



**Figure 5. External Data Memory Read Cycle**

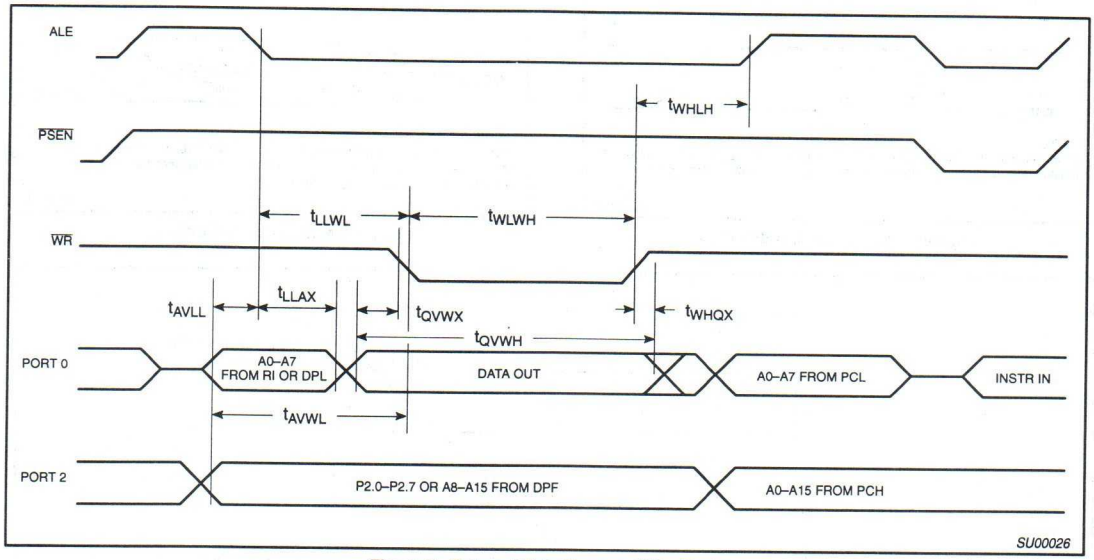



**Figure 6. External Data Memory Write Cycle**

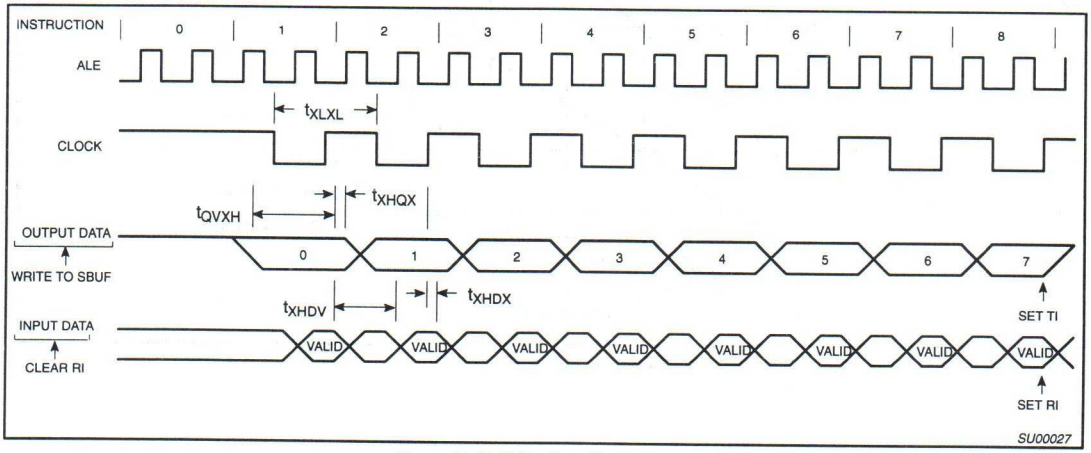



**Figure 7. Shift Register Mode Timing**

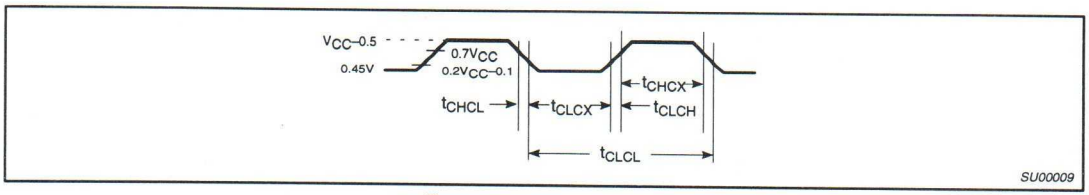
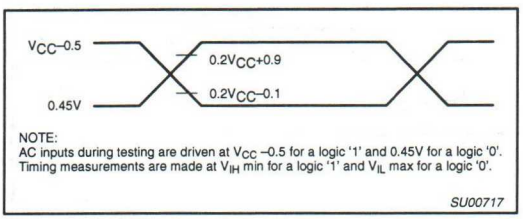


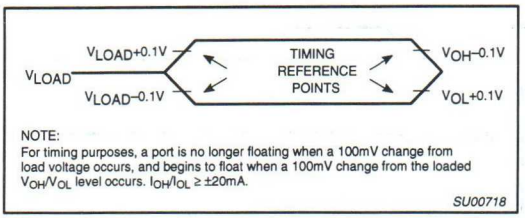

**Figure 8. External Clock Drive**




NOTE:
AC inputs during testing are driven at $V_{CC}$ –0.5 for a logic '1' and 0.45V for a logic '0'. Timing measurements are made at $V_{IH}$ min for a logic '1' and $V_{IL}$ max for a logic '0'.

SU00717

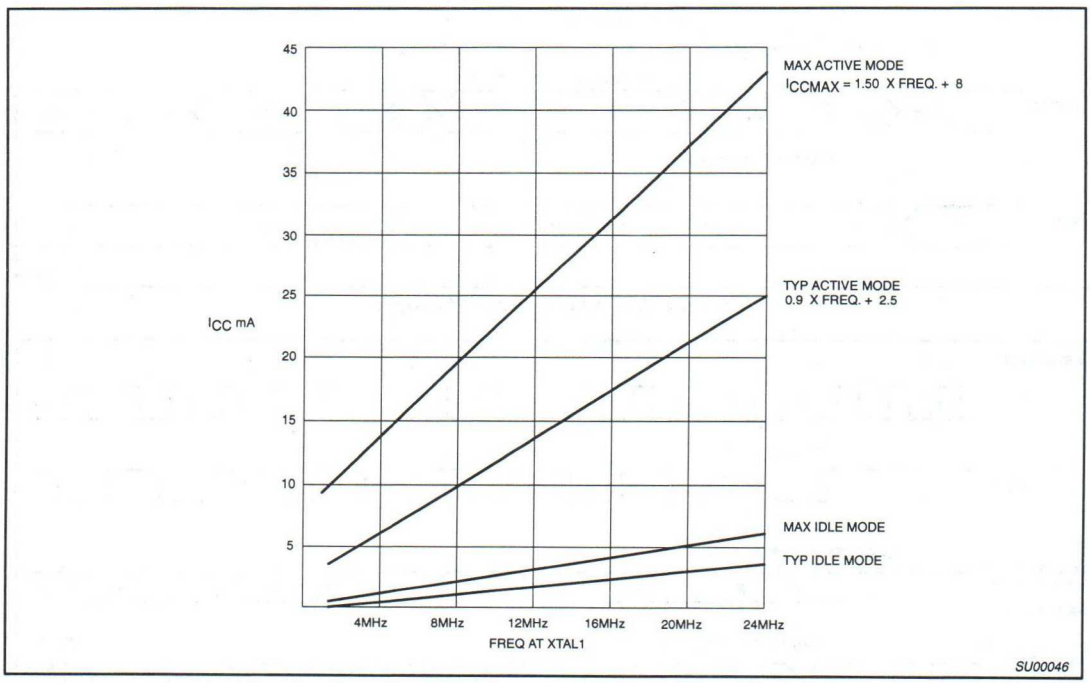
**Figure 9. AC Testing Input/Output**




NOTE:
For timing purposes, a port is no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded $V_{OH}/V_{OL}$ level occurs. $I_{OH}/I_{OL} \geq \pm 20$mA.

SU00718

**Figure 10. Float Waveform**




SU00046

**Figure 11. $I_{CC}$ vs. Frequency**

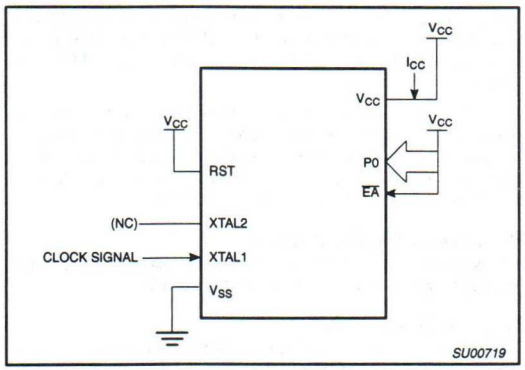



**Figure 12. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

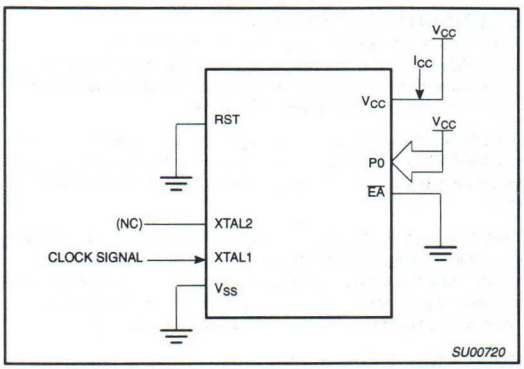



**Figure 13. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

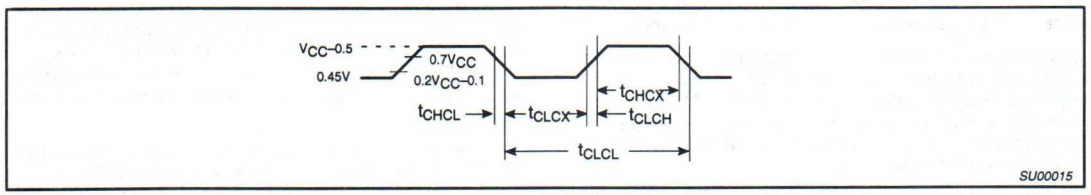



**Figure 14. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

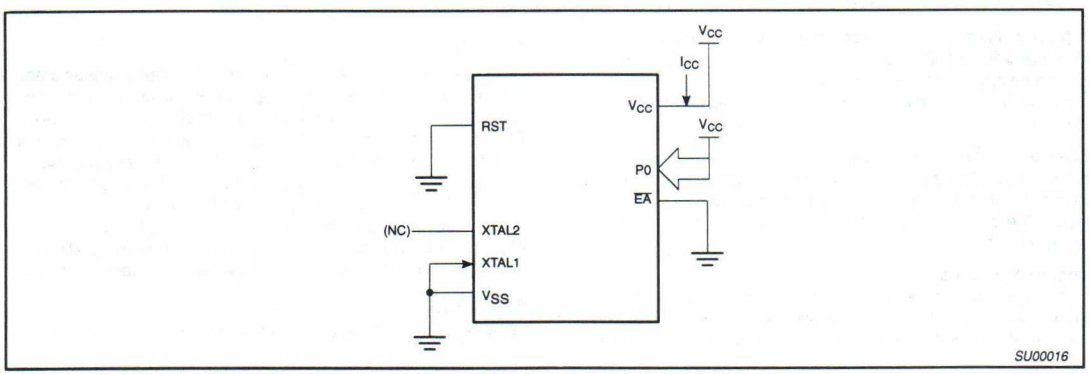



**Figure 15. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2V to 5.5V**

## EPROM CHARACTERISTICS

The 87C58 is programmed by using a modified Improved Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C58 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C58 manufactured by Philips.

Table 4 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the security bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 16 and 17. Figure 18 shows the circuit configuration for normal program memory verification.

### Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 16. Note that the 87C58 is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1 and 2, as shown in Figure 16. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 3 specified in Table 4 are held at the 'Program Code Data' levels indicated in Table 4. The ALE/$\overline{PROG}$ is pulsed low 5 times as shown in Figure 17.

To program the encryption table, repeat the 25 pulse programming sequence for addresses 0 through 1FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the security bits, repeat the 25 pulse programming sequence using the 'Pgm Security Bit' levels. After one security bit is programmed, further programming of the code memory and encryption table is disabled. However, the other security bit can still be programmed.

Note that the $\overline{EA}$/$V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

#### Program Verification

If security bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 1 and 2 as shown in Figure 18. The other pins are held at the 'Verify Code Data' levels indicated in Table 4. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the 32 byte encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

### Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 4, and which satisfies the timing specifications, is suitable.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-s/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

### Security Bits

With none of the security bits programmed the code in the program memory can be verified. If the encryption table is programmed, the code will be encrypted when verified. When only security bit 1 (see Table 5) is programmed, MOVC instructions executed from external program memory are disabled from fetching code bytes from the internal memory, EA is latched on Reset and all further programming of the EPROM is disabled. When security bits 1 and 2 are programmed, in addition to the above, verify mode is disabled. When all three security bits are programmed, all of the conditions above apply and all external program memory execution is disabled.

### Encryption Array

32 bytes of encryption array are initially unprogrammed (all 1s).

™Trademark phrase of Intel Corporation.

### Table 4. EPROM Programming Modes

| MODE | RST | PSEN | ALE/PROG | EA/$V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 | P3.3 |
|---|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 | 1 |
| Pgm security bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 | 1 |
| Pgm security bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 | 1 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V±10% during programming and verification.
* ALE/PROG receives 5 programming pulses (only for user array; 25 pulses for encryption or security bits) while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

### Table 5. Program Security Bits

| PROGRAM LOCK BITS[1, 2] | | | |
|---|---|---|---|
| | SB1 | SB2 | PROTECTION DESCRIPTION |
| 1 | U | U | No Program Security features enabled. (Code verify will still be encrypted by the Encryption Array if programmed.) |
| 2 | P | U | MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory, EA is sampled and latched on Reset, and further programming of the EPROM is disabled. |
| 3 | P | P | Same as 2, also verify is disabled. |

**NOTES:**
1. P – programmed. U – unprogrammed.
2. Any other combination of the security bits is not defined.

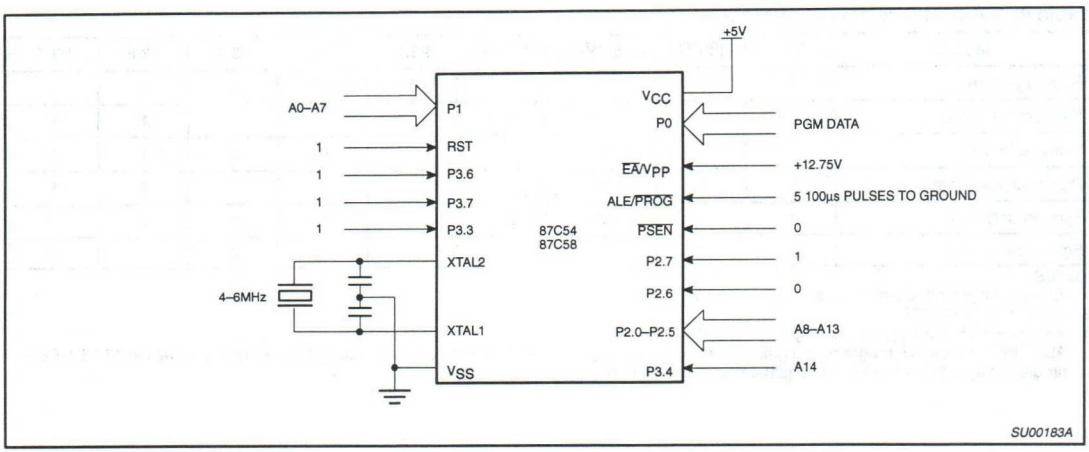



**Figure 16. Programming Configuration**

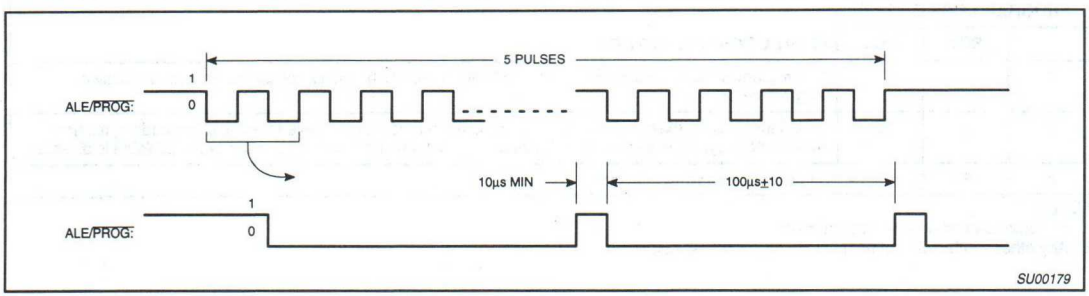



**Figure 17. PROG Waveform**

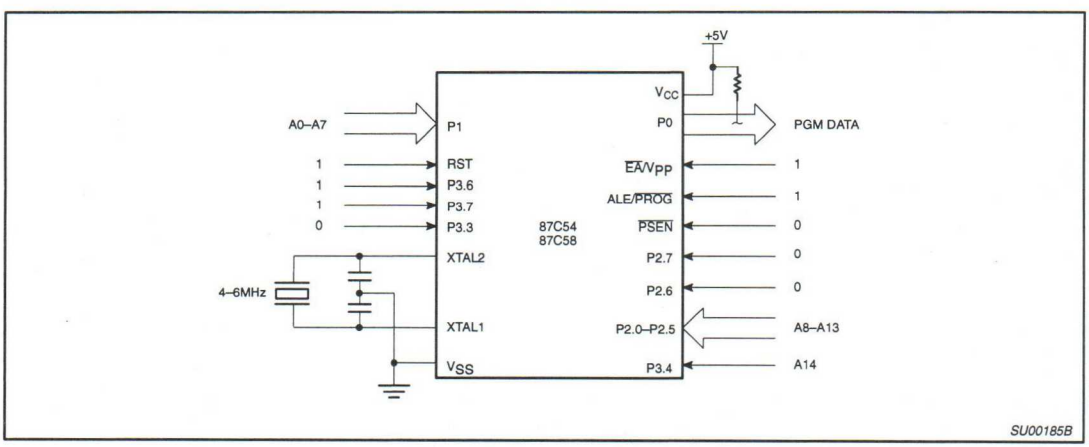


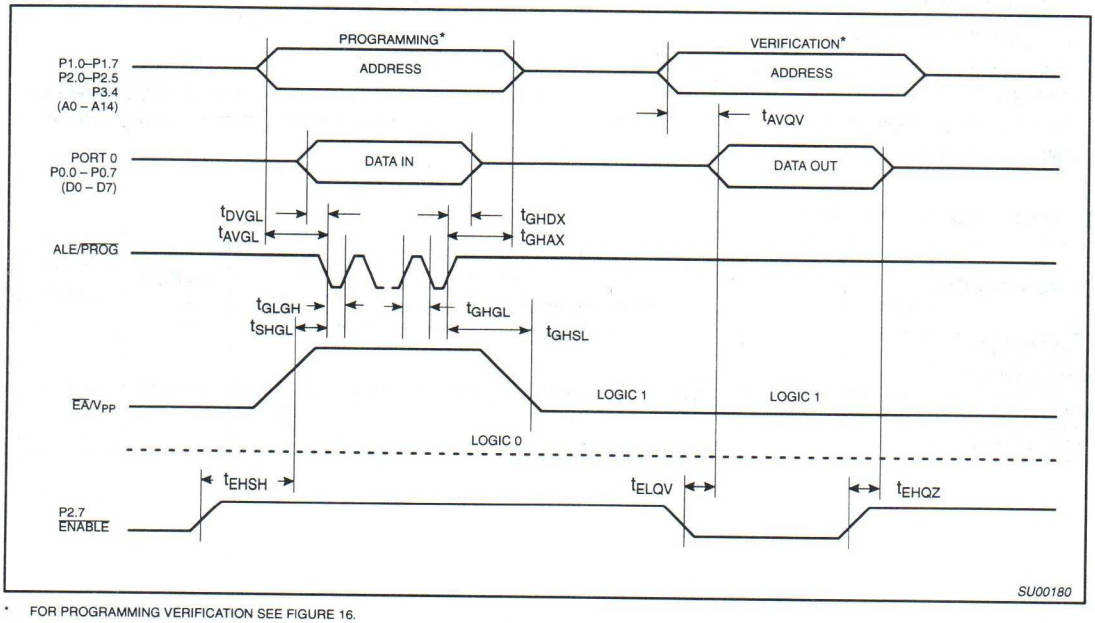

**Figure 18. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (See Figure 19)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 [1] | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |

**NOTE:**
1. Not tested.




* FOR PROGRAMMING VERIFICATION SEE FIGURE 16.
FOR VERIFICATION CONDITIONS SEE FIGURE 18.

**Figure 19. EPROM Programming and Verification**

# Microcontrollers for TV and video (MTV)

# 83C145; 83C845 83C055; 87C055

## 1 FEATURES

- Masked ROM sizes:
  - 8 kbytes (83C845)
  - 12 kbytes (83C145)
  - 16 kbytes (83C055)
  - 16 kbytes OTP (87C055)
- RAM: 256 bytes
- On Screen Display (OSD) controller
- Three digital video outputs
- Multiplexer/mixer and background intensity controls
- Flexible formatting with OSD New Line option
- 128 x 10 bits display RAM
- Designed for reduced Radio Frequency Interference (RFI)
- Character generator ROM:
  - character format 18 lines x 14 dots
  - 60 visible characters
  - 4 special characters
- Eight text shadowing modes
- Text colour selectable per character
- Background colour selectable per word
- Background colour versus video selectable per character
- Eight 6-bit Pulse Width Modulators (PWM) for analog voltage integration
- One 14-bit PWM for high-precision voltage integration
- Digital-to-analog converter and comparator with 3 inputs multiplexer
- Nine dedicated I/Os plus 28 port bits (15 port bits with alternative uses)
- 4 high current open-drain port outputs
- 12 high voltage (+12 V) open-drain outputs
- Programmable video input and output polarities
- 80C51 instruction set
- No external memory capability
- Plastic shrink dual in-line package (0.07 inch centre pins)
- High-speed CMOS technology
- Power supply: 5 V ±10%.

## 2 DESCRIPTION

The 83C055, Microcontroller for Television and Video (MTV) applications, is a derivative of Philips' industry standard 80C51 microcontroller.

The 83C055 is intended for use as the central control mechanism in a television receiver or tuner.

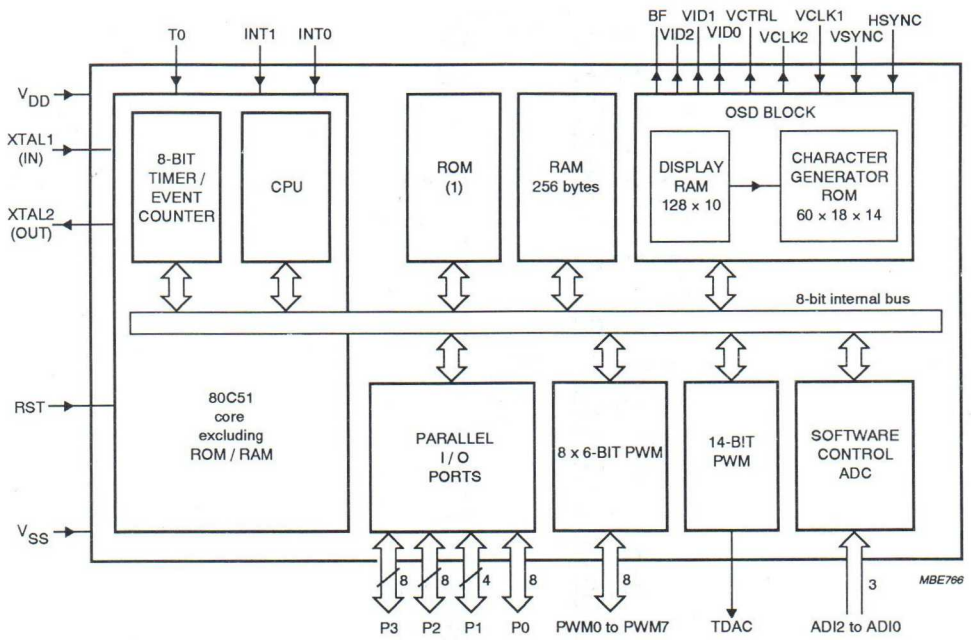
## 3 APPLICATIONS

Providing tuner functions and an OSD facility, it represents a next generation replacement for the currently available parts.

## 4 ORDERING INFORMATION

| TYPE NUMBER | PACKAGE | | | TEMP. RANGE (°C) | FREQ. (MHz) |
|---|---|---|---|---|---|
| | NAME | DESCRIPTION | VERSION | | |
| P83C055BBP | SDIP42 | plastic shrink dual in-line package; 42 leads (600 mil) | SOT270-1 | 0 to +70 | 3.5 to 12 |
| P87C055BBP | | | | | |
| P83C145BBP | | | | | |
| P83C845BBP | | | | | |

## 5 BLOCK DIAGRAM




(1) ROM sizes: see Table 1.

Fig.1 Block diagram.

### 5.1 Part options

**Table 1** Differences between the types

| MEMORY | TYPES | | | |
|---|---|---|---|---|
| | **83C845** | **83C145** | **83C055** | **87C055** |
| ROM | 8 kbytes | 12 kbytes | 16 kbytes | – |
| EPROM (OTP) | – | – | – | 16 kbytes |

## 6 PINNING INFORMATION

### 6.1 Pinning

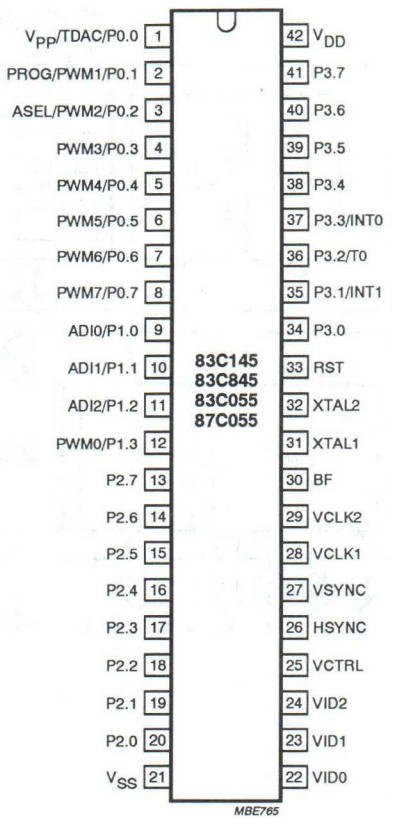



Fig.2 Pin configuration (SOT270-1).

### 6.2 Pin description

**Table 2** Pin description SDIP42 (SOT270-1)

| SYMBOL | PIN | DESCRIPTION |
|---|---|---|
| **Port 0** (notes 1, 2 and 4) | | |
| P0.0/TDAC/$V_{PP}$ | 1 | **P0.0**: open-drain bidirectional port line;<br>**TDAC**: output for the 14-bit high-precision PWM;<br>**$V_{PP}$**: 12 V programming supply voltage during EPROM programming. |
| P0.1/PWM1/PROG | 2 | **P0.1**: open-drain bidirectional port line;<br>**PWM1**: output for the 6-bit lower-precision PWM;<br>**PROG**: input for EPROM programming pulses. |
| P0.2/PWM2/ASEL | 3 | **P0.2**: open-drain bidirectional port line;<br>**PWM2**: output for the 6-bit lower-precision PWM;<br>**ASEL**: input indicating the EPROM address bits that are applied to Port 2. |
| P0.3/PWM3<br>to<br>P0.7/PWM7 | 4 to 8 | **P0.3 to P0.7**: 5 open-drain bidirectional port lines;<br>**PWM3 to PWM7**: 5 outputs for the 6-bit lower-precision PWM. |
| **Port 1** (notes 1, 2 and 5) | | |
| P1.0/ADI0<br>to<br>P1.2/ADI2 | 9 to 11 | **P1.0 to P1.2:** 3 open-drain bidirectional port lines;<br>**ADI0 to ADI2**: inputs for the software analog-to-digital facility. |
| P1.3/PWM0 | 12 | **P1.3**: open-drain bidirectional port line; **PWM0**: output for the 6-bit lower-precision PWM. PWM0 can be externally pulled up as high as +12 V ±5% |
| **Port 2** | | |
| P2.7 to P2.0 | 13 to 20 | **Port 2**: 8-bit open-drain bidirectional port; P2.3 to P2.0 have high current capability (10 mA at 0.5 V) for driving LEDs. Port 2 pins that have logic 1s written to them float, and in that state can be used as high-impedance inputs. Any of the Port 2 pins are driven LOW if the port register bit is written as a logic 0. The state of the pin can always be read from the port register by the program. |
| **Port 3** (note 1 and 3) | | |
| P3.0 | 34 | **P3.0**: open-drain bidirectional port line. |
| P3.1/INT1 | 35 | **P3.1**: open-drain bidirectional port line; **INT1**: External interrupt 1. |
| P3.2/T0 | 36 | **P3.2**: open-drain bidirectional port line; **T0**: Timer 0 external input. |
| P3.3/INT0 | 37 | **P3.3**: open-drain bidirectional port line; **INT0**: External interrupt 0. |
| P3.4 to P3.7 | 38 to 41 | **P3.4 to P3.7**: 4 open-drain bidirectional port lines. |
| **General** | | |
| $V_{SS}$ | 21 | **Ground**: 0 V reference. |
| VID2 to VID0 | 22 to 24 | **Digital Video bus**: Three totem-pole outputs comprising digital RGB (or other colour encoding) from the OSD facility. The polarity of these outputs is controlled by a programmable register bit (register OSCON; bit Po). |
| VCTRL | 25 | **Video Control**: A totem-pole output indicating whether the OSD facility is currently presenting active video on the VID2 to VID0 outputs. Signal is used to control an external multiplexer (mixer) between normal video and the video derived from VID2 to VID0. The polarity of this output is controlled by a programmable register bit (register OSCON; bit Pc). |

| SYMBOL | PIN | DESCRIPTION |
|---|---|---|
| HSYNC | 26 | **Horizontal Sync**: A dedicated input for a TTL-level version of the horizontal sync pulse. The polarity of this pulse is programmable; its trailing edge is used by the OSD facility as the reference for horizontal positioning. |
| VSYNC | 27 | **Vertical Sync**: A dedicated input for a TTL-level version of the vertical sync pulse. The polarity of this pulse is programmable, and either edge can serve as the reference for vertical timing. |
| VCLK1 | 28 | **VCLK1: Video Clock 1**; input for the horizontal timing reference for the OSD facility. **VCLK2: Video Clock 2**; output from the on-chip video oscillator. VCLK1 and VCLK2 are intended to be used with an external LC circuit to provide an on-chip oscillator. The period of the video clock is determined such that the width of a pixel in the OSD is equal to the inter-line separation of the raster. |
| VCLK2 | 29 | |
| BF | 30 | **Background/Foreground**: A totem-pole output which, when VCTRL is active, indicates whether the current video data represents a Foreground (LOW) or Background (HIGH) dot in a character. This signal can be used to reduce the intensity of the background colour and thus emphasize the text. |
| XTAL1 | 31 | **XTAL1**: Input to the inverting (oscillator) amplifier and clock generator circuit that provides the timing reference for all 83C055 logic other than the OSD facility. **XTAL2**: Oscillator output terminal for system clock. XTAL1 and XTAL2 can be used with a quartz crystal or ceramic resonator to provide an on-chip oscillator. Alternatively, XTAL1 can be connected to an external clock, and XTAL2 left unconnected. |
| XTAL2 | 32 | |
| RST | 33 | **Reset**: If this pin is HIGH for two machine cycles (24 oscillator periods) while the oscillator is running, the MTV is reset. This pin is also used as a serial input to enter a test or EPROM programming mode, as on the 87C751. |
| $V_{DD}$ | 42 | **Power supply**: for normal and Power-down operation. |

**Notes**

1. Port 0, Port 1 , and Port 3 pins that have logic 1s written to them float, and in that state can be used as high-impedance inputs.
2. The state of the pin can always be read from the port register by the program.
3. P3.0, P3.4, and P3.7 can be externally pulled up as high as +12 V ±5%; while P3.5 and P3.6 have 10 mA drive capability.
4. For each PWM block, a register bit (register PWMn; bit PWnE; n = 0 to 7) controls whether the corresponding pin is controlled by the block or by Port 0; Port 0 controls the pin immediately after a reset. Regardless of how each pin is controlled, it can be externally pulled up as high as +12 V ±5%.
5. Any of the Port 1 pins are driven LOW if the corresponding port register bit is written as a logic 0, or for P1.3 only, if the TDAC module presents a logic 0.

## 7 DESCRIPTION OF STANDARD FUNCTIONS

For a description of the standard functions please refer to the *"Data Handbook IC20; Section 2: 80C51 Technical Description"*.

## 8 INPUT/OUTPUT (I/O)

The I/O structure of the 83C055 is similar to the standard I/O structure in the 80C51, except for the points described in Table 5.

## 9 DESCRIPTION OF DERIVATIVE FUNCTIONS

### 9.1 General description

Although the 83C055 is specifically referred to throughout this data sheet, the information applies to all the devices. The differences to 80C51 features and the derivative functions are described in the following Sections and Chapters.

Figure 1 shows the block diagram of the 83C055.

#### 9.1.1 NOT IMPLEMENTED FUNCTIONS

Standard functions to the 80C51 that are not implemented in the 83C055:

- As Data and Program Memory are not externally expandable on the 83C055, the ALE, $\overline{EA}$, and $\overline{PSEN}$ signals are not implemented.
- Idle mode.
- Power-down mode.

#### 9.1.2 INTERRUPT FACILITIES DIFFERENCES

The interrupt facilities of the 83C055 differ from those of the 80C51 as follows:

- The IP register is not used, and the IE register (address A8H) is similar to that on the 80C51;see Table 36.
- The VSYNC input used by the OSD facility can generate an interrupt. The active polarity of the pulse is programmable (see Section 13.7); interrupt occurs at the leading edge of the pulse.
- Since there is no serial port, there are no interrupts nor control bits relating to this interrupt. The interrupts and their vector addresses are shown in Table 3.
- External Interrupt 1 is modified so that an interrupt is generated when the input switches are in either direction (on the 80C51, there is a programmable choice between interrupt on a negative edge or a LOW level on INT1). This facility allows for software pulse-width measurement handling of a remote control.

**Table 3** Program Memory address

| EVENT | PROGRAM MEMORY ADDRESS |
|---|---|
| Reset | 000H |
| External INT0 | 003H |
| Timer 0 | 00BH |
| External INT1 | 013H |
| Timer 1 | 01BH |
| VSync Start | 023H |

#### 9.1.3 PCON REGISTER DIFFERENCE

The PCON register format is shown in Table 4. Bits GF1 and GF0 are general purpose flag bits.

**Table 4** PCON Register format (address 87H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | GF1 | GF0 | – | – |

#### 9.1.4 I/O PORTS DIFFERENCES

**Table 5** I/O ports differences

| I/O | STANDARD 80C51 | 83C055 |
|---|---|---|
| Port 0 | external memory expansion | 8-bit open-drain bidirectional port; and includes: alternative use for PWM outputs |
| Port 1 | 8-bit general purpose quasi-bidirectional | 4-bit open-drain port, and includes alternative uses for analog inputs and a PWM output |
| Port 2 | quasi-bidirectional and can be used for external memory expansion | open-drain and general purpose |
| Port 3 | quasi-bidirectional; all eight bits have alternate uses | 3 port bits have some of the same alternative uses as on the 80C51 but not necessarily on the same pins; 5 pins are open-drain and general purpose |

## 10 6-BIT PWM DACS

Figure 3 shows the 6-bit PWM DAC logic circuit, consisting of 8 PWMn modules.

The basic MCU clock is divided by 4 to get a waveform that clocks a 14-bit counter which is common to all the PWMs (including the 14-bit PWM). This divided clock is hereafter called the PWM clock.

As illustrated in Fig.3, the lower-precision (6-bit) PWMs use the least significant part of the 14-bit counter.

Figure 4 shows the circuit diagram of a 6-bit PWM module. Each PWM module has a Special Function Register PWMn; n = 0 to 7. The register format is shown in Table 6.

### 10.1 PWM DAC operation

Value field PVn5 to PVn0 of each PWMn register (n = 0 to 7) is compared to the 6 LSBs of the common counter (14-bit counter).

When the value matches, the output flip-flop is cleared, so that the output pin is driven LOW.

When the value rolls over to zero, the output flip-flop is set, so that the output pin is released. Thus the output waveform has a fixed period of 64 PWM clock cycles; its duty cycle is determined by contents of PWMn.5 to PWMn.0 (PVn5 to PVn0).

Three of the nine total PWM modules (8 PWMn and the 14-bit PWM DAC) operate as previously described; for three others, both the rising and falling edges of the output are delayed by one PWM clock; for the remaining three, both edges are delayed by two PWM clocks. This feature reduces the radio-frequency emission that would otherwise occur when the counter rolled over to zero and all nine open-drain outputs were released.

### 10.2 Special Function Register PWMn (n = 0 to 7)

**Table 6** Special Function Register PWMn (n = 0 to 7; addresses D4H to DFH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| PWnE | – | PVn5 | PVn4 | PVn3 | PVn2 | PVn1 | PVn0 |

**Table 7** Description of PWMn bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | PWnE | PWM module enable bit. If for a particular PWM block (n) the bit:<br>PWnE = 1, then the block is active and controls its assigned port pin.<br>PWnE = 0, the corresponding port pin is controlled by the port. |
| 6 | – | Reserved. |
| 5 to 0 | PVn5 to PVn0 | Value field for PWMn register. |

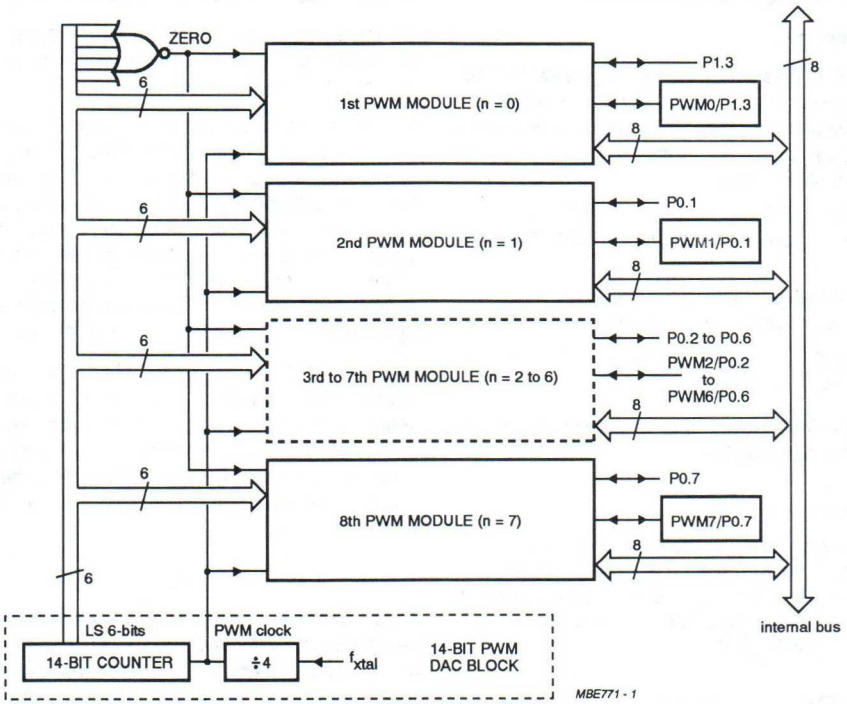
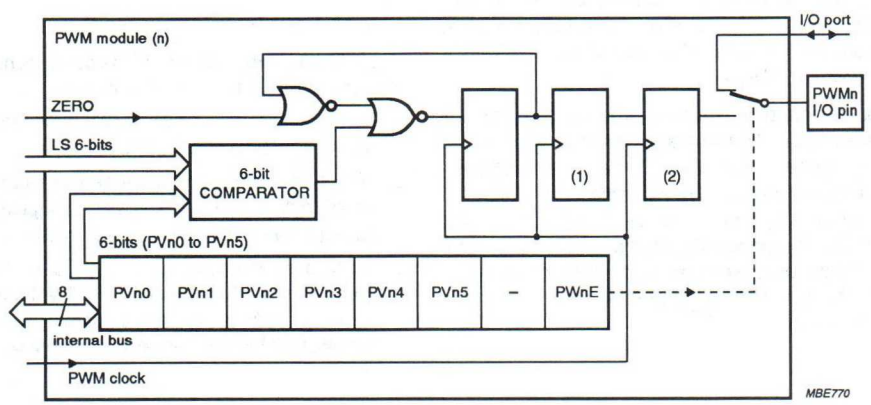



Fig.3 6-bit PWM DAC logic circuit.




(1) This flip-flop occurs in 5 of the 8 PWMn modules.
(2) This flip-flop occurs in 3 of the 8 PWMn modules.

Fig.4 A 6-bit PWM module.

## 11 14-BIT PWM DAC (TDAC)

### 11.1 14-bit counter

The 14-bit counter was already mentioned in Section 10. The nature of the counter is such that it can achieve a stable output value through its MSB, and the value can propagate through logic like that shown in Fig.5. The logic output can be stable within:

- one period of the PWM clock (e.g. 250 ns) if edge-triggered logic is used to capture the logic output, or
- one phase of the PWM clock (e.g. 125 ns) if a phase of the PWM clock is used to capture the logic output.

The 14-bit (TDAC) counter is a ripple counter (cost and die-size reasons).

The 14-bit PWM DAC is controlled by two special function registers TDACL and TDACH.

### 11.2 14-bit DAC operation

When software wishes to change the 14-bit value (TD0 to TD13), it should first write to TDACL and then write to TDACH. Alternatively, if the required precision of the duty cycle is satisfied by 6 bits or less, software can simply write to TDACH (TD8 to TD13).

#### 11.2.1 LOW PRECISION OPERATION

Figure 5 shows that this block includes an 'extra' 14-bit latch between TDACL - TDACH and the comparator and other logic. The programmed value is clocked into the operative latch when the 7 low-order bits of the counter roll over to zero, provided that the software is not in the midst of loading a new 14-bit value, i.e. it is not between writing TDACL and writing TDACH.

In a similar fashion to the lower-precision PWMs, this facility has an output flip-flop that is set when the lower 7 bits of the counter overflow/wrap. The more significant 7 bits of the operative latch's programmed value are compared for equality against the less significant 7 bits of the counter, and the output FF is cleared when they match. Thus this output has a fixed period of 128 PWM clock cycles, and the duty cycle is determined by the programmed value.

#### 11.2.2 HIGH PRECISION OPERATION

For the higher-precision aspect of this feature, the 7 MSBs of the counter are used in a logic block with the 7 LSBs of the programmed value.

The 7$^{th}$ LSB (binary value 64) of the programmed value is ANDed with the 7$^{th}$ MSB (128) of the counter, the 6$^{th}$ LSB of the value is ANDed with the counter's 6$^{th}$ and 7$^{th}$ MSBs being 10, and so on through the LSB of the programmed value being ANDed with the counter's 7 MSBs being 100000. Then these 7 ANDed terms are ORed. If the result is true (logic 1) at the time the 7 LSBs of the counter match the MSBs of the programmed value, the output is forced high for 1 (additional) PWM clock cycle.

The result is that, if the value-64 bit of the 14-bit value is programmed to a logic 1, every other cycle of 128 PWM counter clocks has its duty cycle stretched by one counter clock; if the value-32 bit is programmed to logic 1, every 4$^{th}$ cycle is stretched, and so on through, if the value-1 bit is programmed to logic 1, one cycle out of each 128 is stretched.

#### 11.2.3 14-BIT DAC OUTPUT

Assuming the external integrator can handle all this, the net effect is a PWM DAC that has the period of a 7-bit design (which makes the integrator easier and more feasible to design) with the accuracy of a 14-bit one.

An obvious prerequisite for such precision is that the load on the voltage must be very light, like a single op-amp or comparator.

##### *11.2.3.1 Note*

The TDAC feature differs from the corresponding features of predecessor parts in several ways:

1. The 14-bit value is functionally composed of major and minor portions of 7 bits each.
2. The 14-bit value is programmed as a contiguous multi-register value that can be manipulated straight-forwardly via arithmetic instructions.
3. As discussed for the 6-bit DACs, both of the preceding parts had a feature whereby the PWM output could be inverted, redundantly with complementing the 14-bit value. This feature has been eliminated.

### 11.3 Special Function Register TDACL

**Table 8** Special Function Register TDACL format (address D2H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| TD7 | TD0 | TD1 | TD2 | TD3 | TD4 | TD5 | TD6 |

**Table 9** Description of TDACL bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 to 0 | TD7, TD0 to TD6 | 8 LSBs of the 14-bit value. |

### 11.4 Special Function Register TDACH

**Table 10** Special Function Register TDACH format (address D3H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| TDE | – | TD13 | TD12 | TD11 | TD10 | TD9 | TD8 |

**Table 11** Description of TDACH bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | TDE | Enable bit. |
| 6 | – | Reserved. |
| 5 to 0 | TD13 to TD8 | 6 MSBs of the 14-bit value. |

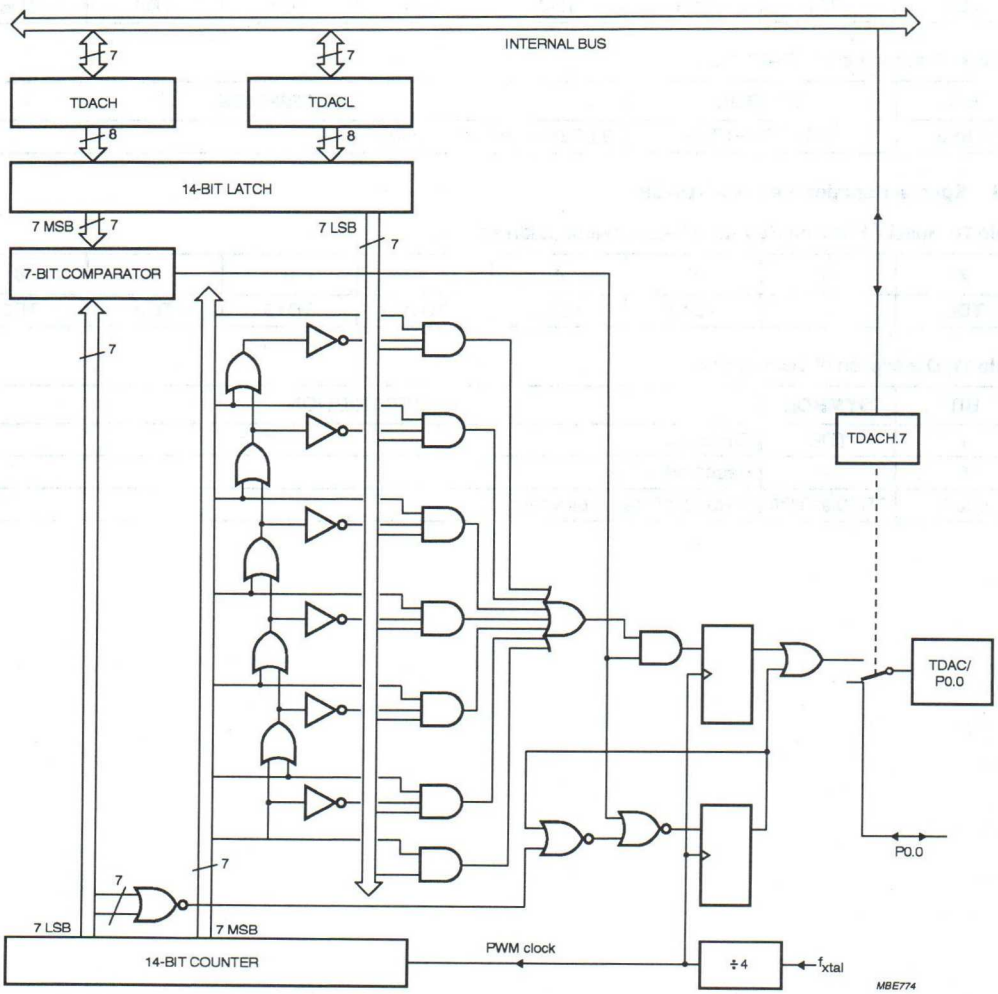



Fig.5 14-bit PWM logic circuit.

## 12 SOFTWARE ANALOG-TO-DIGITAL FACILITY

Figure 6 shows the software analog-to-digital facility block diagram. The block includes Special Function Register SAD.

### 12.1 Special Function Register SAD

**Table 12** Special Function Register SAD format (address D8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| VHi | CH1 | CH0 | St | SAD3 | SAD2 | SAD1 | SAD0 |

**Table 13** Description of SAD bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | VHi | The comparator output bit; bit addressable. |
| 6 | CH1 | The channel field controls which pin, if any, is connected to this facility; see Table 14. |
| 5 | CH0 | |
| 4 | St | The St bit should be written as a logic 1 in order to initiate a voltage comparison. |
| 3 to 0 | SAD3 to SAD0 | 4 LSBs of the SAD register. |

### 12.2 Software ADC operation

Port pins P1.0/ADI0 to P1.2/ADI2 can be alternately selected as inputs of a linear voltage comparator. The other input of the comparator is connected to a 4-bit DAC.

This DAC is controlled by bits SAD3 to SAD0 and produces a reference voltage:

nominally 0.15625 to 4.84375 V in increments of 0.3125 V.

The output of the comparator (HIGH or LOW) can be read by the program as the MSB of the SAD register i.e. bit VHi.

After writing St = 1, the program should include intervening instructions totalling at least 6 machine cycles (72 clock periods or 6 μs at 12 MHz), before the instruction that accesses and tests VHi.

**Table 14** Pin selection: P1.n/ADIn

| CH1 | CH0 | P1.n/ADIn[1] |
|---|---|---|
| 0 | 0 | none |
| 0 | 1 | P1.0/ADI0 |
| 1 | 0 | P1.1/ADI1 |
| 1 | 1 | P1.2/ADI2 |

**Note**

1. Port 1 has open-drain drivers which will not materially affect an analog voltage as long as any and all pins used for software analog-to-digital measurement have corresponding logic 1s in the port register; n = 0, 1, 2.

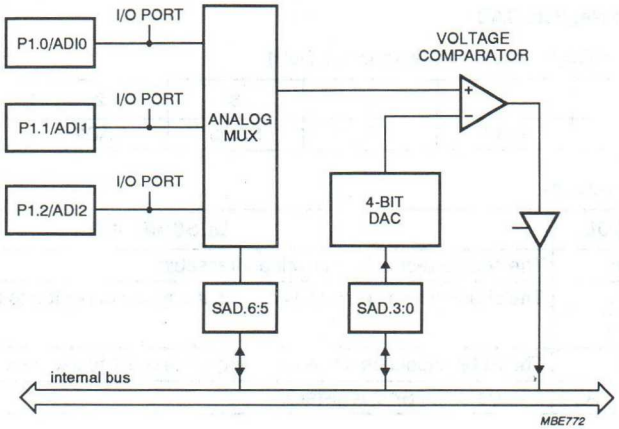



Fig.6 Software analog-to-digital facility.

## 13 ON SCREEN DISPLAY (OSD)

Figure 7 shows the OSD block diagram. It shows the CPU writing into the 128 x 10 display RAM, which is dual-ported to allow the CPU to write into it at any time, including when it is being read out by the OSD logic. The 10-bit wide data coming out of the display RAM is used to access the appropriate character in the Character Generator memory (6-bits) and to specify character and display control functions (4-bits).

Timing for the OSD is controlled by the HSYNC, VSYNC, and dot clock input VCLK1.

### 13.1 OSD features

The 83C055 features an advanced OSD function with some unique features as described in Sections 13.1.1 to 13.1.10.

#### 13.1.1 User-definable display format

The OSD does not restrict the user to a fixed number of lines with a fixed number of characters per line:

- Using a fixed number of lines restricts the generation of displays that can be differentiated from others that use the same chip and places limits on screen content.
- Using a fixed number of characters per line wastes display RAM if a line has less than the full number of displayable characters (it has to be padded with non-visible characters).

The OSD on the 83C055 defines a control character:

- New Line, that has the same function as a Carriage Return and Line Feed.

When the OSD circuitry fetches this character from display RAM it stops displaying further characters, waits for the next horizontal scan line, and starts displaying the next character in display RAM after the New Line character was received.

The number of lines is thus up to the user, within the limits of the display and memory, as are the number of characters per line. This allows far better control of the appearance of the OSD.

#### 13.1.2 Colours selectable by character

Characters can be displayed on a background of the base video or a programmable background colour.
The background colour is selectable by word and the choice of background (base video/user programmed colour) by character.

#### 13.1.3 Dual-Ported Display RAM

The OSD has a true display RAM instead of a character line buffer. This display RAM is dual-ported to allow updating the display RAM at any time instead of having to wait for a vertical retrace.

Vertical Sync (VSYNC) interrupts are supported if flicker-free updates are required.

#### 13.1.4 Programmable character size

- Normal characters are displayed as 18 x 14 bit maps.
- In an interlaced display:
  - 2 fields are displayed so that one actually sees a 36 x 14 pixel size character.
  - The part has a double height and width mode which displays 36 x 28 pixel size bit maps per field.
- For use in non-interlaced systems, the part has a double height mode so that the displayed characters have the same pixel size (36 x 14) as on an interlaced display.

#### 13.1.5 Character shadowing

When characters are displayed overlaid on a background of base video, a black border around the characters makes them highly legible. This feature is called shadowing. The 83C055 has 8 shadowing modes to allow the user to select various partial shadow modes as well as full surround shadow; see Fig.8 and Table 28.

#### 13.1.6 Programmable polarities

Inputs to and outputs from the OSD can be programmed to be recognized as active LOW or HIGH. In conjunction with the 12 V outputs, this allows direct interfacing to most video signal processing circuits.

#### 13.1.7 Character Generator memory in EPROM

On the 87C055, the Character Generator memory is in EPROM. This feature allows quick and inexpensive font development and refinement against the alternative of creating a masked ROM version to see how the final fonts will appear.

#### 13.1.8 HSYNC locked dot clock oscillator

The 83C055 is designed to use an LC oscillator circuit that is started at the trailing edge of HSYNC and stopped at its leading edge. In practice, this gives a highly consistent delay from HSYNC to oscillator start and is stable from scan line to scan line so that no left margin effects are seen.

#### 13.1.9 SHORT ROWS

This mode only displays 4 horizontal lines and is used for generating underlines.

#### 13.1.10 PROGRAMMABLE HORIZONTAL AND VERTICAL POSITIONS

Bit pairs HS4 to HS0 and VS2 to VS0 in register OSORG (Table 30) define the starting point of the display.

### 13.2 General description of the OSD module

This block is the largest of the additions that are specific to this product. Its basic function is to superimpose text on the television video image, to indicate various parameters and settings of the receiver or tuner. External circuitry handles the mixing (multiplexing) of the text and the TV video. The OSD block has 4 input pins:

- Two for a video clock: VCLK1 and VCLK2
- Horizontal sync signal: HSYNC
- Vertical sync signal: VSYNC.

The block has 4 outputs:

- 3 colour video signals
- a control signal.

Since this block is the major feature of the part, its main inputs and outputs are dedicated pins, without alternate port bits. The OSD of the 83C055 differs from that in preceding devices in one major way:

- It does not fix the number and size of displayed rows of text.

Several predecessor parts allowed two displayed rows of 16 characters each. The 83C055 simply has 128 locations of Display RAM, each of which can contain:

- a displayed character, or
- a New Line character that indicates the end of a row. A variant of the New Line character is used to indicate the end of displayed data.

A number of changes in the OSD architecture have reduced the number of other Special Function Registers involved in the feature, below the number needed with predecessor devices:

1. The elimination of certain options such as 4, 6, or 8 x character sizes and alternate use of two of the video outputs.
2. The moving of certain other options from central registers to Display RAM, such as foreground colour codes (Fcolor) and background (B) selection.

Figure 7 shows the 3 major elements of the OSD facility:

- OSD logic
- Display RAM
- Character Generator ROM.

### 13.3 OSD logic

For a standard NTSC TV signal with an HSYNC frequency of 15.750 kHz and a VSYNC frequency of nominally 60 Hz, there are roughly 50 μs of active horizontal scan line available.

A typical pixel clock frequency is 8 MHz, and therefore roughly 400 pixels of resolution can be obtained. At 14 dots per character, this means 28 character per horizontal scan line. If the 12 dot per character display mode is used, that means 33 character per horizontal scan line. Allowing for edge effects, 26 characters (14 across) or 31 characters (12 across) can be displayed.

Note that VGA rates and higher can be used. The minimum character dot size will be a function of the VGA frequency used. For a 640 x 480 display, running at 33 kHz, the equivalent 83C055 pixel resolution is about 160 across (because of the 8 MHz clock and allowing for overscan). This means that status and diagnostic information can be displayed on video monitors.

#### 13.3.1 ON-CHIP VIDEO OSCILLATOR

The video clock pins (VCLK1 and VCLK2) are used to connect a LC circuit to an on-chip video oscillator that is independent of the normal MCU clock.

The L and C values are chosen so that a video pulse, of a duration equal to the VCLK period, will produce a more-or-less square dot on the screen, that is, a dot having a width approximately equal to the vertical distance between consecutive scan lines.

The video oscillator is stopped (with VCLK2 = LOW) while:

- HSYNC (Horizontal Sync) is maintained, and
- is released to operate at the trailing edge of HSYNC.

This technique helps provide uniform horizontal positioning of characters/dots from one scan line to the next.

### 13.4 Character Generator ROM

Character Generator ROM. Containing 60 displayable bit maps, i.e. 64 minus 4, comprising:

- One for each of new line: New Line, and
- Three space characters:
  - Space
  - BSpace
  - SplitBSpace.

Each bit map includes 18 scan lines by 14 dots.

The Character Generator ROM is maskable or programmable along with the Program ROM to allow for various character sets and languages.

### 13.5 Display RAM organization

Each Display RAM location includes:

- 6 data bits, and
- 4 attribute bits.

The 6 data bits from Display RAM, along with a line-within-row count, act as addresses into the Character Generator ROM. Except in special test modes that are beyond the scope of this data sheet, Display RAM cannot be read by the MCU program.

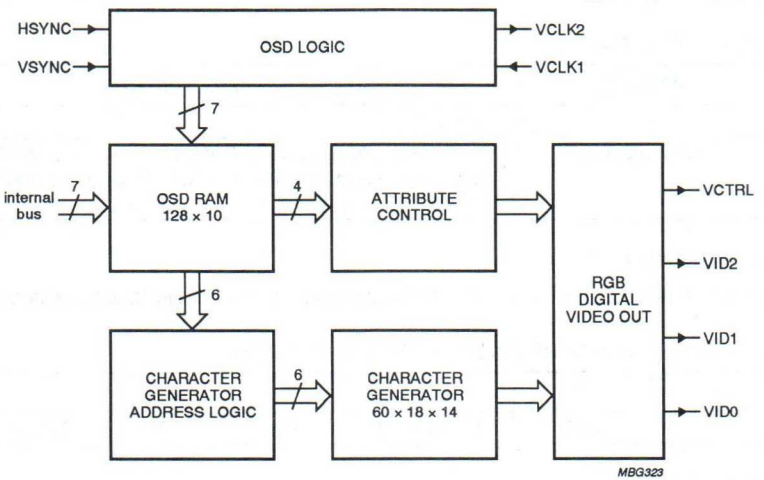



Fig.7 OSD block diagram.

### 13.6 OSD Special Function Registers

The programming interface to Display RAM is provided by three Special Function Registers as shown in Tables 15, 17 and 20.

Writing OSAT simply latches the attribute bits into a register, while writing OSDT causes the data bus information, plus the contents of the OSAT register, to be written into display RAM.

Thus, for a given Display RAM location, OSAT should be written before OSDT. If successive characters are to be written into Display RAM with the same attributes, OSAT need not be rewritten for each character, only prior to writing OSDT for the first character with those particular attributes.

The OSAT attribute bits associated with the BSpace, SplitBSpace and New Line characters (see Table 19) are interpreted differently from those that accompany other data characters. With BSpace and SplitBSpace, B is interpreted as described above, but the 3 colour bits specify the background colour (Bcolor) for subsequent characters. For BSpace, a change in B and Bcolor becomes effective at the left edge of the character's bit map.

#### 13.6.1 Special Function Register OSAD

**Table 15** Special Function Register OSAD (On Screen ADdress; address 9AH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | OSAD6 | OSAD5 | OSAD4 | OSAD3 | OSAD2 | OSAD1 | OSAD0 |

**Table 16** Description of OSAD bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | – | Reserved. |
| 6 to 0 | OSAD6 to OSAD0 | These 7-bits hold the Display RAM address into which data will be loaded. OSAD is automatically incremented by one each time OSDT and Display RAM are written to. |

#### 13.6.2 Special Function Register OSDT

Writing OSDT causes the data bus information, plus the contents of the OSAT register, to be written into display RAM.

**Table 17** Special Function Register OSDT (On Screen DaTa; address 99H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | – | OSDT5 | OSDT4 | OSDT3 | OSDT2 | OSDT1 | OSDT0 |

**Table 18** Description of OSDT bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 to 6 | – | Reserved. |
| 5 to 0 | OSDT5 to OSDT0 | Character data; see Table 19. In reality, there is a potential conflict between the timing of a write to OSDT and an access to display RAM by the OSD logic for data display. This is resolved by the use of a true dual-ported RAM for display memory. |

**Table 19** Special characters related to OSDT register

| SPECIAL CHARACTER | OSDT5 | OSDT4 | OSDT3 | OSDT2 | OSDT1 | OSDT0 |
|---|---|---|---|---|---|---|
| New Line | 1 | 1 | 1 | 1 | 0 | 1 |
| Space (normal) | 1 | 1 | 1 | 1 | 0 | 0 |
| BSpace | 1 | 1 | 1 | 1 | 1 | 0 |
| SplitBspace | 1 | 1 | 1 | 1 | 1 | 1 |

13.6.3 SPECIAL FUNCTION REGISTER OSAT

**Table 20** Special Function Register OSAT ( On Screen ATtributes; address 98H)

| WITH OSDT = | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| New Line | – | – | – | E | – | SR | D | Sh |
| BSpace | – | – | – | B | – | BC2 | BC1 | BC0 |
| SplitBSpace | – | – | – | B | – | BC2 | BC1 | BC0 |
| Any other character | – | – | – | B | – | FC2 | FC1 | FC0 |

**Table 21** Description of OSAT bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 to 5, 3 | – | Reserved. |
| **With OSDT = New Line;** note 1 | | |
| 4 | E | End; If the E bit is 1, no further rows are displayed on the screen. |
| 2 | SR | Short row; If E = 0 and SR = 1, the next row is a 'short row', i.e. it is only 4 or 8 scan lines high rather than 18 or 36. Short rows can be used for underlined text. |
| 1 | D | Double height; If E = 0 and D = 1, all of the characters in the following row are displayed with 'double height and width'. |
| 0 | Sh | Shadowing; If E = 0 and Sh = 1, all of the characters in the following row are displayed with 'shadowing'; see Section 13.8. |
| **With OSDT = BSpace or SplitBspace;** note 2 | | |
| 4 | B | Background; B indicates whether 'background pixels' should show the current background colour (B = 1), or television video (B = 0). |
| 2 to 0 | BC2 to BC0 | Bcolor: Background colour (notes 3 and 4; see Table 22). |
| **With OSDT = Any other character** | | |
| 4 | B | Background; B indicates whether 'background pixels' should show the current background colour (B = 1), or television video (B = 0). |
| 2 to 0 | FC2 to FC0 | Fcolor: Foreground colour. Fcolor indicates the colour of 'foreground pixels' in the ROM bit map for this character (see Table 22). |

**Notes to the description of OSAT bits**

1. The latches in which the E,SR, D, and Sh bits are captured are cleared to zero at the start of each vertical scan. This means that if the first text line on the screen is a short row, or if it contains either double size or shadowing, the text must be preceded by a New Line character. Like all such characters, this initial New Line advances the vertical screen position; the VStart value (see register OSORG; Section 13.9) should take this fact into account.
2. For SplitBSpace, a change in B and Bcolor occurs halfway through the character horizontally.
3. The normal Space character has no effect on the Bcolor value.
4. The Bcolor value is not cleared between vertical scans, so that if a single background colour is all that is needed in an application, it can be set via a single BSpace character during program initialization, and never changed thereafter. In order for such a BSpace to actually affect the 83C055 internal Bcolor register the Mode field of the OSMOD register must be set to '01B' (or higher) so that the OSD hardware is operating (see register OSMOD; Section 13.8).

**Table 22** OSD outputs related to character bit map value, Fcolor, Bcolor and B bits

| CHARACTER BIT MAP VALUE | OSD OUTPUTS (notes 1 and 2) | | | |
|---|---|---|---|---|
| | VID2 | VID1 | VID0 | VCTRL |
| logic 1 | FC2 | FC1 | FC0 | driven active |
| logic 0 | BC2 | BC1 | BC0 | B |

**Notes**

1. Bcolor (BC2,BC1,BC0) values '000' and '111' minimize the occurrence of transient states among the VID2 to VID0 outputs.
2. The background colour defined by the most recently encountered BSpace or SplitBSpace character is maintained on the VID2 to VID0 pins except at the following times:
   a) During the active time of HSYNC.
   b) During the active time of VSYNC.
   c) During those pixels of an active character that correspond to a logic 1 in the character's bit map.
   d) During a 'shadow' bit.

### 13.7 OSD Control Register OSCON

**Table 23** OSD Control Register OSCON (address C0H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| IV | Pv | Lv | Ph | Pc | Po | DH | BFe |

**Table 24** Description of OSCON bits (see note 1)

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | IV | Interrupt flag for the OSD feature. Bit IV is set by the leading edge of the VSYNC pulse, and is cleared by the hardware when the VSYNC interrupt routine is vectored to. It can also be set or cleared by software writing a logic 1 or logic 0 to this bit. |
| 6 | Pv | Pv defines the active VSYNC input polarity. If Pv = 0, then VSYNC input is active HIGH; if Pv = 1, then VSYNC input is active LOW.<br>One effect of bit Pv is that the VID2 to VID0 and VCTRL outputs are blocked (held at black/inactive) during the active time of VSYNC. The IV bit is set on the leading edge of the VSYNC pulse; thus Pv controls whether the OSD interrupt occurs in response to a HIGH-to-LOW or LOW-to-HIGH transition on VSYNC. |
| 5 | Lv | Lv defines the active edge of VSYNC. The active edge (leading or trailing) of VSYNC (as defined by Pv), clears the state counter which determines the vertical start of on screen data. Time reference for the video field is the leading edge of VSYNC, if Lv = 0, or the trailing edge of VSYNC, if Lv = 1. |
| 4 | Ph | Ph defines the active HSYNC input polarity. If Ph = 0, then HSYNC input is active HIGH; if Ph = 1, then HSYNC input is active LOW. |
| 3 | Pc | Pc defines the active VCTRL output polarity; VCTRL output active means: show the colour on VID2 to VID0. If Pc = 0, then VCTRL output is active HIGH; If Pc = 1, then VCTRL output is active LOW. |
| 2 | Po | Po defines the VID2 to VID0 outputs polarity; bit is needed only because the Shadowing feature needs to generate black pixels without reference to a register value. Internally, the 3-bit code '000B' always designates black.<br>If Po = 0, a logic 0 internal to the 83C055 corresponds to a LOW on one of the VID2 to VID0 pins.<br>If Po = 1, a logic 1 internal to the 83C055 corresponds to a LOW on one of the VID2 to VID0 pins. |
| 1 | DH | If DH = 1, character sizes are doubled vertically but not horizontally. This feature allows the 83C055 to be used in 'improved definition' systems that are not interlaced.<br>The vertical doubling imposed by DH does not affect the VStart logic as described in Table 30; it operates in HSync units regardless of DH or D. |
| 0 | BFe | Background/Foreground enable; output BF. If BFe = 1, then the BF output tracks whether each bit in displayed characters is a Foreground bit (LOW), or a Background bit (HIGH). If BFe = 0, then the BF pin remains HIGH. |

**Note**

1. It is theoretically possible that a VSYNC interrupt could be missed, or an extra one generated, if OSCON is read, then modified internally (e.g. in ACC), and the result written back to OSCON. However, none of the other bits in OSCON are reasonable candidates for dynamic change. Special provisions are included in the 83C055 logic so that IV will not be changed by a single 'read-modify-write' instruction such as SETB or CLR, unless the instruction specifically changes IV.

### 13.8 OSD Control Register OSMOD

Under some conditions writing to OSMOD while the display is active can cause a temporary flicker during that display field. This can be avoided by only writing to OSMOD during the vertical sync interval.

**Table 25** OSD Control Register OSMOD (address C1H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| Wc | – | Mode1 | Mode0 | – | SHM2 | SHM1 | SHM0 |

**Table 26** Description of OSMOD bits (see note )

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | Wc | If Wc = 1, then each displayed character is horizontally terminated after 12 bits have been output, as opposed to after 14 bits if Wc = 0. This allows text to be 'packed' more tightly so that more characters can be displayed per line. In effect, the 2 bits out of the display ROM, which would otherwise be the rightmost 2 of the 14, are ignored when Wc is 1. Clearly, if this feature is to be used, it must be accounted for in the design of the bit maps in the display ROM. |
| 6 | – | Reserved. |
| 5 | Mode1 | Display mode select bits; see Table 27. |
| 4 | Mode0 | |
| 3 | – | Reserved. |
| 2 to 0 | SHM2 to SHM0 | Shadowing mode (ShMode); determines how characters are shadowed in rows for which the row attribute Sh = 1 (register OSAT; see Table 21); for the shadowing modes see Fig.8 and Table 28. |

**Table 27** Selection of Display Modes

| Mode1 | Mode0 | DISPLAY MODE |
|---|---|---|
| 0 | 0 | Mode 0 The OSD feature is disabled. VCLK oscillator is disabled, VID2 to VID0 are set to black, and VCTRL is held inactive.This is the mode to which the 83C055 OSD logic is reset; note 1. |
| 0 | 1 | Mode 1 The VCLK oscillator is enabled and the OSD logic operates normally internally, but VID2 to VID0 are set to black and VCTRL is held inactive; note 2. |
| 1 | 0 | Mode 2 Normal OSD operation. Active characters can be shown against TV video (for characters with B = 0) or (for characters with B = 1) against a background of the colour defined as an attribute of BSpace and SplitBSpace characters. |
| 1 | 1 | Mode 3 Characters can be displayed but all of the receiver's normal video is inhibited by holding VCTRL asserted throughout the active portion of each scan line; see note 3. |

**Notes**

1. A direct transition from this mode to 'active display' (Mode1, Mode0 = 1X) would result in undefined operation and visual effects for the duration of the current video field (until the next VSYNC).
2. The OSD feature can be toggled between this state and 'active display' as desired to achieve real-time special effects such as 'vertical wiping'.
3. Since VID2 to VID0 are driven with the current background colour during this time, except during the foreground portion of displayed characters, this produces text against a solid background. This mode is useful for extensive displays that require user concentration.

**Table 28** Shadowing modes determined by bits SHM2 to SHM0 (register OSMOD) and Sh (register OSAT)

| SHM2 | SHM1 | SHM0 | Sh | SHADOWING MODE[1] |
|---|---|---|---|---|
| 0 | 0 | 0 | 1 | South-west |
| 0 | 0 | 1 | 1 | West |
| 0 | 1 | 0 | 1 | North-west |
| 0 | 1 | 1 | 1 | North |
| 1 | 0 | 0 | 1 | North-east |
| 1 | 0 | 1 | 1 | East |
| 1 | 1 | 0 | 1 | South-east |
| 1 | 1 | 1 | 1 | Full surround |
| X | X | X | 0 | No Shadowing |

**Note**

1. The mode names are based on the position of an apparent light source, ranging from the lower left (South-west) clockwise to the lower right (South-east); see Fig.8.

### 13.9 OSD Control Register OSORG

**Table 29** OSD Control Register OSORG (address C2H)

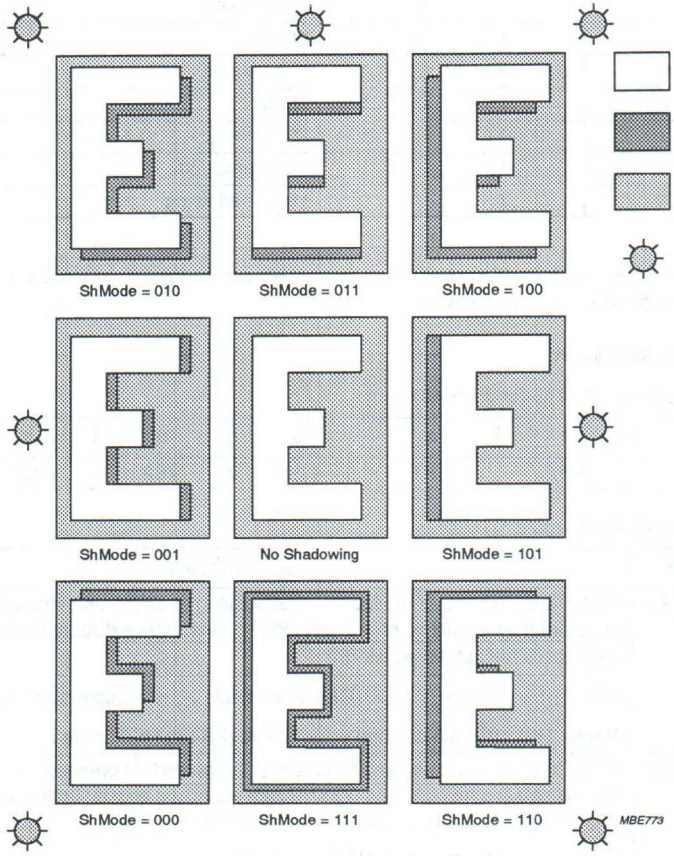
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| HS4 | HS3 | HS2 | HS1 | HS0 | VS2 | VS1 | VS0 |

**Table 30** Description of OSORG bits (note 1)

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 to 3 | HS4 to HS0 | HStart field; defines the horizontal start position of all the on-screen character rows, as approximately a multiple of 4 VCLK clock cycles. Active display begins after the trailing edge of HSYNC at the position:<br>HP = [4 × (HStart) + 1] × VCLK clock cycle + (one single-sized character width)<br>Where (HStart) is the decimal value of bits (HS4 to HS0); note 2. |
| 2 to 0 | VS2 to VS0 | VStart field; defines the vertical start position of the first on-screen character row, as approximately a multiple of 4 HSYNC pulses. Active display begins after the field's time reference point (a range of 3 to 31)at the position:<br>VP = [4 × (VStart) − 1] × HSYNC pulses<br>Where (VStart) is the decimal value of bits (VS2 to VS0); note 3. |

**Notes**

1. Neither the Hstart nor Vstart parameter is affected by the D line attribute that is used to display double-sized characters.
2. Counting variations in Wc, there may be 17 to 143 VCLK clock cycles from the end of HSYNC to the start of the first character of each row.
3. Subsequent character rows occur directly below the first, such that the last scan line of one row is directly followed by the first scan line of the next row. Successive New Line characters (with or without the Short Row designation) can be used to vertically separate text rows on the screen.




ShMode = (SHM2, SHM1, SHM0)

Fig.8 Effect of shadowing on the letter 'E'.

## 14 PROGRAMMING CONSIDERATIONS

### 14.1 EPROM Characteristics

The 87C055 is programmed by using a modified Quick-Pulse Programming algorithm similar to that used for devices such as the 87C751. It differs from these devices in that a serial data stream is used to place the 87C055 in the programming mode.

Figure 9 shows a block diagram of the programming configuration for the 87C055.

**Table 31** Pin usage for Programming

| PIN | USAGE |
|---|---|
| XTAL1 | Oscillator input and receives the master system clock. This clock should be between 1.2 and 6 MHz. |
| RESET | Used to accept the serial data stream that places the 87C055 into various programming modes. This pattern consists of a 10-bit code with the LSB sent first. Each bit is synchronized to the clock input, XTAL1. |
| **Port 0** | |
| $V_{PP}$/TDAC/P0.0 | Used as the programming voltage supply input ($V_{PP}$ signal). |
| PROG/PWM1/P0.1 | Used as the program PROG signal. This pin is used for the 25 programming pulses. |
| **Port 2** | |
| P2.7 to P2.0 | Address input for the byte to be programmed and accepts both the high- and low-order components of the 11-bit address; note 1. |
| **Port 3** | |
| P3.7 to P3.0 | Used as a bidirectional data bus during programming and verify operations. During programming mode, it accepts the byte to be programmed. During verify mode, it provides the contents of the EPROM location specified by the address which has been supplied to Port 2. |

**Note**

1. Multiplexing of these address components is performed using the ASEL input:
   a) ASEL input is driven HIGH and then drive Port 2 with the high-order bits of the address. ASEL should remain HIGH for at least 13 clock cycles.
   b) ASEL may then be driven LOW which latches the high-order bits of the address internally. The high-order address should remain on Port 2 for at least 2 clock cycles after ASEL is driven LOW.
   c) Port 2 may then be driven with the low byte of the address. The low-order address will be internally stable 13 clock cycles later. The address will remain stable provided that the low byte placed on Port 2 is held stable and ASEL is kept LOW.
   d) ASEL needs to be pulsed HIGH only to change the high byte of the address.

### 14.2 Programming operation

Figures 10 and 11 show the timing diagrams for the Program/Verify cycle. Programming operation:

1. RST should initially be held HIGH for at least 2 machine cycles. P0.1 (PROG) and P0.0 ($V_{PP}$) will be at $V_{OH}$ as a result of the RST operation. At this point, these pins function as normal quasi-bidirectional I/O ports and the programming equipment may pull these lines LOW. However, prior to sending the 10-bit code on the RST pin, the programming equipment should drive these pins HIGH ($V_{IH}$).
2. The RST pin may now be used as the serial data input for the data stream which places the 87C055 in the Programming Mode. Data bits are sampled during the clock HIGH time and thus should only change during the time that the clock is LOW. Following transmission of the last data bit, the RST pin should be held LOW.
3. Next the address information for the location to be programmed is placed on Port 2 and ASEL is used to perform the address multiplexing, as previously described (see Table 31; note 1).
   a) At this time, Port 1 functions as an output.
   b) A high voltage $V_{PP}$ level is then applied to the $V_{PP}$ input (P0.0). This sets Port 1 as an input port.
   c) The data to be programmed into the EPROM array is then placed on Port 3. This is followed by a series of programming pulses applied to the PROG pin (P0.1). These pulses are created by driving P0.1 LOW and then HIGH. This pulse is repeated until a total of 25 programming pulses have occurred. At the conclusion of the last pulse, the PROG signal should remain HIGH.
4. The $V_{PP}$ signal may now be driven to the $V_{OH}$ level, placing the 87C055 in the Verify Mode; Port 3 is now used as an output port. After four machine cycles (48 clock periods), the contents of the addressed location in the EPROM array will appear on Port 3.
5. The next programming cycle may now be initiated by:
   a) Placing the address information at the inputs of the multiplexed buffers.
   b) Driving the $V_{PP}$ pin to the $V_{PP}$ voltage level.
   c) Providing the byte to be programmed to Port 3 and issuing the 26 programming pulses on the PROG pin.
   d) Bringing $V_{PP}$ back down to the $V_{OH}$ level and verifying the byte (see Table 33).

### 14.3 Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4000 Angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure.

For this and secondary effects, it is recommended that an opaque label be placed over the window. For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorless (part number 2345-5) or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 Angstroms) to an integrated dose of at least 15 $Ws/cm^2$.

Exposing the EPROM to an ultraviolet lamp of 12000 $\mu W/cm^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient. Erasure leaves the array in an all logic 1s state.

### 14.4 Reading Signature Bytes

The Signature Bytes are read by the same procedure as a normal verify of locations 30H and 31H (the values are shown in Table 32), except that the serial code indicated in Table 33 for reading signature bytes should be used.

**Table 32** Programming and Verification codes

| ADDRESS | CONTENT | INDICATION |
|---|---|---|
| 30H | 15H | manufactured by Philips |
| 31H | 4BH | 87C055 |

**Table 33** Implementing Program/Verify Modes

| OPERATION | SERIAL CODE | P0.1 (PROG) | P0.0 ($V_{PP}$) |
|---|---|---|---|
| Program user EPROM | 286H | –[1] | $V_{PP}$ |
| Verify user EPROM | 286H | $V_{IH}$ | $V_{IH}$ |
| Read Signature Bytes | 280H | $V_{IH}$ | $V_{IH}$ |

**Note**

1. Pulsed from $V_{IH}$ to $V_{IL}$ and returned to $V_{IH}$.

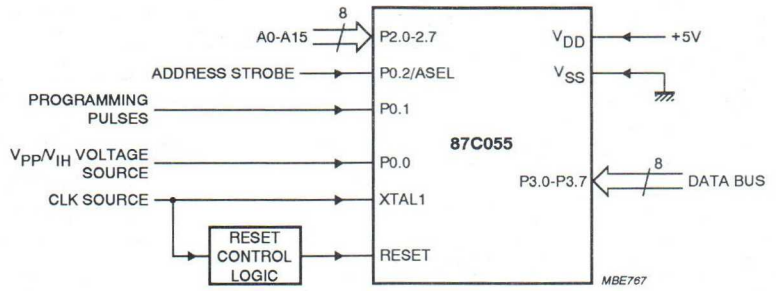



Fig.9 Programming Configuration.

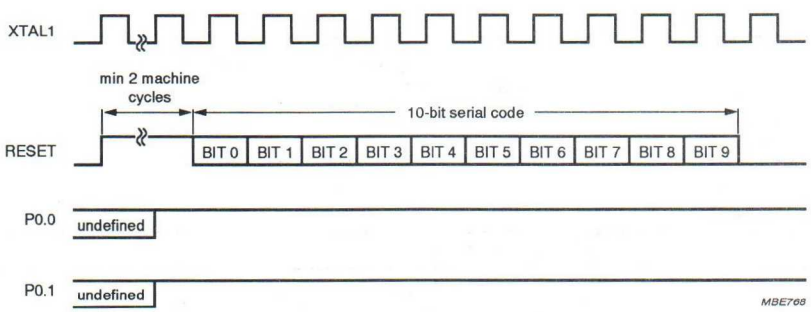



Fig.10 Entry into Program/Verify Modes.

### 14.5 EPROM Programming and Verification

$V_{DD}$ = 5 V ±10%; $V_{SS}$ = 0 V; $T_{amb}$ = 21 to 27 °C.

| SYMBOL | PARAMETER | MIN. | MAX. | UNIT |
|---|---|---|---|---|
| $1/t_{CLCL}$ | Oscillator/clock frequency | 1.2 | 6 | MHz |
| $t_{AVGL}$[1] | Address setup to P0.1 (PROG) LOW | $10 + 24t_{CLCL}$ | – | µs |
| $t_{GHAX}$ | Address hold after P0.1 (PROG) HIGH | $48t_{CLCL}$ | – | µs |
| $t_{DVGL}$ | Data setup to P0.1 (PROG) LOW | $38t_{CLCL}$ | – | µs |
| $t_{GHDX}$ | Data hold after P0.1 (PROG) HIGH | $36t_{CLCL}$ | – | µs |
| $t_{SHGL}$ | $V_{PP}$ setup to P0.1 (PROG) LOW | 10 | – | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after P0.1 (PROG) HIGH | 10 | – | µs |
| $t_{GLGH}$ | P0.1 (PROG) width | 90 | 110 | µs |
| $t_{AVQV}$[1] | $V_{PP}$ ($V_{DD}$) LOW to data valid | – | $48t_{CLCL}$ | µs |
| $t_{GHGL}$ | P0.1 (PROG) HIGH to P0.1 (PROG) LOW | 10 | – | µs |
| $t_{SYNL}$ | P0.0 (sync pulse) LOW | $4t_{CLCL}$ | – | µs |
| $t_{SYNH}$ | P0.0 (sync pulse) HIGH | $8t_{CLCL}$ | – | µs |
| $t_{MASEL}$ | ASEL HIGH time | $13t_{CLCL}$ | – | µs |
| $t_{HAHLD}$ | Address hold time | $2t_{CLCL}$ | – | µs |
| $t_{HASET}$ | Address setup to ASEL | $13t_{CLCL}$ | – | µs |
| $t_{ADSTA}$ | Low address to address stable | $13t_{CLCL}$ | – | µs |

**Note**

1. Address should be valid at least $24t_{CLCL}$ before the rising edge of P0.0 ($V_{PP}$).

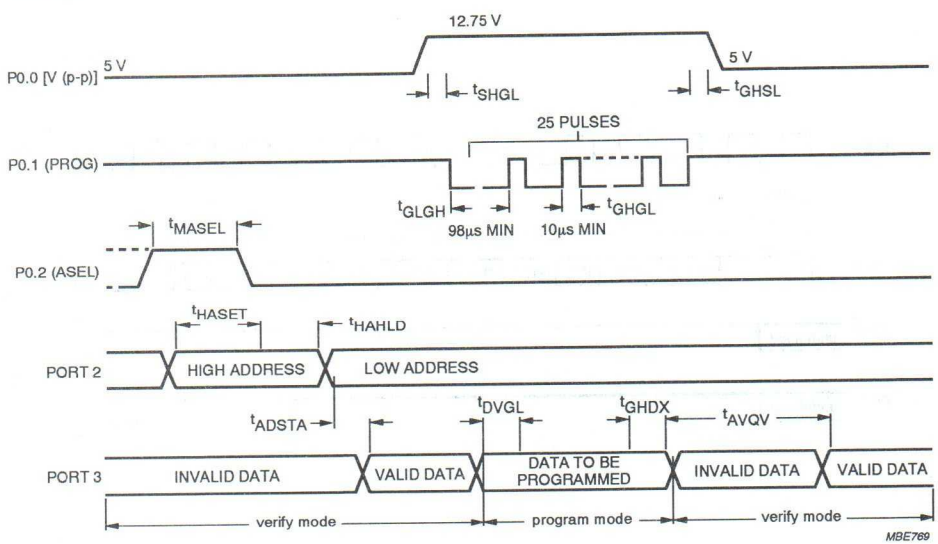



Fig.11 Program/Verify cycle.

## 15 PROGRAMMING THE OSD EPROM

### 15.1 Overview

The OSD EPROM space starts at location C000H and ends at location CFFFH. However, due to the addressing scheme of the OSD, not all locations within this space are used.The start location of the next character can be calculated by adding 40H to the start location of the previous character. For example, character 1 starts at C000H; then characters 2, 3, and 4 start at C040H, C080H, and C0C0H, respectively.

### 15.2 Character description and programming

An example of an OSD character bit map, and the program data to obtain that character is shown in Table 34.

Each character is 14 bits wide by 18 lines high.A character is split about a vertical axis into two sections UPPER and LOWER as illustrated in Table 34:

- Each section contains 7 bits of the character, such that:
  - the LOWER section contains bits 7 to 1, and
  - the UPPER section contains bits 14 to 8.
- The LOWER section of the character is programmed when the LSB of the program address equals a logic 0, and the UPPER section when the LSB equals a logic 1.

During Programming and Verification, each section is programmed using bytes of program data. The MSB of the program data is not used; however, the MSB location physically exists, and so will Program and Verify.

### 15.3 OSD EPROM bit map

The mapping for the full OSD EPROM is shown in Table 35. To program the example character into the first character location of the OSD EPROM would require the data at the address as shown in Table 34.

**Table 34** Example of an OSD Character Bit Map (note 1)

| LINE | CHARACTER BIT MAP | | PROGRAM DATA | | ADDRESS (HEX) | |
|---|---|---|---|---|---|---|
| | UPPER (BIT 14 TO 8) | LOWER (BIT 7 TO 1) | UPPER | LOWER | UPPER | LOWER |
| Line 1 | 0000000 | 0000000 | X0000000 | X0000000 | C001 | C000 |
| Line 2 | 0000000 | 0000000 | X0000000 | X0000000 | C003 | C002 |
| Line 3 | 0011110 | 0001100 | X0011110 | X0001100 | C005 | C004 |
| Line 4 | 0011110 | 0001100 | X0011110 | X0001100 | C007 | C006 |
| Line 5 | 0011110 | 0001100 | X0011110 | X0001100 | C009 | C008 |
| Line 6 | 0011110 | 0001100 | X0011110 | X0001100 | C00B | C00A |
| Line 7 | 0011110 | 0001100 | X0011110 | X0001100 | C00D | C00C |
| Line 8 | 0011110 | 0001100 | X0011110 | X0001100 | C00F | C00E |
| Line 9 | 0011111 | 1111100 | X0011111 | X1111100 | C011 | C010 |
| Line 10 | 0011111 | 1111100 | X0011111 | X1111100 | C013 | C012 |
| Line 11 | 0011111 | 1111100 | X0011111 | X1111100 | C015 | C014 |
| Line 12 | 0011110 | 0001100 | X0011110 | X0001100 | C017 | C016 |
| Line 13 | 0011110 | 0001100 | X0011110 | X0001100 | C019 | C018 |
| Line 14 | 0011110 | 0001100 | X0011110 | X0001100 | C01B | C01A |
| Line 15 | 0011110 | 0001100 | X0011110 | X0001100 | C01D | C01C |
| Line 16 | 0011110 | 0001100 | X0011110 | X0001100 | C01F | C01E |
| Line 17 | 0000000 | 0000000 | X0000000 | X0000000 | C021 | C020 |
| Line 18 | 0000000 | 0000000 | X0000000 | X0000000 | C023 | C022 |

**Note**

1. X can be a logic 0 or logic 1, and will Program and Verify correctly.

**Table 35** OSD EPROM Bit Map

| CHARACTER NO. | ADDRESS (HEX) | | CHARACTER LINE NO. |
|---|---|---|---|
| | LOWER BYTE | UPPER BYTE | |
| 0 | C000 | C001 | 1 |
| | C002 | C003 | 2 |
| | C004 | C005 | 3 |
| | C006 | C007 | 4 |
| | C008 | C009 | 5 |
| | C00A | C00B | 6 |
| | C00C | C00D | 7 |
| | C00E | C00F | 8 |
| | C010 | C011 | 9 |
| | C012 | C013 | 10 |
| | C014 | C015 | 11 |
| | C016 | C017 | 12 |
| | C018 | C019 | 13 |
| | C01A | C01B | 14 |
| | C01C | C01D | 15 |
| | C01E | C01F | 16 |
| | C020 | C021 | 17 |
| | C022 | C023 | 18 |
| | C024 to C03F | | not used |
| 1[1] | C040 to C063 | | 1 to 18 |
| | C064 to C07F | | not used |
| 2[1] | C080 to C0A3 | | 1 to 18 |
| | C0A4 to C0BF | | not used |
| 3 to 59[1] | – | | – |
| 60[2] | CF00 to CF23 | | 1 to 18 |
| | CF24 to CF3F | | not used |
| 61[2] | CF40 to CF63 | | 1 to 18 |
| | CF64 to CF7F | | not used |
| 62[2] | CF80 to CFA3 | | 1 to 18 |
| | CFA4 to CFBF | | not used |
| 63[2] | CFC0 to CFE3 | | 1 to 18 |
| | CFE4 to CFFF | | not used |

**Notes**

1. Characters 1 to 59 are setup in the similar way as character 0; due to space and simplicity this is not fully displayed.
2. Locations 60, 61, 62 and 63 should be programmed to logic 0s. The character names are: character no. 60 = Normal Space; character no. 61 = New Line; character no. 62 = BSpace; character no. 63 = SplitBSpace.

## 16 REGISTER MAP

**Table 36** Register map

Values within parenthesis show the bit state after a reset operation; 'X' denotes an undefined state.

| ADDR. (HEX) | REGISTER | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| E0 | ACC[1] | ACC7 (0) | ACC6 (0) | ACC5 (0) | ACC4 (0) | ACC3 (0) | ACC2 (0) | ACC1 (0) | ACC0 (0) |
| F0 | B[1] | B7 (0) | B6 (0) | B5 (0) | B4 (0) | B3 (0) | B2 (0) | B1 (0) | B0 (0) |
| 83 | DPH | DPH7 (0) | DPH6 (0) | DPH5 (0) | DPH4 (0) | DPH3 (0) | DPH2 (0) | DPH1 (0) | DPH0 (0) |
| 82 | DPL | DPL7 (0) | DPL6 (0) | DPL5 (0) | DPL4 (0) | DPL3 (0) | DPL2 (0) | DPL1 (0) | DPL0 (0) |
| A8 | IE[1] | EA (0) | – (X) | – (0) | EVS (0) | ET1 (0) | EX1 (0) | ET0 (0) | EX0 (0) |
| 9A | OSAD | – (X) | OSAD6 (X) | OSAD5 (X) | OSAD4 (X) | OSAD3 (X) | OSAD2 (X) | OSAD1 (X) | OSAD0 (X) |
| 9F to 98 | OSAT[1][2] | – (X) | – (X) | – (X) | E (X) | – (X) | SR (X) | D (X) | Sh (X) |
| | OSAT[1][3] | – (X) | – (X) | – (X) | B (X) | – (X) | BC2 (X) | BC1 (X) | BC0 (X) |
| | OSAT[1][4] | – (X) | – (X) | – (X) | B (X) | – (X) | FC2 (X) | FC1 (X) | FC0 (X) |
| 99 | OSDT | – (X) | – (X) | OSDT5 (X) | OSDT4 (X) | OSDT3 (X) | OSDT2 (X) | OSDT1 (X) | OSDT0 (X) |
| C0 | OSCON[1] | IV (X) | Pv (X) | Lv (X) | Ph (X) | Pc (X) | Po (X) | DH (X) | BFe (X) |
| C1 | OSMOD | Wc (X) | – (X) | Mode1 (X) | Mode0 (X) | – (X) | SHM2 (X) | SHM1 (X) | SHM0 (X) |
| C2 | OSORG | HS4 (X) | HS3 (X) | HS2 (X) | HS1 (X) | HS0 (X) | VS2 (X) | VS1 (X) | VS0 (X) |
| 80 | P0[1] | P07 (1) | P06 (1) | P05 (1) | P04 (1) | P03 (1) | P02 (1) | P01 (1) | P00 (1) |
| 90 | P1[1] | P17 (1) | P16 (1) | P15 (1) | P14 (1) | P13 (1) | P12 (1) | P11 (1) | P10 (1) |
| A0 | P2[1] | P27 (1) | P26 (1) | P25 (1) | P24 (1) | P23 (1) | P22 (1) | P21 (1) | P20 (1) |
| B0 | P3[1] | P37 (1) | P36 (1) | P35 (1) | P34 (1) | P33 (1) | P32 (1) | P31 (1) | P30 (1) |
| 87 | PCON | – (0) | – (X) | – (X) | – (X) | GF1 (X) | GF0 (X) | – (X) | – (X) |
| D0 | PSW[1] | CY (0) | AC (0) | F0 (0) | RS1 (0) | RS0 (0) | OV (0) | – (0) | P (0) |
| D4 | PWM0 | PW0E (0) | – (0) | PV05 (0) | PV04 (0) | PV03 (0) | PV02 (0) | PV01 (0) | PV00 (0) |

| ADDR. (HEX) | REGISTER | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| D5 | PWM1 | PW1E (0) | – (0) | PV15 (0) | PV14 (0) | PV13 (0) | PV12 (0) | PV11 (0) | PV10 (0) |
| D6 | PWM2 | PW2E (0) | – (0) | PV25 (0) | PV24 (0) | PV23 (0) | PV22 (0) | PV21 (0) | PV20 (0) |
| D7 | PWM3 | PW3E (0) | – (0) | PV35 (0) | PV34 (0) | PV33 (0) | PV32 (0) | PV31 (0) | PV30 (0) |
| DC | PWM4 | PW4E (0) | – (0) | PV45 (0) | PV44 (0) | PV43 (0) | PV42 (0) | PV41 (0) | PV40 (0) |
| DD | PWM5 | PW5E (0) | – (0) | PV55 (0) | PV54 (0) | PV53 (0) | PV52 (0) | PV51 (0) | PV50 (0) |
| DE | PWM6 | PW6E (0) | – (0) | PV65 (0) | PV64 (0) | PV63 (0) | PV62 (0) | PV61 (0) | PV60 (0) |
| DF | PWM7 | PW7E (0) | – (0) | PV75 (0) | PV74 (0) | PV73 (0) | PV72 (0) | PV71 (0) | PV70 (0) |
| D8 | SAD(1) | VHi (0) | CH1 (0) | CH0 (0) | St (0) | SAD3 (0) | SAD2 (0) | SAD1 (0) | SAD0 (0) |
| 81 | SP | SP7 (0) | SP6 (0) | SP5 (0) | SP4 (0) | SP3 (0) | SP2 (0) | SP1 (0) | SP0 (0) |
| D3 | TDACH | TDE (0) | – (0) | TD13 (0) | TD12 (0) | TD11 (0) | TD10 (0) | TD9 (0) | TD8 (0) |
| D2 | TDACL | TD7 (0) | TD0 (0) | TD1 (0) | TD2 (0) | TD3 (0) | TD4 (0) | TD5 (0) | TD6 (0) |
| 88 | TCON(1) | TF1 (0) | TR1 (0) | TF0 (0) | TR0 (0) | IE1 (0) | IT1 (0) | IE0 (0) | IT0 (0) |
| 8C | TH0 | TH07 (0) | TH06 (0) | TH05 (0) | TH04 (0) | TH03 (0) | TH02 (0) | TH01 (0) | TH00 (0) |
| 8D | TH1 | TH17 (0) | TH16 (0) | TH15 (0) | TH14 (0) | TH13 (0) | TH12 (0) | TH11 (0) | TH10 (0) |
| 8A | TL0 | TL07 (0) | TL06 (0) | TL05 (0) | TL04 (0) | TL03 (0) | TL02 (0) | TL01 (0) | TL00 (0) |
| 8B | TL1 | TL17 (0) | TL16 (0) | TL15 (0) | TL14 (0) | TL13 (0) | TL12 (0) | TL11 (0) | TL10 (0) |
| 89 | TMOD | GATE (0) | C/T (0) | M1 (0) | M0 (0) | GATE (0) | C/T (0) | M1 (0) | M0 (0) |
| C3 | RAMCHR | for test purposes only | | | | | | | |
| C4 | RAMATT | for test purposes only | | | | | | | |

**Notes**

1. Bit addressable.
2. With OSDT = New Line.
3. With OSDT = BSpace or SplitBSpace.
4. With OSDT = Any other character.

## 17 LIMITING VALUES

In accordance with the Absolute Maximum Rating System (IEC 34); see notes 1 and 2.

| SYMBOL | PARAMETER | MIN. | MAX. | UNIT |
|---|---|---|---|---|
| $V_{DD}$ | supply voltage | 4.5 | 5.5 | V |
| $V_I$ | input voltage on any pin with respect to ground ($V_{SS}$) | –0.5 | 6.5 | V |
| $I_{OH}$ | maximum source current for all port lines | – | –1.5 | mA |
| $I_{OL}$ | maximum sink current for all port lines | – | 15 | mA |
| $P_{tot}$ | total power dissipation | – | 1.5 | W |
| $T_{amb}$ | operating ambient temperature | 0 | 70 | °C |
| $T_{stg}$ | storage temperature | –65 | 150 | °C |

**Notes**

1. Stresses above those listed under Limiting Values may cause permanent damage to the device.
2. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## 18 HANDLING

Inputs and outputs are protected against electrostatic discharge in normal handling. However it is good practice to take normal precautions appropriate to handling MOS devices (see *"Handling MOS devices"*).

## 19 DC CHARACTERISTICS

$V_{DD}$ = 5 V ±10% $T_{amb}$ = 0 to +70 °C; all voltages with respect to $V_{SS}$; unless otherwise specified.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|---|
| **Supply** | | | | | | |
| $V_{DD}$ | operating supply voltage | | 4.5 | 5.0 | 5.5 | V |
| $I_{DD}$ | operating supply current | $V_{DD}$ = 5.5 V; note 1 | – | – | 30 | mA |
| $V_{IL}$ | LOW level input voltage | | −0.5 | – | $0.2V_{DD}$ − 0.1 | V |
| $V_{IL1}$ | LOW level input voltage; VSYNC and HSYNC | | −0.5 | – | $0.15V_{DD}$ | V |
| $V_{IH}$ | HIGH level input voltage; XTAL, VCLK1 and RST | | $0.7V_{DD}$ | – | $V_{DD}$ + 0.5 | V |
| $V_{IH1}$ | HIGH level input voltage; P1.2 to P1.0, P3.6 to P3.5 and P3.3 to P3.1 | | $0.2V_{DD}$ + 0.9 | – | $V_{DD}$ + 0.5 | V |
| $V_{IH2}$ | HIGH level input voltage; P1.3, P3.7,P3.4 and P3.0 | | $0.2V_{DD}$ + 0.9 | – | 12.6 | V |
| $V_{IH3}$ | HIGH level input voltage; VSYNC and HSYNC | | $0.67V_{DD}$ | – | $V_{DD}$ + 0.5 | V |
| $V_{IH} - V_{DD}$ | HIGH level input voltage with respect to $V_{DD}$; Port 0, P1.3, P3.7, P3.4 and P3.0 | note 2 | $0.7V_{DD}$ | – | $V_{DD}$ + 0.5 | V |
| $V_{OL1}$ | LOW level output voltage; P2.7 to P2.0 and P3.6 to P3.5 | $I_{OL}$ = 10 mA; note 3 | – | – | 0.5 | V |
| $V_{OL2}$ | LOW level output voltage; TDAC and PWM0 to PWM7 | $I_{OL}$ = 700 μA; note 4 | – | – | 0.5 | V |
| $V_{OL3}$ | LOW level output voltage; all other outputs | $I_{OL}$ = 1.6 mA | – | – | 0.45 | V |
| $V_{OH}$ | HIGH level output voltage; Port 1, VID2 to VID0, VCTRL and BF | $I_{OH}$ = −60 μA | 2.4 | – | – | V |
| $R_{RST}$ | Reset (RST) pull-down resistor | | 50 | – | 300 | kΩ |
| $C_{IO}$ | Pin capacitance; except P0.0 and P0.7 | test freq. = 1 MHz; $T_{amb}$ = 25 °C; note 5 | – | – | 10 | pF |
| HYS | Hysteresis; VSYNC and HSYNC | | 0.8 | – | – | V |

**Notes**

1. $I_{DD}$ measured with OSD block initialized and RST remaining LOW.
2. This maximum applies at all times, including during power switching, and must be accounted for in power supply design. During a Power-on process, the +12 V source used for external pull-up resistors should not precede the $V_{DD}$ of the 83C055 up their respective voltage ramps by more than this margin, nor, during a Power-down process, should $V_{DD}$ precede +12 V down their respective voltage ramps by more than this margin.
3. No more than 6 (any 6) of these 10 high current outputs may be used at the $V_{OL1}$ ($I_{OL}$ = 10 mA) specification. The other 4 should comply with the $V_{OL3}$ specification ($I_{OL}$ = 1.6 mA).
4. The specified current rating applies when any of these pins is used as a Pulse Width Modulated (PWM) output. For use as a port output, the rating is as given subsequently.
5. The capacitance of pins P0.0 and P0.7 for the 87C055 exceeds 10 pF; for P0.0 this is maximum 40 pF, while for P0.7 it is maximum 20 pF.

## 20 AC CHARACTERISTICS

$V_{DD}$ = 5 V ±10%; $T_{amb}$ = 0 to +70 °C; all voltages with respect to $V_{SS}$; unless otherwise specified.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|---|
| $1/t_{CLCL}$ | XTAL frequency | note 1 | 6 | – | 12 | MHz |
| $t_{CHCX}$ | XTAL1 clock HIGH time | note 2 | 20 | – | – | ns |
| $t_{CLCX}$ | XTAL1 clock LOW time | | 20 | – | – | ns |
| $t_{CLCH}$ | XTAL1 clock rise time | | – | – | 20 | ns |
| $t_{CHCL}$ | XTAL1 clock fall time | | 5 | – | 20 | ns |
| $1/t_{VCLCL}$ | VCLK frequency | | 5 | – | 8 | MHz |
| $\lvert t_{VCOH} - t_{VCOL} \rvert$ | Rise versus fall time skew on any one of VID2 to VID0, VCTRL and BF | note 3 | – | – | 40 | ns |
| $\lvert t_{VCOH1} - t_{VCOH2} \rvert$ | Rise time skew between any two of VID2 to VID0, VCTRL and BF | | – | – | 30 | ns |
| $\lvert t_{VCOL1} - t_{VCOL2} \rvert$ | Fall time skew between any two of VID2 to VID0, VCTRL and BF | | – | – | 30 | ns |

**Notes**

1. The 83C055 is tested at its maximum XTAL frequency, but not at any other (lower) rate.
2. These parameters apply only when an external clock signal is used.
3. These parameters assume equal loading at $C_L$ = 100 pF, for all the referenced outputs. These parameters are specified but not tested.

# Low voltage/low power single-chip 8-bit microcontroller with I²C

# 80CL410/83CL410

## DESCRIPTION

The 80CL410/83CL410 (hereafter generically referred to as 8XCL410) is manufactured in an advanced CMOS process that allows the part to operate at supply voltages down to 1.8V and oscillator frequencies down to DC. The 8XCL410 has the same instruction set as the 80C51.

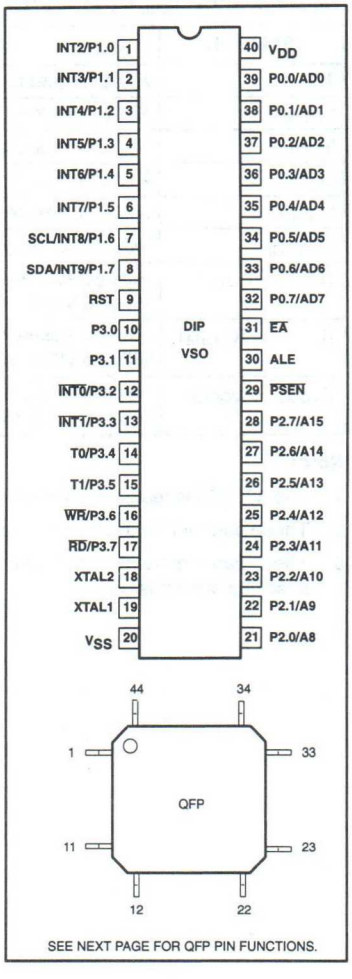
The 8XCL410 features a 4k byte ROM (83CL410), 128 bytes RAM (both ROM and RAM are externally expandable to 64k bytes), four 8-bit ports, two 16-bit timer/counters, an I²C serial interface, a thirteen source, two priority level nested interrupt structure, and on-chip oscillator circuitry suitable for quartz crystal, ceramic resonator, RC, or LC.

The 8XCL410 has two reduced power modes that are the same as those on the standard 80C51. The special reduced power feature of this part is that it can be stopped and then restarted. Running from an external clock source, the clock can be stopped and after a period of time restarted. The 8XCL410 will resume operation from where it was when the code stopped with no loss of internal state, RAM contents, or Special Function Register contents. If the internal oscillator is used the part cannot be stopped and started, but the power-down mode, which can be terminated via an interrupt, can be used to achieve similar power savings and then restart without loss of on-chip RAM and Special Function Register values.

## FEATURES

- Single supply voltage 1.8V to 6.0V
- Frequency from DC to 12MHz
- 80C51 based architecture
  - 4k × 8 ROM (64k external)
  - 128 × 8 RAM (64k external)
  - Four 8-bit I/O ports
  - Two 16-bit timer/counters
  - A thirteen-source, two-level, nested priority interrupt structure
  - 10 external interrupts
- Fully static 80C51 CPU
- I²C Serial Interface
- Two power control modes
  - Idle mode
  - Power-down mode – can be terminated by reset or external interrupt
- Wake-up via external interrupts at port 1
- Single supply voltage 1.8V to 6.0V
- Frequency range of DC to 12MHz
- On-chip oscillator (quartz crystal, ceramic resonator, RC, LC)
- Very low power consumption
- Operating temperature range: –40 to +85°C

## PIN CONFIGURATION




SEE NEXT PAGE FOR QFP PIN FUNCTIONS.

## ORDERING CODE

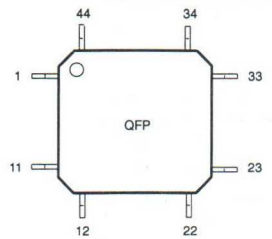
| PHILIPS PART ORDER NUMBER PART MARKING | | PHILIPS NORTH AMERICA PART ORDER NUMBER[1] | | TEMPERATURE °C AND PACKAGE | FREQUENCY | Drawing Number |
|---|---|---|---|---|---|---|
| ROMless | ROM | ROMless | ROM | | | |
| P80CL410HFP | P83CL410HFP | P80CL410HF N | P83CL410HF N | –40 to +85, 40-Pin Plastic Dual In-line Package | 32kHZ to 12MHz | SOT129-1 |
| P80CL410HFT | P83CL410HFT | P80CL410HF D | P83CL410HF D | –40 to +85, 40-Pin Plastic Very Small Outline Package | 32kHZ to 12MHz | SOT158-1 |
| | P83CL410HFH | | | –40 to +85, 44-Pin Plastic Quad Flat Pack | 32kHZ to 12MHz | SOT307-2 |

**NOTE:**

1. Parts ordered by the Philips North America part number will be marked with the Philips part marking.

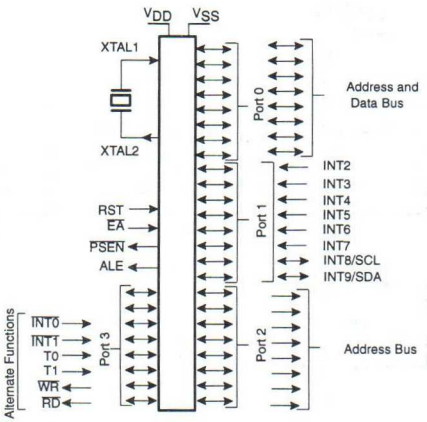
For emulation purposes, the P85CL000 (Piggyback version) with 256 bytes of RAM is recommended.

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS

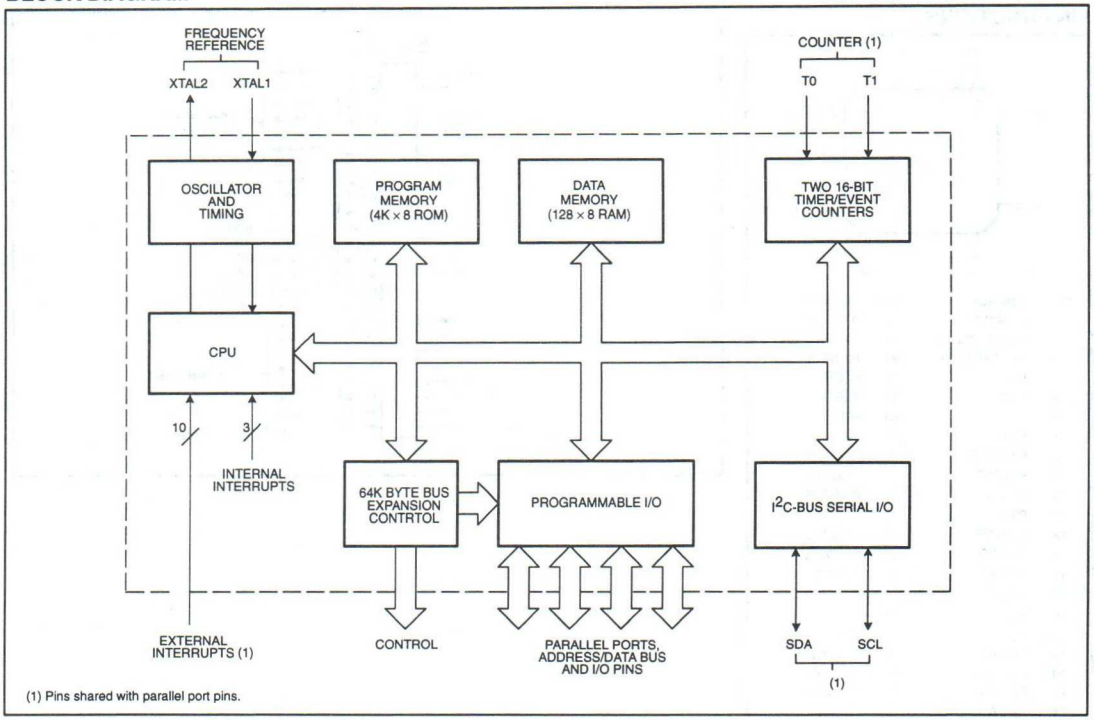



| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5/INT7 | 23 | P2.5/A13 |
| 2 | P1.6/INT8/SCL | 24 | P2.6/A14 |
| 3 | P1.7/INT9/SDA | 25 | P2.7/A15 |
| 4 | RST | 26 | $\overline{PSEN}$ |
| 5 | P3.0 | 27 | ALE |
| 6 | NC | 28 | NC |
| 7 | P3.1 | 29 | $\overline{EA}$ |
| 8 | P3.2/$\overline{INT0}$ | 30 | P0.7/AD7 |
| 9 | P3.3/$\overline{INT1}$ | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/$\overline{WR}$ | 34 | P0.3/AD3 |
| 13 | P3.7/$\overline{RD}$ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | $V_{SS}$ | 38 | $V_{DD}$ |
| 17 | NC | 39 | NC |
| 18 | P2.0/A8 | 40 | P1.0/INT2 |
| 19 | P2.1/A9 | 41 | P1.1/INT3 |
| 20 | P2.2/A10 | 42 | P1.2/INT4 |
| 21 | P2.3/A11 | 43 | P1.3/INT5 |
| 22 | P2.4/A12 | 44 | P1.4/INT6 |

## LOGIC SYMBOL



## BLOCK DIAGRAM




(1) Pins shared with parallel port pins.

## PIN DESCRIPTION

| MNEMONIC | PIN NO. | | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|
| | QFP | DIL40/ VSO40 | | |
| $V_{SS}$ | 16 | 20 | I | **Ground:** 0V reference. |
| $V_{DD}$ | 38 | 40 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 30–37 | 39–32 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. |
| P1.0–P1.7 | 40–44<br>1–3 | 1–8 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Additional functions include: |
| | | 7 | I/O | **SCL (P1.6):** I²C serial bus clock. |
| | | 8 | I/O | **SDA (P1.7):** I²C serial bus data. |
| | | 1–8 | I | **INT2–INT9 (P1.0–P1.7):** Additional external interrupts. |
| P2.0–P2.7 | 18–25 | 21–28 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |
| P3.0–P3.7 | 5, 7–13 | 10–17 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 8 | 12 | I | **$\overline{INT0}$ (P3.2):** External interrupt 0 |
| | 9 | 13 | I | **$\overline{INT1}$ (P3.3):** External interrupt 1 |
| | 10 | 14 | I | **T0 (P3.4):** Timer 0 external input |
| | 11 | 15 | I | **T1 (P3.5):** Timer 1 external input |
| | 12 | 16 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 13 | 17 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 4 | 9 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{DD}$. |
| ALE | 27 | 30 | O | **Address Latch Enable:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. |
| $\overline{PSEN}$ | 26 | 29 | O | **Program Store Enable:** The read strobe to external program memory. When the device is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}$ | 29 | 31 | I | **External Access Enable:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H to 0FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 0FFFH. |
| XTAL1 | 15 | 19 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input for an external clock source. |
| XTAL2 | 14 | 18 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**Table 1. 8XCL410 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes): | | | | | | | | | | |
| DPH | High byte | 83H | | | | | | | | | 00H |
| DPL | Low byte | 82H | | | | | | | | | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP0*# | Interrupt priority 0 | B8H | – | – | PS1 | – | PT1 | PX1 | PT0 | PX0 | xx000000B |
| | | | FF | FE | FD | FC | FB | FA | F9 | F8 | |
| IP1*# | Interrupt priority 1 | F8H | PX9 | PX8 | PX7 | PX6 | PX5 | PX4 | PX3 | PX2 | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IEN0*# | Interrupt enable 0 | A8H | EA | – | ES1 | – | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | EF | EE | ED | EC | EB | EA | E9 | E8 | |
| IEN1*# | Interrupt enable 1 | E8H | EX9 | EX8 | EX7 | EX6 | EX5 | EX4 | EX3 | EX2 | 00H |
| | | | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 | |
| IRQ1*# | Interrupt request flag | C0H | IQ9 | IQ8 | IQ7 | IQ6 | IQ5 | IQ4 | IQ3 | IQ2 | 00H |
| IX1# | Interrupt polarity | E9H | | | | | | | | | 00H |
| P0* | Port 0 | 80H | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | FFH |
| P1* | Port 1 | 90H | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | FFH |
| P2* | Port 2 | A0H | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | FFH |
| P3* | Port 3 | B0H | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | FFH |
| PCON | Power control | 87H | SMOD | – | – | – | GF1 | GF0 | PD | IDL | 0xxx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| S1ADR# | Slave address | DBH | | | | | | | | | 00H |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| S1CON*# | Serial control | D8H | – | ENS1 | STA | STO | SI | AA | CR1 | CR0 | x0000000B |
| S1DAT# | Serial data | DAH | | | | | | | | | 00H |
| S1STA# | Serial status | D9H | | | | | | | | | 11111000B |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer/counter control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| TMOD | Timer/counter mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |
| TH0 | Timer 0 high byte | 8CH | | | | | | | | | 00H |
| TH1 | Timer 1 high byte | 8DH | | | | | | | | | 00H |
| TL0 | Timer 0 low byte | 8AH | | | | | | | | | 00H |
| TL1 | Timer 1 low byte | 8BH | | | | | | | | | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

## PORT OPTIONS

The pins of port 1 (not P1.6/SCL or P1.7/SDA), port 2, and port 3 may be individually configured with one of the following port options (see Figure 1):

Option 1: **Standard Port**—quasi-bidirectional I/O with pull-up. The strong booster pull-up p1 is turned on for two oscillator periods after a 0-to-1 transition in the port latch. See Figure 1(a).

Option 2: **Open Drain**—quasi-bidirectional I/O with n-channel open drain output. Use as an output requires the connection of an external pull-up resistor. See Figure 1(b).

Option 3: **Push-Pull**—output with drive capability in both polarities. Under this option, pins can only be used as outputs. See Figure 1(c).

The definition of port options for port 0 is slightly different.

Two cases have to be examined. First, accesses to external memory ($\overline{EA}$ = 0 or access above the built-in memory boundary), and second, I/O accesses.

### External Memory Accesses

Option 1: True 0 and 1 are written as address to the external memory (strong pull-up is used).

Option 2: An external pull-up resistor is needed for external accesses.

Option 3: Not allowed for external memory accesses as the port can only be used as output.

### I/O Accesses

Option 1: When writing a 1 to the port latch, the strong pull-up p1 will be on for two oscillator periods. No weak pull-up exists. Without an external pull-up, this option can be used as a high-impedance input.

Option 2: Open drain—quasi-bidirectional I/O with n-channel open drain output. Use as an output requires the connection of an external pull-up resistor. See Figure 1(c).

Option 3: Push-Pull—output with drive capability in both polarities. Under this option, pins can only be used as outputs.

Individual mask selection of the post-reset state is available on any of the above pins. Make your selection by appending "S" or "R" to option 1, 2, or 3 above (e.g., 1S for a standard I/O to be set after RESET or 2R for an open-drain I/O to be reset after RESET.

Option S: **Set**—after reset, this pin will be initialized High.

Option R: **Reset**—after reset, this pin will be initialized Low.

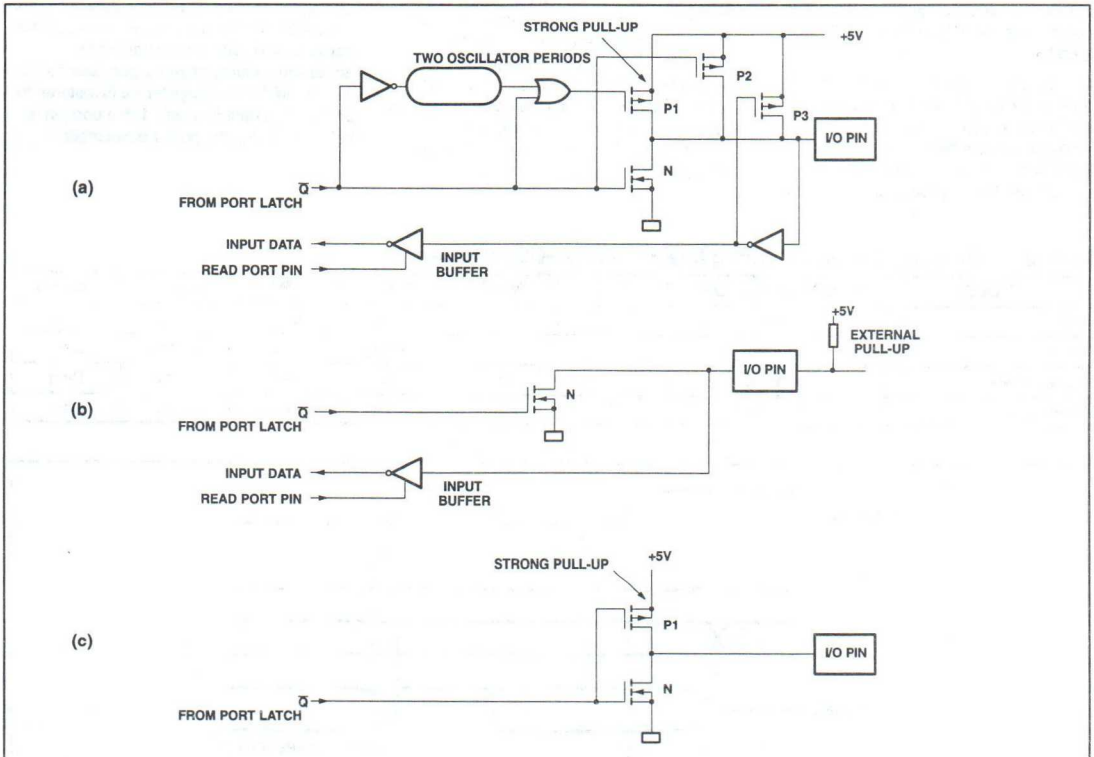



**Figure 1. Ports**

## POWER-DOWN MODE

The instruction setting PCON.1 is the last executed prior to going into the power-down mode. In power-down mode, the oscillator is stopped. The contents of the the on-chip RAM and SFRs are preserved. The port pins output the values held by their respective SFRs. ALE and $\overline{PSEN}$ are held low.

In the power-down mode, $V_{DD}$ may be reduced to minimize power consumption. However, the supply voltage must not be reduced until the power-down mode is active, and must be restored before the hardware reset is applied and frees the oscillator. Reset must be held active until the oscillator has restarted and stabilized.

From the power-down mode the part can be restarted by using either the wake-up mode or the Reset Mode.

### Wake-Up Mode

Setting both PD and IDL bits in the PCON register forces the controller into the power-down mode. Setting both bits enable the controller to be woken-up from the power-down mode via either an enabled external interrupt INT2–INT9, or a reset operation.

An external interrupt for an enabled interrupt INT2–INT9 at port 1 starts both the oscillator and the delay counter. To ensure that the oscillator is stable before the controller restarts, the internal clock will remain inactive for 1536 oscillator periods after the interrupt is detected. This is controlled by the on-chip delay counter. After this, the PD flag will be reset, the controller is now in the Idle mode and the interrupt will be handled in the normal way.

### Reset Mode

Setting only the PD bit in the PCON register again forces the controller into the power-down mode, but in this case it can only be restored to normal operation with a direct reset operation.

To restore normal operation, the RESET pin has to be kept High for a minimum of 24 oscillator periods. The on-chip delay counter is inactive. The user has to insure that the oscillator is stable before any operation is attempted. Figure 2 illustrates the two possibilities for wake-up.

## IDLE MODE

The instruction that sets PCON.0 is the last instruction executed before going into idle mode. In idle mode, the internal clock is stopped for the CPU, but not for the interrupt, timer, and serial port functions. The CPU status is preserved along with the stack pointer, program counter, program status word and accumulator. The RAM and all other registers maintain their data during idle mode. The port pins retain the logical states they held at idle mode activation. ALE and $\overline{PSEN}$ hold at the logic high level.

There are two methods used to terminate the idle mode. Activation of any interrupt will cause PCON to be cleared by hardware; terminating idle mode. The interrupt is serviced, and following the instruction RETI, the next instruction to be executed will be the one following the instruction that put the device in the the idle mode.

Flag bits GF0 and GF1 can be used to determine whether the interrupt was received during normal execution or idle mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When idle mode is terminated by an interrupt, the service routine can examine the status of the flag bits.

The second method of terminating the idle mode is with an external hardware reset. Since the oscillator is still running, the hardware reset is required to be active for only two machine cycles to complete the reset operation. Reset redefines all SFRs, but does not affect the state of the on-chip RAM.

The status of the external pins during idle and power-down mode is shown in Table 2. If the power-down mode is activated while accessing external memory, port data held in the special function register P2 is restored to port 2. If the data is a logic 1, the port pin is held high during the power-down mode.

**Table 2. External Pin Status During Idle and Power-Down Modes**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Floating | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Floating | Data | Data | Data |

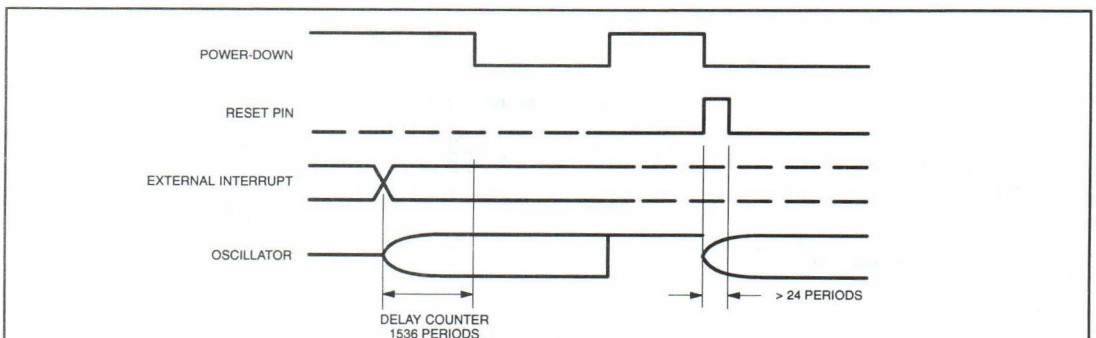



**Figure 2. Wake-Up Operation**

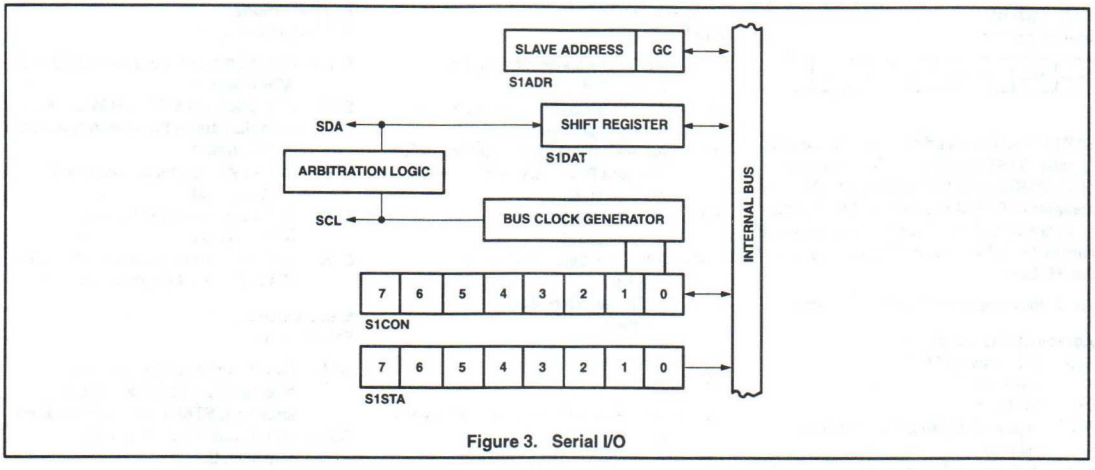



**Figure 3. Serial I/O**

## I²C-BUS SERIAL I/O

The serial port supports the twin line I²C-bus. The I²C-bus consists of a data line (SDA) and a clock line (SCL). These lines also function as I/O port lines P1.7 and P1.6 respectively. The system is unique because data transport, clock generation, address recognition and bus control arbitration are all controlled by hardware. The I²C-bus serial I/O has complete autonomy in byte handling and operates in four modes:

- Master transmitter
- Master receiver
- Slave transmitter
- Slave receiver

These functions are controlled by the S1CON register. S1STA is the status register whose contents may also be used as a vector to various service routines. S1DAT is the data shift register and S1ADR the slave address register. Slave address recognition is performed by hardware.

**S1CON (D8H)**
**Serial control register**

| CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |
|---|---|---|---|---|---|---|---|

**CR0, CR1, CR2**

These three bits determine the serial clock frequency when SIO is in a master mode.

**AA**

Assert acknowledge bit. When the AA flag is set, an acknowledge (low level to SDA) will be returned during the acknowledge clock pulse on the SCL line when:

- own slave address is received
- general call address is received (S1ADR.0 = 1)
- data byte received while device is programmed as master
- data byte received while device is selected slave

With AA = 0, no acknowledge will be returned. Consequently, no interrupt is requested when the "own slave address" or general call address is received.

**SI**

SIO interrupt flag. When the SI flag is set, an acknowledge is returned after any one of the following conditions:

- a start condition is generated in master mode
- own slave address received during AA = 1
- general call address received while S1ADR.0 and AA = 1
- data byte received or transmitted in master mode (even if arbitration is lost)
- data byte received or transmitted as selected slave
- stop or start condition received as selected slave receiver or transmitter

**STO**

STOP flag. With this bit set while in master mode, a STOP condition is generated. When a STOP condition is detected on the bus, the SIO hardware clears the STO flag. In the slave mode, the STO flag may also be set to recover from an error condition. In this case, no STOP condition is transmitted to the I²C-bus. However, the SIO hardware behaves as if a STOP condition has been received and releases SDA and SCL. The SIO then switches to the "not addressed" slave receiver mode. The STO flag is automatically cleared by hardware.

**STA**

START flag. When the STA bit is set in slave mode, the SIO hardware checks the status of the I²C-bus and generates a START condition if the bus is free. If STA is set while the SIO is in master mode, SIO transmits a repeated START condition.

**ENS1**

When ENS1 = 0, the SIO is disabled. The SDA and SCL outputs are in a high-impedance state; P1.6 and P1.7 function as open drain ports.

When ENS1 = 1, the SIO is enabled. The P1.6 and P1.7 port latches must be set to logic 1.

**S1STA (D9H)**
**Status register**

| SC4 | SC3 | SC2 | SC1 | SC0 | 0 | 0 | 0 |
|---|---|---|---|---|---|---|---|

S1STA is an 8-bit read-only special function register. S1STA.3–S1STA.7 hold a status code. S1STA.0–S1STA.2 are held LOW. The contents of S1STA may be used as a vector to a service routine. This optimizes response time of the software and consequently that of the I²C-bus.

The following is a list of the status codes:

**Abbreviations used:**

SLA: 7-bit slave address
R: Read bit
W: Write bit
$\overline{ACK}$: Acknowledgement (acknowledge bit = 0)
ACK: Not Acknowledge (acknowledge bit = 1)
DATA: 8-bit byte to or from the I²C-bus
MST: Master
SLV: Slave
TRX: Transmitter
REC: Receiver

**MST/TRX mode**
S1STA value

08H – a START condition has been transmitted
10H – a repeated START condition has been transmitted
18H – SLA and W have been transmitted, ACK received
20H – SLA and W have been transmitted, $\overline{ACK}$ received
28H – DATA of S1DAT has been transmitted, ACK received
30H – DATA of S1DAT has been transmitted, $\overline{ACK}$ received
38H – Arbitration lost in SLA, R/W or DATA

**MST/REC mode**
S1STA value

08H – a START condition has been transmitted
10H – a repeated START condition has been transmitted
38H – Arbitration lost while returning $\overline{ACK}$
40H – SLA and R have been transmitted, ACK received
48H – SLA and R have been transmitted, $\overline{ACK}$ received
50H – DATA has been received, ACK returned
58H – DATA has been received, $\overline{ACK}$ returned

**SLV/REC mode**
S1STA value

60H – Own SLA and W have been received, ACK returned
68H – Arbitration lost in SLA, R/W as MST. Own SLA and W have been received, ACK returned
70H – General CALL has been received, ACK returned
78H – Arbitration lost in SLA, R/W as MST. General CALL has been received
80H – Previously addressed with own SLA. DATA byte received, ACK returned
88H – Previously addressed with own SLA. DATA byte received, $\overline{ACK}$ returned
90H – Previously addressed with general CALL. DATA byte has been received, ACK has been returned
98H – Previously addressed with general CALL. DATA byte has been received, $\overline{ACK}$ has been returned
A0H – A STOP condition or repeated START condition has been received while still addressed as SLV/REC or SLV/TRX

**SLV/TRX mode**
S1STA value

A8H – Own SLA and R have been received, ACK returned
B0H – Arbitration lost in SLA, R/W as MST. Own SLA and R have been received, ACK returned
B8H – DATA byte has been transmitted, ACK received
C0H – DATA byte has been transmitted, $\overline{ACK}$ received
C8H – Last DATA byte has been transmitted (AA = logic 0), ACK received

**Miscellaneous**
S1STA value

00H – Bus error during MST mode or selected SLV mode, due to an erroneous START or STOP condition
F8H – No relevant state interruption available, SI = 0.

**S1DAT (DAH)**
**Data Shift Register**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|

**Data shift register S1DAT**
This register contains the serial data to be transmitted or data that has just been received. Bit 7 is transmitted or received first, i.e., data is shifted from left to right.

**S1ADR (DBH)**
**Slave Address Register**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|

S1ADR.0, GC: 0 = general CALL address is not recognized
1 = general CALL address is recognized

S1ADR.7-1: own slave address

This 8-bit register may be loaded with the 7-bit slave address, to which the controller will respond when programmed as a slave receiver/transmitter. The LSB bit (GC) is used to determine whether the general CALL address is recognized.

**Table 3. SCL Frequency**

| CR2 | CR1 | CR0 | $f_{OSC}$ DIVIDED BY | BIT RATE (kHz) at $f_{OSC}$ | | |
|---|---|---|---|---|---|---|
| | | | | 3.58MHz | 6MHz | 12MHz |
| 0 | 0 | 0 | 256 | 14.0 | 23.4 | 46.9 |
| 0 | 0 | 1 | 224 | 16.0 | 26.8 | 53.6 |
| 0 | 1 | 0 | 192 | 18.6 | 31.3 | 62.5 |
| 0 | 1 | 1 | 160 | 22.4 | 37.5 | 75.0 |
| 1 | 0 | 0 | 960 | 3.73 | 6.25 | 12.5 |
| 1 | 0 | 1 | 120 | 29.8 | 50 | 100 |
| 1 | 1 | 0 | 60 | 59.7 | 100 | – |
| 1 | 1 | 1 | not allowed | – | – | – |

## INTERRUPT SYSTEM

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution, a multiple-source, two-priority level, nested interrupt system is provided. The 8XCL410 acknowledges interrupt requests from thirteen sources, as follows:

- INT0 and INT1
- Timer 0 and timer 1
- I²C-bus serial I/O interrupt
- INT2 to INT9 (port 1)

Each interrupt vectors to a separate location in program memory for its service routine. Each source can be individually enabled or disabled by corresponding bits in the internal enable registers (IEN0, IEN1) The priority level is selected via the interrupt priority register (IP0, IP1). All enabled sources can be globally disabled or enabled.

### External Interrupts INT2–INT9

Port 1 lines serve an alternative purpose as eight additional interrupts INT2–INT9. When enabled, each of these lines can "wake-up" the device from power-down mode. Using the IX1 register, each pin may be initialized to either active high or low. IRQ1 is the interrupt request flag register. Each flag, if the interrupt is enabled, will be set on an interrupt request but it must be cleared by software.

**IEN0 (A8H)**
**Interrupt enable register**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| EA | — | ES1 | — | ET1 | EX1 | ET0 | EX0 |

| Bit | Symbol | Function |
|---|---|---|
| IEN0.7 | EEA | General enable/disable control<br>0 = no interrupt is enabled<br>1 = any individually enabled interrupt will be accepted |
| IEN0.6 | — | (unused) |
| IEN0.5 | ES1 | Enable $I_2C$ SIO interrupt |
| IEN0.4 | — | (unused) |
| IEN0.3 | ET1 | Enable Timer T1 interrupt |
| IEN0.2 | EX1 | Enable external interrupt 1 |
| IEN0.1 | ET0 | Enable Timer T0 interrupt |
| IEN0.0 | EX0 | Enable external interrupt 0 |

**IEN1 (E8H)**
**Interrupt enable register**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| EX9 | EX8 | EX7 | EX6 | EX5 | EX4 | EX3 | EX2 |

| Bit | Symbol | Function |
|---|---|---|
| IEN1.7 | EX9 | Enable external interrupt 9 |
| IEN1.6 | EX8 | Enable external interrupt 8 |
| IEN1.5 | EX7 | Enable external interrupt 7 |
| IEN1.4 | EX6 | Enable external interrupt 6 |
| IEN1.3 | EX5 | Enable external interrupt 5 |
| IEN1.2 | EX4 | Enable external interrupt 4 |
| IEN1.1 | EX3 | Enable external interrupt 3 |
| IEN1.0 | EX2 | Enable external interrupt 2 |

where 0 = interrupt disabled
1 = interrupt enabled

**IP0 (B8H)**
**Interrupt priority register**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| — | — | PS1 | — | PT1 | PX1 | PT0 | PX0 |

| Bit | Symbol | Function |
|---|---|---|
| IP0.7 | — | (unused) |
| IP0.6 | — | (unused) |
| IP0.5 | PS1 | $I_2C$ SIO interrupt priority level |
| IP0.4 | — | (unused) |
| IP0.3 | PT1 | Timer 1 interrupt prioity level |
| IP0.2 | PX1 | External interrupt 1 priority level |
| IP0.1 | PT0 | Timer 0 interrupt prioity level |
| IP0.0 | PX0 | External interrupt 0 priority level |

**IP1 (F8H)**
**Interrupt priority register**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| PX9 | PX8 | PX7 | PX6 | PX5 | PX4 | PX3 | PX2 |

| Bit | Symbol | Function |
|---|---|---|
| IP1.7 | PX9 | External interrupt 9 priority level |
| IP1.6 | PX8 | External interrupt 8 priority level |
| IP1.5 | PX7 | External interrupt 7 priority level |
| IP1.4 | PX6 | External interrupt 6 priority level |
| IP1.3 | PX5 | External interrupt 5 priority level |
| IP1.2 | PX4 | External interrupt 4 priority level |
| IP1.1 | PX3 | External interrupt 3 priority level |
| IP1.0 | PX2 | External interrupt 2 priority level |

Interrupt priority is as follows:
0 – low priority
1 – high priority

**IX1 (E9H)**
**Interrupt polarity register**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| IL9 | IL8 | IL7 | IL6 | IL5 | IL4 | IL3 | IL2 |

| Bit | Symbol | Function |
|---|---|---|
| IX1.7 | IL9 | External interrupt 9 polarity level |
| IX1.6 | IL8 | External interrupt 8 polarity level |
| IX1.5 | IL7 | External interrupt 7 polarity level |
| IX1.4 | IL6 | External interrupt 6 polarity level |
| IX1.3 | IL5 | External interrupt 5 polarity level |
| IX1.2 | IL4 | External interrupt 4 polarity level |
| IX1.1 | IL3 | External interrupt 3 polarity level |
| IX1.0 | IL2 | External interrupt 2 polarity level |

Writing either a "1" or "0" to an IX1 register bit sets the priority level of the corresponding external interrupt to active High or Low, respectively.

**IRQ1 (C0H)**
**Interrupt request flag register**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| IQ9 | IQ8 | IQ7 | IQ6 | IQ5 | IQ4 | IQ3 | IQ2 |

| Bit | Symbol | Function |
|---|---|---|
| IRQ1.7 | IQ9 | External interrupt 9 request flag |
| IRQ1.6 | IQ8 | External interrupt 8 request flag |
| IRQ1.5 | IQ7 | External interrupt 7 request flag |
| IRQ1.4 | IQ6 | External interrupt 6 request flag |
| IRQ1.3 | IQ5 | External interrupt 5 request flag |
| IRQ1.2 | IQ4 | External interrupt 4 request flag |
| IRQ1.1 | IQ3 | External interrupt 3 request flag |
| IRQ1.0 | IQ2 | External interrupt 2 request flag |

| Priority | Vector | Source |
|---|---|---|
| X0 (highest) | 0003H | External 0 |
| S1 | 002BH | I²C port |
| X5 | 0053H | External 5 |
| T0 | 000BH | Timer 0 |
| X6 | 005BH | External 6 |
| X1 | 0013H | External 1 |
| X2 | 003BH | External 2 |
| X7 | 0063H | External 7 |
| T1 | 001BH | Timer 1 |
| X3 | 0043H | External 3 |
| X8 | 006BH | External 8 |
| X4 | 004BH | External 4 |
| X9 (lowest) | 0073H | External 9 |

| Register | Function | SFR Address |
|---|---|---|
| IX1 | Interrupt polarity register | E9H |
| IRQ1 | Interrupt request flag register | C0H |
| IEN0 | Interrupt enable register | A8H |
| IEN1 | Interrupt enable register (INT2–INT9) | E8H |
| IP0 | Interrupt priority register | B8H |
| IP1 | Interrupt priority register (INT2–INT9) | F8H |

## OSCILLATOR CIRCUITRY

The on-chip oscillator circuitry of the 8XCL410 is a single stage inverting amplifier biased by an internal feedback resistor. (See Figure 4.) The oscillator can be operated with a quartz crystal, ceramic resonator, LC network or RC network. See Figure 5 for different configurations. When ordering parts, it is necessary to specify an oscillator option. The options are: RC when an RC network will be used, OSC 2 for oscillator operation below 4MHz, OSC 3 for oscillator operation from 4MHz to 10MHz, OSC 4 for oscillator operation above 10MHz, and 32kHz if 32kHz to 400kHz operation is desired.

For operation as a standard quartz oscillator, no external components are needed (except at 32KHz). When using external capacitors, ceramic resonators, coils, and RC networks to drive the oscillator, five different configurations are supported (see Figure 5 and Table 4).

In the power-down mode the oscillator is stopped and XTAL1 is pulled high. The oscillator inverter is switched off to ensure no current will flow. To drive the device with an external clock source, apply the external clock signal to XTAL1, and leave XTAL2 to float, as shown in Figure 5(f). There are no requirements on the duty cycle of the external clock, since the input to the internal clocking circuitry is split using a flip-flop.

The following options are provided for optimum on-chip oscillator performance. Please state option when ordering:

Osc.1: Figure 5(c). An option for 32kHz clock applications with external trimmer for frequency adjustment.

A 4.7MΩ bias resistor must be connected in parallel with the crystal.

Osc.2: Figure 5(e). An option for low-power, low-frequency operations using LC components or quartz.

Osc.3: An option for medium frequency range applications.

Osc.4: An option for high frequency range applications.

RC: Figure 5(g). An option for an RC oscillator.

The equivalent circuit data of the internal oscillator compares with that of matched crystals.

The externally adjustable RC oscillator has a frequency range from 100kHz to 500kHz. (See Figure 7.)

## Power-on Reset

The 8XCL410 contains on-chip circuitry which switch the port pins to the customer-defined logic level as soon as $V_{DD}$ exceeds 1.3V if the mask option "ON" has been chosen (see Figures 8 and 9). As soon as the minimum supply voltage is reached, the oscillator will start up. However, to ensure that the oscillator is stable before the controller starts, the clock signals are gated away from the CPU for a further 1536 oscillator periods.

An hysteresis of approximately 50mV at a typical power-on switching level of 1.3V will ensure correct operation.

The on-chip power–on reset circuitry can also be switched off via the mask option "OFF". This option reduces the power-down current to typically 800µA and can be chosen if external reset circuitry is used. For applications not requiring the internal reset, option "OFF" should be chosen.

An automatic reset can be obtained at power-on by connecting the RST pin to $V_{DD}$ via a 10µF capacitor. At power-on, the voltage on the RST pin is equal to $V_{DD}$ minus the capacitor voltage, and decreases from $V_{DD}$ as the capacitor discharges through the internal resistor $R_{RST}$ to ground. The larger the capacitor, the more slowly $V_{RST}$ decreases. $V_{RST}$ must remain above the lower threshold of the Schmitt trigger long enough to effect a complete reset. The time required is the oscillator start-up time, plus 2 machine cycles.

## P80CL410: ROM-less VERSION OF P83CL410

The P80CL410 is a low voltage ROMless version of the P83CL410. The mask options on the P80CL410 are fixed as follows:

- Port Options:
  All ports except P16/P17 have option "1S", i.e., standard port, High after reset. The ports P16/P17 have option "2S", i.e., open drain, High after reset.
- Oscillator option: OSC3
- Power-on Reset option: OFF

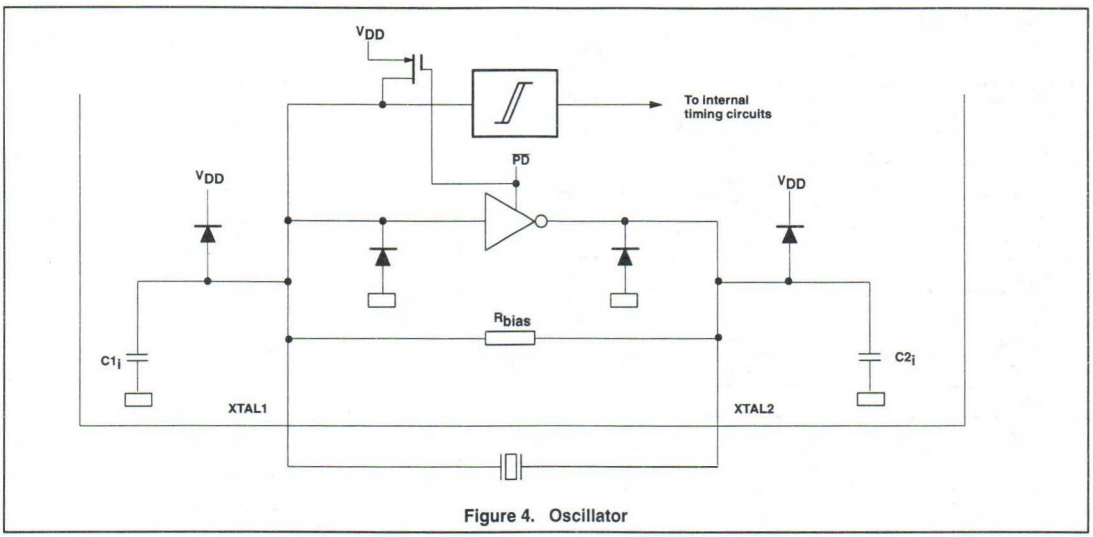



Figure 4. Oscillator

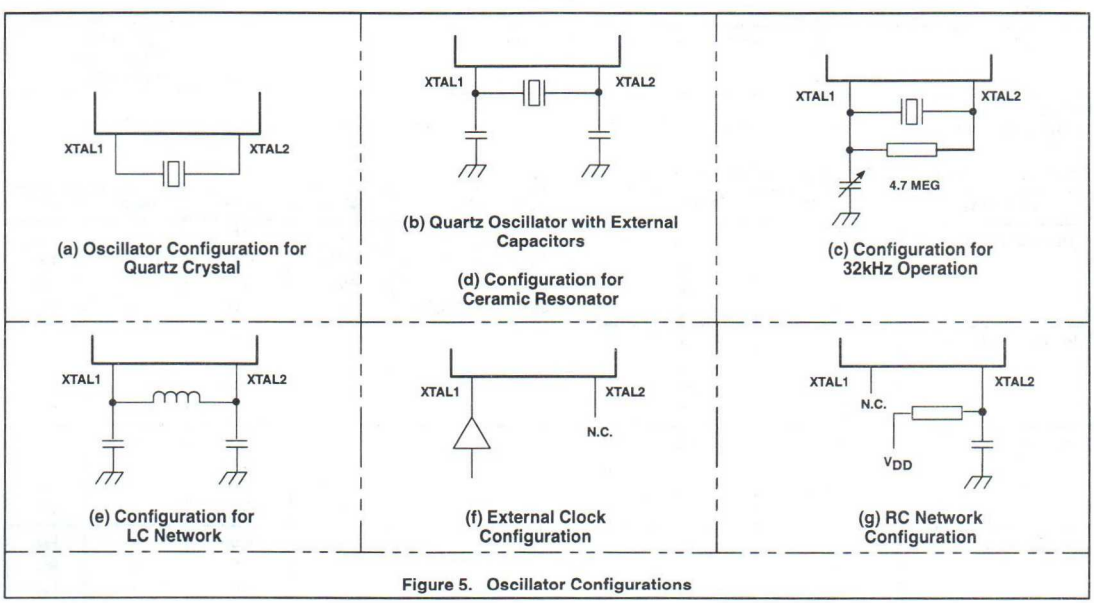



Figure 5. Oscillator Configurations

**Table 4. Oscillator Type Selection Guide**

| RESONATOR | f (MHz) | OPTION | C1 EXT. MIN | C1 EXT. MAX | C2 EXT. MIN | C2 EXT. MAX | MAXIMUM RESONATOR SERIES RESISTANCE |
|---|---|---|---|---|---|---|---|
| Quartz | 0.032 | Osc.1 | 5 | 15 | 0 | 0 | 15kΩ[1] |
| Quartz | 1.0 | Osc.2 | 0 | 30 | 0 | 30 | 600Ω |
| Quartz | 3.58 | Osc.2 | 0 | 15 | 0 | 15 | 100Ω |
| Quartz | 4.0 | Osc.2 | 0 | 20 | 0 | 20 | 75Ω |
| Quartz | 6.0 | Osc.3 | 0 | 10 | 0 | 10 | 60Ω |
| Quartz | 10.0 | Osc.4 | 0 | 15 | 0 | 15 | 60Ω |
| Quartz | 12.0 | Osc.4 | 0 | 10 | 0 | 10 | 40Ω |
| Quartz | 16.0 | Osc.4 | 0 | 15 | 0 | 15 | 20Ω |
| PXE | 0.455 | Osc.2 | 40 | 50 | 40 | 50 | 10Ω |
| PXE | 1.0 | Osc.2 | 15 | 50 | 15 | 50 | 100Ω |
| PXE | 3.58 | Osc.2 | 0 | 40 | 0 | 40 | 10Ω |
| PXE | 4.0 | Osc.2 | 0 | 40 | 0 | 40 | 10Ω |
| PXE | 6.0 | Osc.2 | 0 | 20 | 0 | 20 | 5Ω |
| PXE | 10.0 | Osc.3 | 0 | 15 | 0 | 15 | 6Ω |
| PXE | 12.0 | Osc.4 | 10 | 40 | 10 | 40 | 6Ω |
| LC | | Osc.2 | 20 | 90 | 20 | 90 | 10µH = 1Ω<br>100µH = 5Ω<br>1mH = 75Ω |

**NOTE:**
1. 32kHz quartz crystals with a series resistance higher than 15kΩ will reduce the guaranteed supply voltage range to 2.5 to 3.5V.

**Table 5. Oscillator Equivalent Circuit Parameters (see Figure 6)**

| PARAMETER | OPTION | SYMBOL | CONDITION | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|---|---|
| Transconductance | Osc.1 | $g_m$ | T = +25°C; $V_{DD}$ = 4.5V | – | 15 | – | µs |
| | Osc.2 | $g_m$ | T = +25°C; $V_{DD}$ = 4.5V | 200 | 600 | 1000 | µs |
| | Osc.3 | $g_m$ | T = +25°C; $V_{DD}$ = 4.5V | 400 | 1500 | 4000 | µs |
| | Osc.4 | $g_m$ | T = +25°C; $V_{DD}$ = 4.5V | 1000 | 4000 | 10000 | µs |
| Input capacitance | Osc.1 | $c1_i$ | | – | 3.0 | – | pF |
| | Osc.2 | $c1_i$ | | – | 8.0 | – | pF |
| | Osc.3 | $c1_i$ | | – | 8.0 | – | pF |
| | Osc.4 | $c1_i$ | | – | 8.0 | – | pF |
| Output capacitance | Osc.1 | $c2_i$ | | – | 23.0 | – | pF |
| | Osc.2 | $c2_i$ | | – | 8.0 | – | pF |
| | Osc.3 | $c2_i$ | | – | 8.0 | – | pF |
| | Osc.4 | $c2_i$ | | – | 8.0 | – | pF |
| Output resistance | Osc.1 | R2 | | – | 3800 | – | kΩ |
| | Osc.2 | R2 | | – | 65 | – | kΩ |
| | Osc.3 | R2 | | – | 18 | – | kΩ |
| | Osc.4 | R2 | | – | 5.0 | – | kΩ |

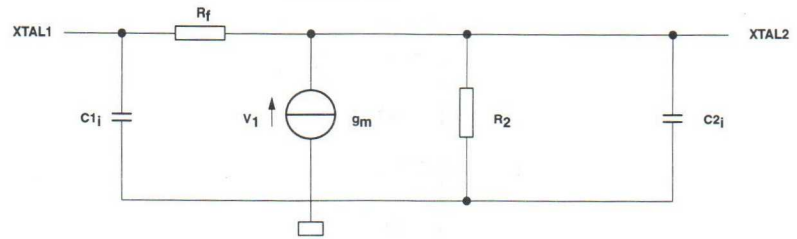



**Figure 6. Equivalent Circuit Diagram**

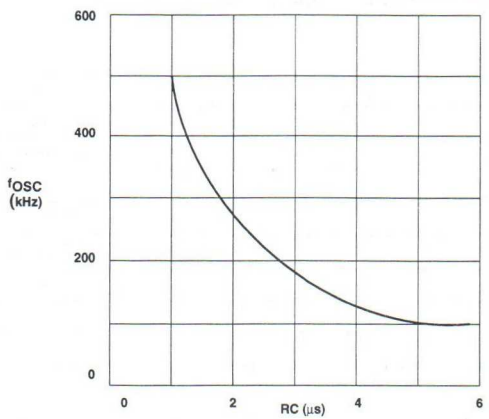



**Figure 7. Frequency as a Function of RC**

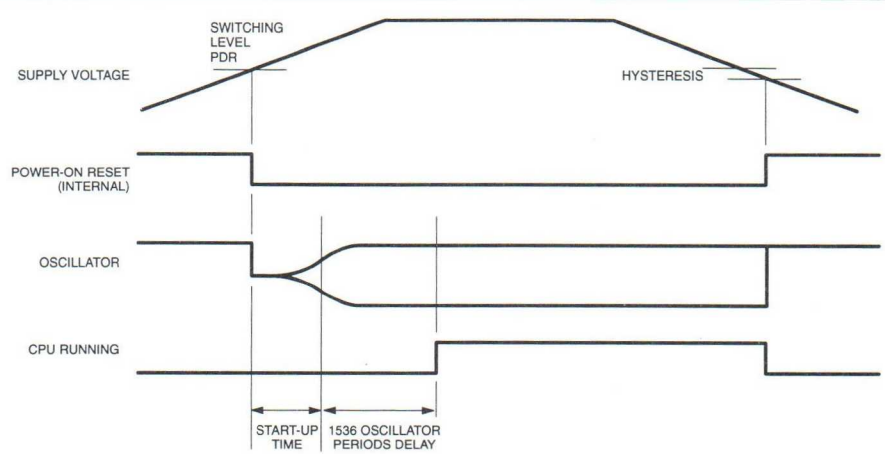



**Figure 8. Power-on Reset Switching Level**

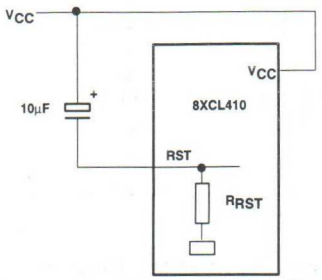



**Figure 9. Recommended Power-on Reset Circuitry**

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Supply voltage | –0.5 to +6.5 | V |
| All input voltages | –0.5 to $V_{DD}$ +0.5 | V |
| DC current into any input or output | 5 | mA |
| Total power dissipation | 300 | mW |
| Storage temperature range | –65 to +150 | °C |
| Operating ambient temperature range | –40 to +85 | °C |
| Operating junction temperature | 125 | °C |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

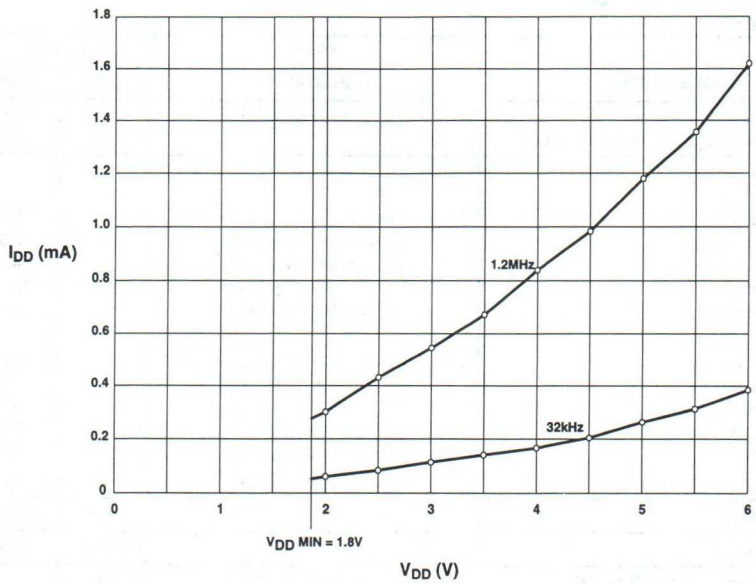
## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = –40°C to +85°C, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|
| $V_{DD}$ | Supply voltage<br>RAM retention voltage in power-down mode | $f_{CLK}$ (see Figure 13) | 1.8<br>1.0 | 6.0<br>— | V<br>V |
| $I_{DD}$ | Power supply current:<br>Operating[1] | | | | |
| | OSC 1 option | $f_{CLK}$ = 32kHz, $V_{DD}$ = 1.8V, $T_{amb}$ = +25°C | — | 50 | µA |
| | OSC 2 option | $f_{CLK}$ = 3.58MHz, $V_{DD}$ = 3V | — | 2.5 | mA |
| | OSC 2 option | $f_{CLK}$ = 10MHz, $V_{DD}$ = 5V | — | 14 | mA |
| | OSC 3 option | $f_{CLK}$ = 12MHz, $V_{DD}$ = 5V | — | 16 | mA |
| | OSC 4 option | $f_{CLK}$ = 12MHz, $V_{DD}$ = 5V | — | 20 | mA |
| | Idle mode[2] | | | | |
| | OSC 1 option | $f_{CLK}$ = 32kHz, $V_{DD}$ = 1.8V, $T_{amb}$ = +25°C | — | 25 | µA |
| | OSC 2 option | $f_{CLK}$ = 3.58MHz, $V_{DD}$ = 3V | — | 1.0 | mA |
| | OSC 2 option | $f_{CLK}$ = 10MHz, $V_{DD}$ = 5V | — | 5.0 | mA |
| | OSC 3 option | $f_{CLK}$ = 12MHz, $V_{DD}$ = 5V | — | 7.0 | mA |
| | OSC 4 option | $f_{CLK}$ = 12MHz, $V_{DD}$ = 5V | — | 8.5 | mA |
| | Power-down mode[3] | $V_{DD}$ = 1.8V, $T_{amb}$ = +25°C | — | 10 | µA |
| $V_{IL}$ | Input low voltage | | $V_{SS}$ | $0.3V_{DD}$ | V |
| $V_{IH}$ | Input high voltage | | $0.7V_{DD}$ | $V_{DD}$ | V |
| $I_{OL}$ | Output sink current, except SDA, SCL | $V_{DD}$ = 5V, $V_{OL}$ = 0.4V<br>$V_{DD}$ = 2.5V, $V_{OL}$ = 0.4V | 1.6<br>0.7 | | mA<br>mA |
| $I_{OL1}$ | Output sink current, SDA, SCL | $V_{DD}$ = 5V, $V_{OL}$ = 0.4V | 3.0 | | mA |
| $I_{OH}$ | Output source current (push-pull options only) | $V_{DD}$ = 5V, $V_{OH}$ = $V_{DD}$ – 0.4V<br>$V_{DD}$ = 2.5V, $V_{OH}$ = $V_{DD}$ – 0.4V | 1.6<br>0.7 | | mA<br>mA |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3 | $V_{DD}$ = 5V, $V_{IN}$ = 0.4V<br>$V_{DD}$ = 2.5V, $V_{IN}$ = 0.4V | | –100<br>–50 | µA<br>µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3 | $V_{DD}$ = 5V, $V_{IN}$ = $V_{DD}$/2<br>$V_{DD}$ = 2.5V, $V_{IN}$ = $V_{DD}$/2 | | –1.0<br>–500 | mA<br>µA |
| $I_{LI}$ | Input leakage current, port 0, $\overline{EA}$ | $V_{SS}$ < $V_I$ < $V_{DD}$ | | ±10 | µA |
| $R_{RST}$ | Internal reset pull-down resistor | | 10 | 200 | kΩ |

**NOTES:**

1. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r$ = $t_f$ = 10ns; $V_{IL}$ = $V_{SS}$, $V_{IH}$ = $V_{DD}$; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = $V_{DD}$; all open drain outputs connected to $V_{SS}$.
2. The idle supply current is measured with all output pins disconnected; XTAL1 driven with $t_r$ = $t_f$ = 10ns; $V_{IL}$ = $V_{SS}$, $V_{IH}$ = $V_{DD}$; XTAL2 not connected; $\overline{EA}$ = Port 0 = $V_{DD}$; RST = $V_{SS}$; all open drain outputs connected to $V_{SS}$.
3. The power-down current is measured with all output pins disconnected; XTAL1 not connected; $\overline{EA}$ = port 0 = $V_{DD}$; RST = $V_{SS}$; all open-drain outputs connected to $V_{SS}$.
4. The RC-oscillator is not implemented in this version.
5. Circuits with "power-on reset" option "OFF" are tested at $V_{DDMIN}$ = 1.8V, with option "ON" (typically 1.3V) are tested at $V_{DDMIN}$ = 2.3V.




**Typical Operating Current Versus Supply and Frequency (32kHz–1.2MHz) at +25°C**

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = –40°C to +85°C, $V_{SS}$ = 0V[1, 2]

| SYMBOL | FIGURE | PARAMETER | 12MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| **Program Memory** | | | | | | | |
| $1/t_{CLCL}$ | | Oscillator frequency | | | 0 | 20 | MHz |
| $t_{LL}$ | 10 | ALE pulse width | 127 | | $2t_{CLCL}$–40 | | ns |
| $t_{AL}$ | 10 | Address valid to ALE low | 43 | | $t_{CLCL}$–40 | | ns |
| $t_{LA}$ | 10 | Address hold after ALE low | 48 | | $t_{CLCL}$–35 | | ns |
| $t_{LIV}$ | 10 | ALE low to valid instruction in | | 233 | | $4t_{CLCL}$–100 | ns |
| $t_{LC}$ | 10 | ALE low to $\overline{PSEN}$ low | 58 | | $t_{CLCL}$–25 | | ns |
| $t_{CC}$ | 10 | $\overline{PSEN}$ pulse width | 215 | | $3t_{CLCL}$–35 | | ns |
| $t_{CIV}$ | 10 | $\overline{PSEN}$ low to valid instruction in | | 125 | | $3t_{CLCL}$–125 | ns |
| $t_{CI}$ | 10 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{CIF}$ | 10 | Input instruction float after $\overline{PSEN}$ | | 63 | | $t_{CLCL}$–20 | ns |
| $t_{AVI}$ | 10 | Address to valid instruction in | | 302 | | $5t_{CLCL}$–115 | ns |
| $t_{AFC}$ | 10 | $\overline{PSEN}$ low to address float | 0 | | 0 | | ns |
| **Data Memory** | | | | | | | |
| $t_{RR}$ | 11 | $\overline{RD}$ pulse width | 400 | | $6t_{CLCL}$–100 | | ns |
| $t_{WW}$ | 12 | $\overline{WR}$ pulse width | 400 | | $6t_{CLCL}$–100 | | ns |
| $t_{LA}$ | 11, 12 | Address hold time after ALE | 48 | – | $t_{CLCL}$–35 | – | ns |
| $t_{RD}$ | 11 | $\overline{RD}$ low to valid data in | | 250 | | $5t_{CLCL}$–165 | ns |
| $t_{DFR}$ | 11 | Data float after $\overline{RD}$ | | 97 | | $2t_{CLCL}$–70 | ns |
| $t_{LD}$ | 11 | ALE low to valid data in | | 517 | | $8t_{CLCL}$–150 | ns |
| $t_{AD}$ | 11 | Address to valid data in | | 585 | | $9t_{CLCL}$–165 | ns |
| $t_{LW}$ | 11, 12 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 200 | 300 | $3t_{CLCL}$–50 | $3t_{CLCL}$+50 | ns |
| $t_{AW}$ | 11, 12 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 203 | | $4t_{CLCL}$–130 | | ns |
| $t_{DWX}$ | 12 | Data valid to $\overline{WR}$ transition | 23 | | $t_{CLCL}$–60 | | ns |
| $t_{DW}$ | 11 | Data valid to $\overline{WR}$ | 433 | – | $7t_{CLCL}$–150 | – | ns |
| $t_{WD}$ | 12 | Data hold after $\overline{WR}$ | 33 | | $t_{CLCL}$–50 | | ns |
| $t_{AFR}$ | 11 | $\overline{RD}$ low to address float[3] | | 12 | | 12 | ns |
| $t_{WHLH}$ | 11, 12 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 43 | 123 | $t_{CLCL}$–40 | $t_{CLCL}$+40 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 50pF, load capacitance for all other outputs = 40pF.
3. Interfacing the 8XCL410 to devices with float time up to 75ns is permitted. This limited bus connection will not cause damage to port 0 drivers.

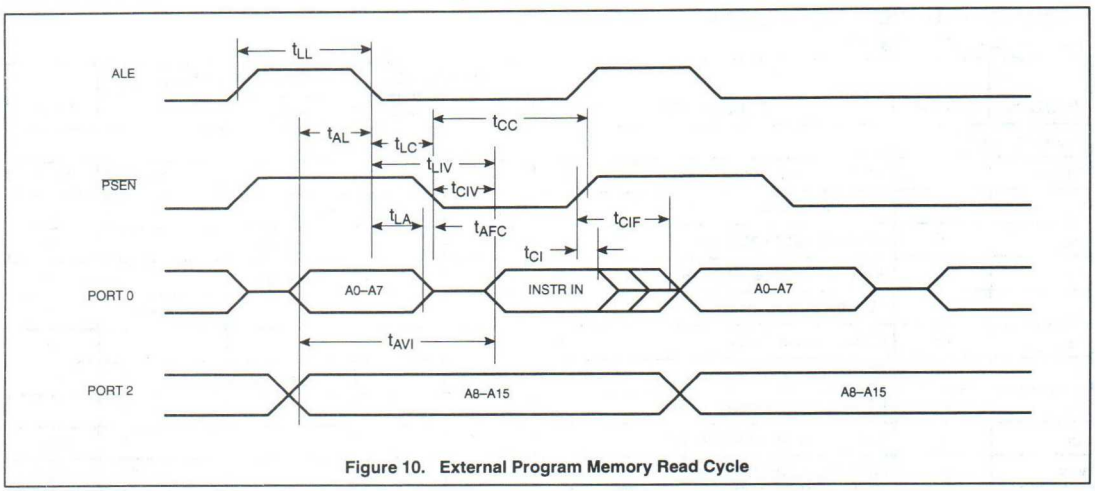



**Figure 10. External Program Memory Read Cycle**

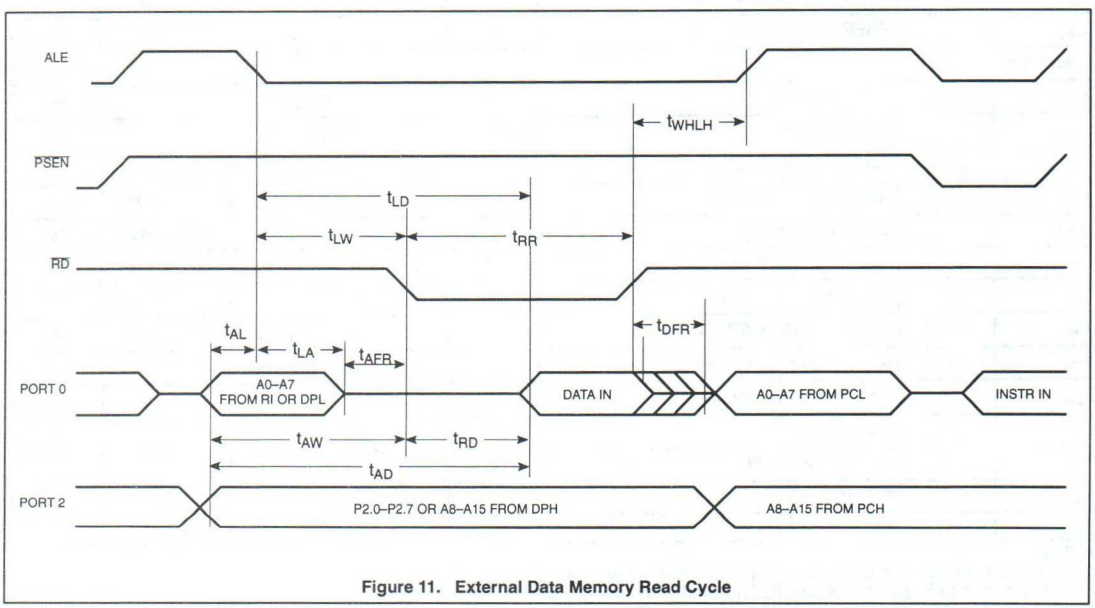



**Figure 11. External Data Memory Read Cycle**

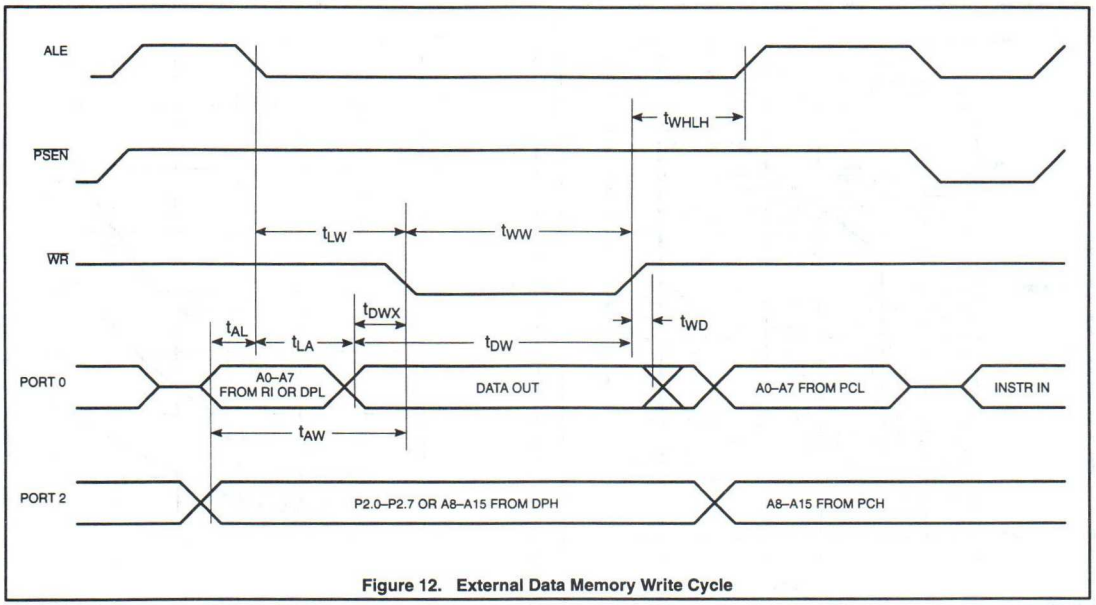



**Figure 12. External Data Memory Write Cycle**

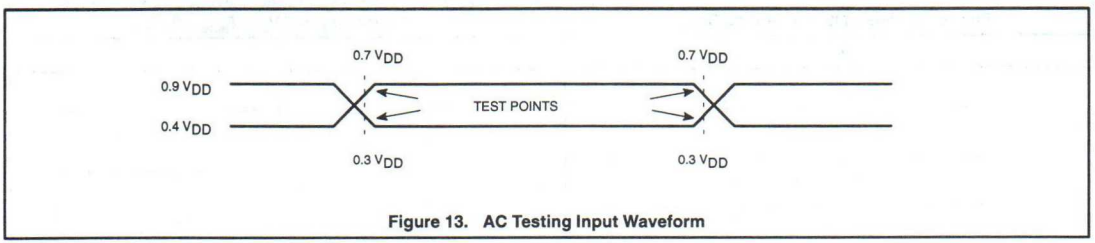



**Figure 13. AC Testing Input Waveform**

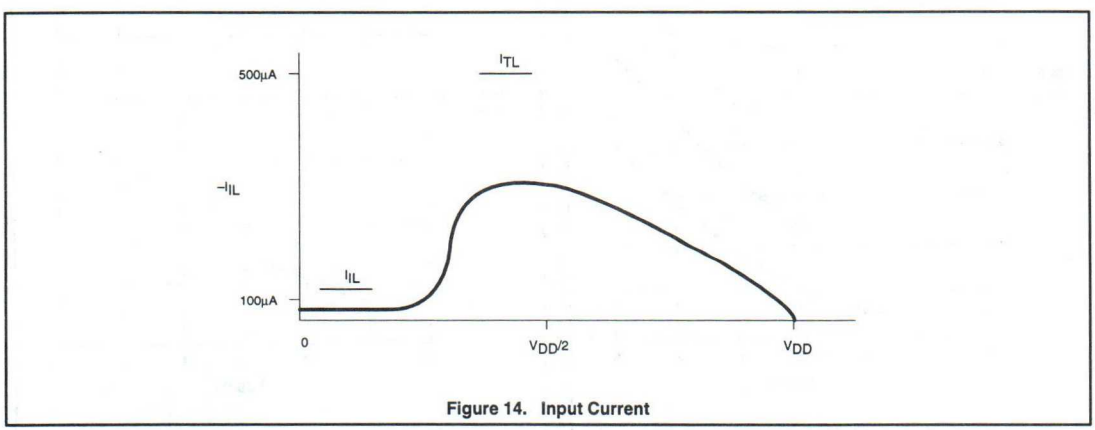
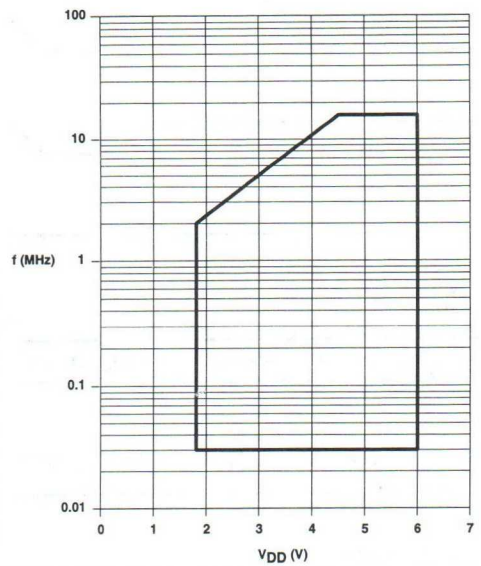



**Figure 14. Input Current**




NOTE: Below 32kHz, clock has to be supplied externally.

**Figure 15. Frequency Operating Range**

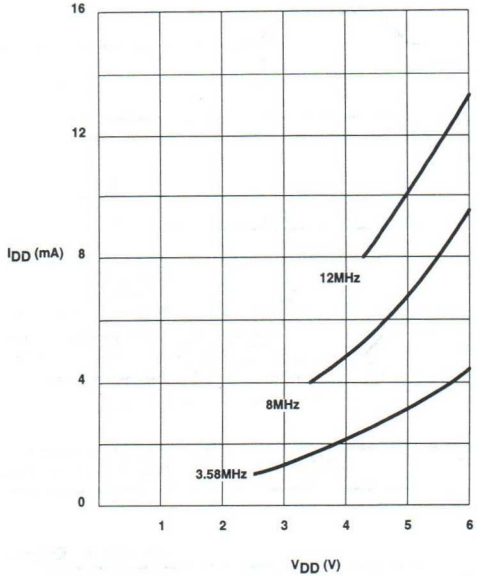



**Figure 16. Typical Operating Current as a Function of Frequency and $V_{DD}$, $T_{amb}$ = 25°C**

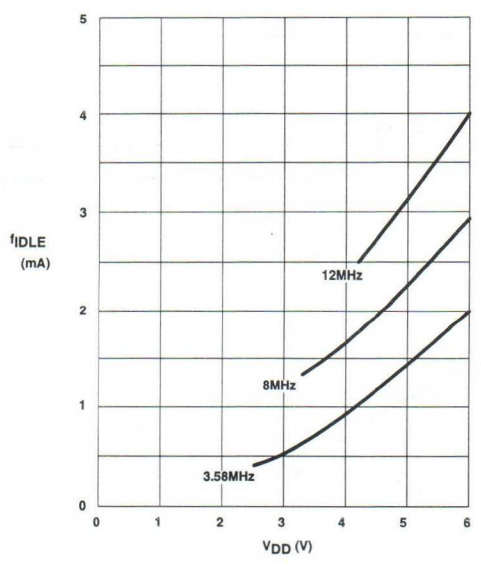



**Figure 17. Typical Idle Current as a Function of Frequency and $V_{DD}$, $T_{amb}$ = 25°C**

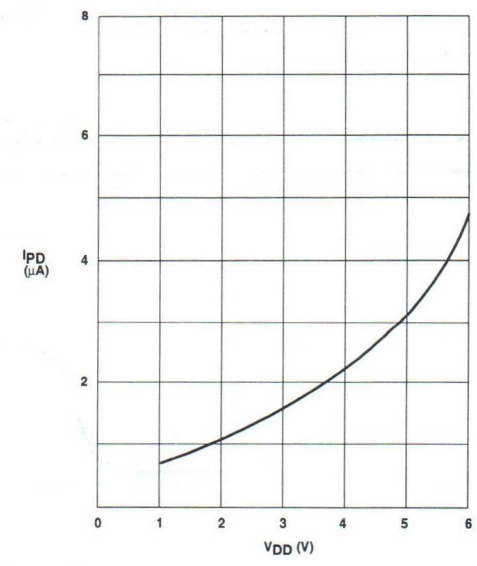



**Figure 18. Typical Power-Down Current Vs. Frequency and $V_{DD}$, $T_{amb}$ = 25°C**

# Low voltage/low power single-chip 8-bit microcontroller with I²C

# 80CL410/83CL410

## PIGGYBACK SPECIFICATION

The differences between the masked version and the piggyback are described herein.

## General Description

The P85CL000HFZ is a piggy-back version with 256 bytes of RAM used for emulation of the P83CL410 microcontroller. The P85CL000HFZ is manufactured in an advanced CMOS technology. The instruction set of the P85CL000HFZ is based on that of the 8051. The device has low power consumption and a wide supply voltage range. The P85CL000HFZ has two software selectable modes of reduced activity for further power reduction: Idle and Power-down. For timing and AC/DC characteristics, please refer to the P83CL410 specifications.

## Features

- Full static 80C51 CPU
- 8-bit CPU, RAM, I/O in a single 40-lead DIP
- Socket for up to 16k external EPROM
- 256 bytes RAM, expandable externally to 64K bytes
- Four 8-bit ports, 32 I/O lines
- Two 16-bit timer/event counters
- External memory expandable up to 128K, external ROM up to 64K and/or RAM up to 64K
- Thirteen source, thirteen vector interrupt structure with two priority levels
- Full duplex serial port (UART)
- I²C-bus interface for serial transfer on two lines
- Enhanced architecture with:
  - non-page oriented instructions
  - direct addressing
  - four eight byte RAM register banks
  - stack depth up to 128 bytes
  - multiply, divide, subtract and compare instructions
- STOP and IDLE instructions
- Wake-up via external interrupts at port 1
- Single supply voltage of 1.8V to 6.0V
- On-chip oscillator (option: oscillator 4)
- Very low current consumption
- Operating temperature range: –40 to +85°C

## STANDARD PIGGYBACK

Types: P85CL000HFZ
Emulation for: P83CL410, P80CL51

List of differences between masked microcontroller and corresponding piggyback:

| PARAMETER | MASKED CONTROLLER | PIGGYBACK |
|---|---|---|
| RAM size | 128 | 256 |
| ROM size | 4k | EPROM size dependent (max 16k) |
| Port option | 1, 2, 3 | 1 |
| Oscillator option | Osc. 1, 2, 3, 4, RC | Osc. 4 |
| Mech. dimensions | Standard Dual In-Line, Small Outline | See SOT158A |
| Current cons. | $I_{DD}$ | $I_{DD}$ (OSC. 4) + $I_{EPROM}$ |
| Voltage range | full | full, limited by EPROM |
| ESD | specification | not tested (different package) |

I²C BUS

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

## DESCRIPTION

The Philips 8XC451 is an I/O expanded single-chip microcontroller fabricated with Philips high-density CMOS technology. Philips epitaxial substrate minimizes latch-up sensitivity.

The 8XC451 (includes the 80C451, 87C451 and 83C451) is a functional extension of the 87C51 microcontroller with three additional I/O ports and four I/O control lines for a total of 68 pins. Four control lines associated with port 6 facilitate high-speed asynchronous I/O functions.

The 8XC451 includes a 4k × 8 ROM (83C451) EPROM (87C451), a 128 × 8 RAM, 56 I/O, two 16-bit timer/counters, a five source, two priority level, nested interrupt structure, a serial I/O port for either a full duplex UART, I/O expansion, or multi-processor communications, and on-chip oscillator and clock circuits. The 80C451 ROMless version includes all of the 83C451 features except the on-board 4k × 8 ROM.

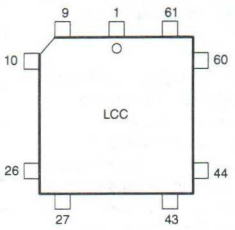
The 87C451 has 4k of EPROM on-chip as program memory and is otherwise identical to the 83C451.

The 8XC451 has two software selectable modes of reduced activity for further power reduction; idle mode and power-down mode. Idle mode freezes the CPU while allowing the RAM, timers, serial port, and interrupt system to continue functioning. Power-down mode freezes the oscillator, causing all other chip functions to be inoperative while maintaining the RAM contents.

## FEATURES

- 80C51 based architecture
- Seven 8-bit I/O ports
- Port 6 features:
  - Eight data pins
  - Four control pins
  - Direct MPU bus interface
  - Parallel printer interface
- On the microcontroller:
  - 4k × 8 ROM (83C451)
    4k × 8 EPROM (87C451)
    ROMless version (80C451)
  - 128 × 8 RAM
  - Two 16-bit counter/timers
  - Two external interrupts
- External memory addressing capability
  - 64k ROM and 64k RAM
- Low power consumption:
  - Normal operation: less than 24mA at 5V, 12MHz
  - Idle mode
  - Power-down mode

## PIN CONFIGURATION




| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | $\overline{EA}$/Vpp | 24 | P4.2 | 47 | P5.3 |
| 2 | P2.0/A8 | 25 | P4.1 | 48 | P5.4 |
| 3 | P2.1/A9 | 26 | P4.0 | 49 | P5.5 |
| 4 | P2.2/A10 | 27 | P1.0 | 50 | P5.6 |
| 5 | P2.3/A11 | 28 | P1.1 | 51 | P5.7 |
| 6 | P2.4/A12 | 29 | P1.2 | 52 | XTAL2 |
| 7 | P2.5/A13 | 30 | P1.3 | 53 | XTAL1 |
| 8 | P2.6/A14 | 31 | P1.4 | 54 | $V_{SS}$ |
| 9 | P2.7/A15 | 32 | P1.5 | 55 | $\overline{ODS}$ |
| 10 | P0.7/AD7 | 33 | P1.6 | 56 | $\overline{IDS}$ |
| 11 | P0.6/AD6 | 34 | P1.7 | 57 | BFLAG |
| 12 | P0.5/AD5 | 35 | RST | 58 | AFLAG |
| 13 | P0.4/AD4 | 36 | P3.0/RxD | 59 | P6.0 |
| 14 | P0.3/AD3 | 37 | P3.1/TxD | 60 | P6.1 |
| 15 | P0.2/AD2 | 38 | P3.2/$\overline{INT0}$ | 61 | P6.2 |
| 16 | P0.1/AD1 | 39 | P3.3/$\overline{INT1}$ | 62 | P6.3 |
| 17 | P0.0/AD0 | 40 | P3.4/T0 | 63 | P6.4 |
| 18 | $V_{CC}$ | 41 | P3.5/T1 | 64 | P6.5 |
| 19 | P4.7 | 42 | P3.6/$\overline{WR}$ | 65 | P6.6 |
| 20 | P4.6 | 43 | P3.7/$\overline{RD}$ | 66 | P6.7 |
| 21 | P4.5 | 44 | P5.0 | 67 | $\overline{PSEN}$ |
| 22 | P4.4 | 45 | P5.1 | 68 | ALE/$\overline{PROG}$ |
| 23 | P4.3 | 46 | P5.2 | | |

SU00084A

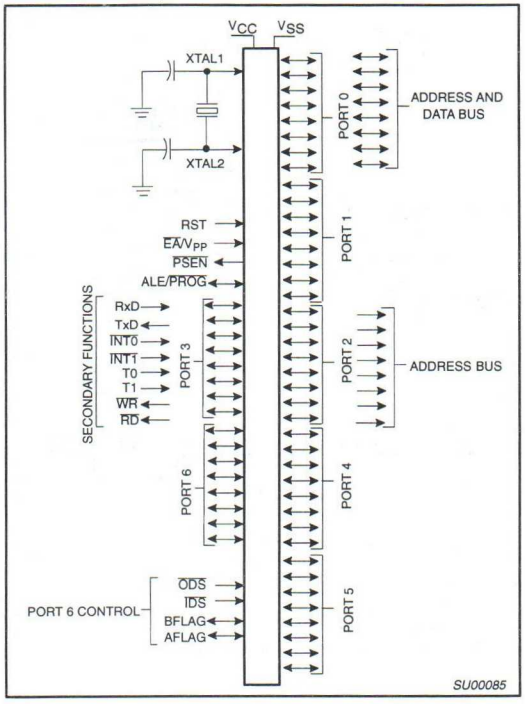
## ORDERING INFORMATION

| ROMless | ROM | EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQ MHz | DRAWING NUMBER |
|---|---|---|---|---|---|---|
| SC80C451CCA68 | SC83C451CCA68 | SC87C451CCA68 | OTP | 0 to +70, Plastic Leaded Chip Carrier, | 3.5 to 12 | SOT188-3 |
| SC80C451CGA68 | SC83C451CGA68 | SC87C451CGA68 | OTP | 0 to +70, Plastic Leaded Chip Carrier | 3.5 to 16 | SOT188-3 |
| SC80C451ACA68 | SC83C451ACA68 | SC87C451ACA68 | OTP | –40 to +85, Plastic Leaded Chip Carrier | 3.5 to 12 | SOT188-3 |
| SC80C451AGA68 | SC83C451AGA68 | SC87C451AGA68 | OTP | –40 to +85, Plastic Leaded Chip Carrier | 3.5 to 16 | SOT188-3 |
| | | SC87C451CGK68 | OTP | 0 to +70, Ceramic Leaded Chip Carrier | 3.5 to 16 | 1473A |

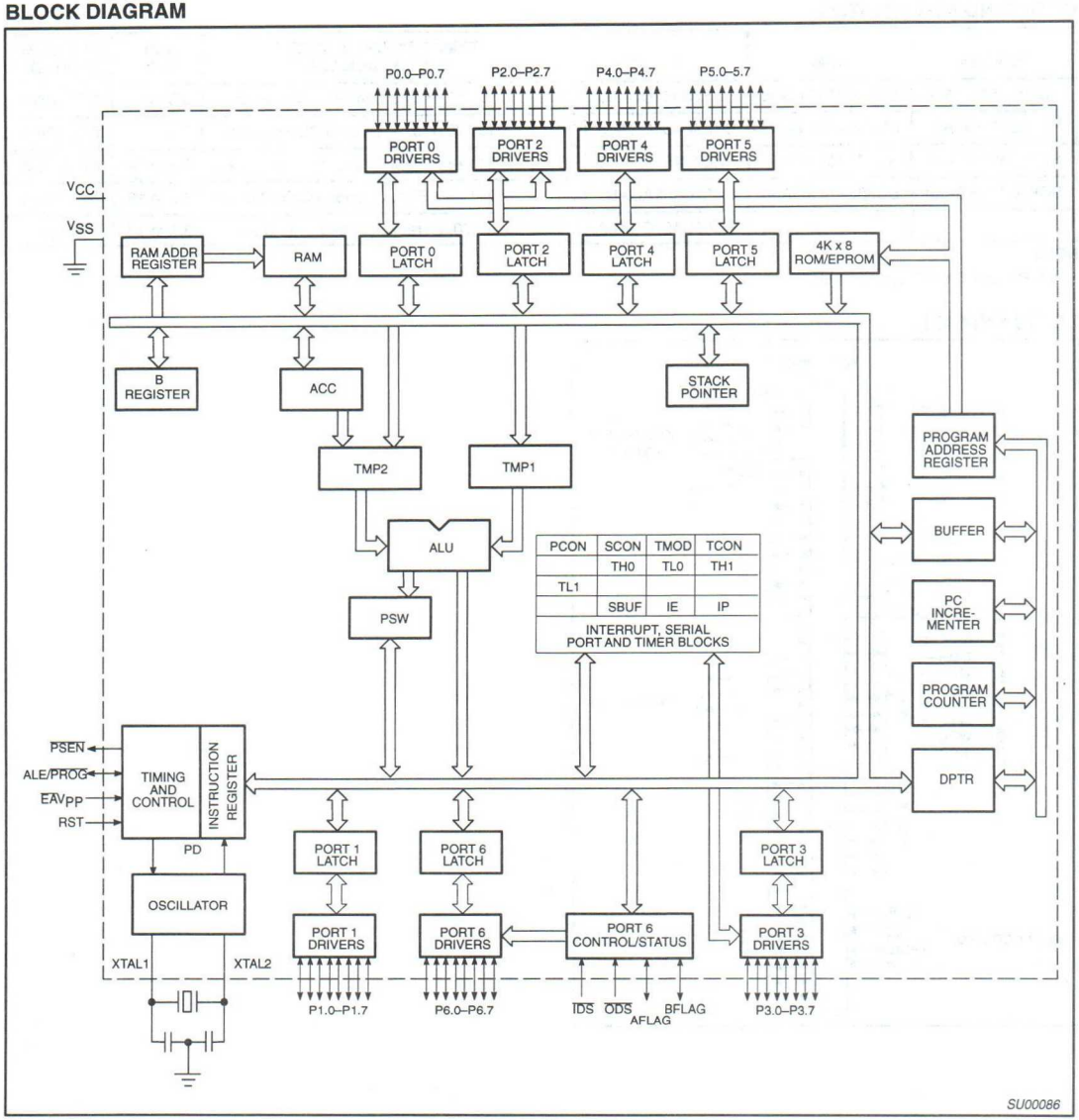
**NOTE:**

1. OTP = One Time Programmable

## LOGIC SYMBOL



## BLOCK DIAGRAM



## PIN DESCRIPTION

| MNEMONIC | PIN NO. | TYPE | NAME AND FUNCTION |
|---|---|---|---|
| $V_{SS}$ | 54 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 18 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 17-10 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 is also the multiplexed data and low-order address bus during accesses to external memory. External pull-ups are required during program verification. Port 0 can sink/source eight LS TTL inputs. |
| P1.0–P1.7 | 27-34 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 receives the low-order address bytes during program memory verification. Port 1 can sink/source three LS TTL inputs, and drive CMOS inputs without external pull-ups. |
| P2.0–P2.7 | 2-9 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 emits the high-order address bytes during access to external memory and receives the high-order address bits and control signals during program verification. Port 2 can sink/source three LS TTL inputs, and drive CMOS inputs without external pull-ups. |
| P3.0–P3.7 | 36-43 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 can sink/source three LS TTL inputs, and drive CMOS inputs without external pull-ups. Port 3 also serves the special functions listed below: |
| | 36 | I | **RxD (P3.0):** Serial input port |
| | 37 | O | **TxD (P3.1):** Serial output port |
| | 38 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 39 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 40 | I | **T0 (P3.4):** Timer 0 external input |
| | 41 | I | **T1 (P3.5):** Timer 1 external input |
| | 42 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 43 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| P4.0–P4.7 | 26-19 | I/O | **Port 4:** Port 4 is a 8-bit (LCC) bidirectional I/O port with internal pull-ups. Port 4 can sink/source three LS TTL inputs and drive CMOS inputs without external pull-ups. |
| P5.0–P5.7 | 44-51 | I/O | **Port 5:** Port 5 is a 8-bit (LCC) bidirectional I/O port with internal pull-ups. Port 5 can sink/source three LS TTL inputs and drive CMOS inputs without external pull-ups. |
| P6.0–P6.7 | 59-66 | I/O | **Port 6:** Port 6 is a specialized 8-bit bidirectional I/O port with internal pull-ups. This special port can sink/source three LS TTL inputs and drive CMOS inputs without external pull-ups. Port 6 can be used in a strobed or non-strobed mode of operation. Port 6 works in conjunction with four control pins that serve the functions listed below: |
| $\overline{ODS}$ | 55 | I | **$\overline{ODS}$:** Output data strobe |
| $\overline{IDS}$ | 56 | I | **$\overline{IDS}$:** Input data strobe |
| BFLAG | 57 | I/O | **BFLAG:** Bidirectional I/O pin with internal pull-ups |
| AFLAG | 58 | I/O | **AFLAG:** Bidirectional I/O pin with internal pull-ups |
| RST | 35 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal pull-down resistor permits a power-on reset using only an external capacitor connected to $V_{CC}$. |
| ALE/$\overline{PROG}$ | 68 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. ALE is activated at a constant rate of 1/6 the oscillator frequency except during an external data memory access, at which time one ALE is skipped. ALE can sink/source three LS TTL inputs and drive CMOS inputs without external pull-ups. This pin is also the program pulse during EPROM programming. |
| $\overline{PSEN}$ | 67 | O | **Program Store Enable:** The read strobe to external program memory. $\overline{PSEN}$ is activated twice each machine cycle during fetches from external program memory. However, when executing out of external program memory, two activations of $\overline{PSEN}$ are skipped during each access to external program memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. $\overline{PSEN}$ can sink/source eight LS TTL inputs and drive CMOS inputs without an external pull-up. This pin should be tied low during programming. |
| $\overline{EA}$/$V_{PP}$ | 1 | I | **Instruction Execution Control/Programming Supply Voltage:** When $\overline{EA}$ is held high, the CPU executes out of internal program memory, unless the program counter exceeds 0FFFH. When $\overline{EA}$ is held low, the CPU executes out of external program memory. $\overline{EA}$ must never be allowed to float. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. |
| XTAL1 | 53 | I | **Crystal 1:** Input to the inverting oscillator amplifier that forms the oscillator. This input receives the external oscillator when an external oscillator is used. |
| XTAL2 | 52 | O | **Crystal 2:** An output of the inverting amplifier that forms the oscillator. This pin should be floated when an external oscillator is used. |

## I/O Port Structure

The 8XC451 has a total of seven parallel I/O ports. The first four ports, P0 through P3, are identical in function to those present on the 80C51 family. The added ports 4 and 5 are identical in function to port 1; that is, they are standard quasi-bidirectional ports with no alternate functions and the standard output drive characteristics. Port 6 is a specialized 8-bit bidirectional I/O port with internal pullups.

## Ports 4 and 5

Ports 4 and 5 are bidirectional I/O ports with internal pull-ups. Port 4 is an 8-bit port. Port 4 and port 5 pins with ones written to them, are pulled high by the internal pull-ups, and in that state can be used as inputs. Port 4 and 5 are addressed at the special function register addresses shown in Table 1.

## Port 6

Port 6 is a special 8-bit bidirectional I/O port with internal pull-ups (see Figure 1). This special port can sink/source three LS TTL inputs and drive CMOS inputs without external pullups. The flexibility of this port facilitates high-speed parallel data communications. This port can be used as a standard I/O port, or in strobed modes of operation in conjunction with four special control lines: $\overline{ODS}$, $\overline{IDS}$, AFLAG, and BFLAG. Port 6 operating modes are controlled by the port 6 control status register (CSR). Port 6 and the CSR are addressed at the special function register addresses shown in Table 1. The following four control pins are used in conjunction with port 6:

**$\overline{ODS}$** – Output data strobe (Active Low) for port 6. $\overline{ODS}$ can be programmed to control the port 6 output drivers and the output buffer full flag (OBF), or to clear only the OBF flag bit in the CSR (output-always mode). $\overline{ODS}$ is active low for output driver control. the OBF flag can be programmed to be cleared on the negative or positive edge of $\overline{ODS}$.

**$\overline{IDS}$** – Input data strobe (Active Low) for port 6. $\overline{IDS}$ is used to control the port 6 input latch and input buffer full flag (IBF) bit in the CSR. The input data latch can be programmed to be transparent when $\overline{IDS}$ is low and latched on the positive transition of $\overline{IDS}$, or to latch only on the positive transition of $\overline{IDS}$. Correspondingly, the IBF flag is set on the negative or positive transition of $\overline{IDS}$.

**BFLAG** – BFLAG is a bidirectional I/O pin which can be programmed to be an output, set high or low under program control, or to output the state of the input buffer full flag. BFLAG can also be programmed to input an enable signal for port 6. When BFLAG is used as an enable input, port 6 output drivers are in the high-impedance state, and the input latch does not respond to the $\overline{IDS}$ strobe when BFLAG is high. Both features are enabled when BFLAG is low. This feature facilitates the use of the SC8XC451 in bused multiprocessor systems.

**AFLAG** – AFLAG is a bidirectional I/O pin which can be programmed to be an output set high or low under program control, or to output the state of the output buffer full flag. AFLAG can also be programmed to be an input which selects whether the contents of the output buffer, or the contents of the port 6 control status register will output on port 6. This feature grants complete port 6 status to external devices.

Port 6 can be used in a number of different ways to facilitate data communication. It can be used as a processor bus interface, as a standard quasi-bidirectional I/O port, or as a parallel printer port (either polled or interrupt driven).

## Processor Bus Interface

Port 6 allows the use of an 8XC451 as an element on a microprocessor type bus. The host processor could be a general purpose MPU or the data bus of a microcontroller like the 8XC451 itself. Setting up the 8XC451 as a processor bus interface allows single or multiple microcontrollers to be used on a bus as flexible peripheral processing elements. Applications can include: keyboard scanners, serial I/O controllers, servo controllers, etc.

On reset, port 6 is programmed correctly (that is, Special Function registers CSR and P6) for use as a bus interface. This prevents the interface from disrupting data on the bus of a host processor during power-up.

## Standard Quasi-bidirectional I/O Port

To use port 6 as a common I/O port, all of the control pins should be tied to ground. On hardware reset, bits 2-7 of the CSR are set to one. With the control pins grounded, the port's operation and electrical characteristics will be identical to port 1 on the 80C51. No further software initialization is required.

## Parallel Printer Port

The 8XC451 has the capacity to permit all of the intelligent features of a common printer to be handled by a single chip. The features of port 6 allow a parallel port to be designed with only line driving and receiving chips required as additional hardware. The onboard UART allows RS232 interfacing with only level shifting chips added. The 8-bit parallel ports 0 to 6 are ample to drive onboard control functions, even when ports are used for external memory access, interrupts, and other functions. The RAM addressing ability of ports 0 to 2 can be used to address up to 64k bytes of a hardware buffer/spooler.

In addition, either end of a parallel interface can be implemented using port 6, and the interfaces can be interrupt driven or polled in either case. For more detailed information on port 6 usage, refer to the application notes entitled "80C451 Operation of Port 6" and "256k Centronics Printer Buffer Using the SC87C451 Microcontroller."

## CONTROL STATUS REGISTER

The control status register (CSR) establishes the mode of operation for port 6 and indicates the current status of port 6 I/O registers. All control status register bits can be read and written by the CPU, except bits 0 and 1, which are read only. Reset writes ones to bits 2 through 7, and writes zeros to bits 0 and 1 (see Table 3).

**CSR.0 Input Buffer Full Flag (IBF) (Read Only)** – The IBF bit is set to a logic 1 when port 6 data is loaded into the input buffer under control of $\overline{IDS}$. This can occur on the negative or positive edge of $\overline{IDS}$, as determined by CSR.2 IBF is cleared when the CPU reads the input buffer register.

**CSR.1 Output Buffer Full Flag (OBF) (Read Only)** – The OBF flag is set to a logic 1 when the CPU writes to the port 6 output data buffer. OBF is cleared by the positive or negative edge of $\overline{ODS}$, as determined by CSR.3.

**CSR.2 IDS Mode Select (IDSM)** – When CSR.2 = 0, a low-to-high transition on the $\overline{IDS}$ pin sets the IBF flag. The Port 6 input buffer is loaded on the $\overline{IDS}$ positive edge. When CSR.2 = 1, a high-to-low transition on the $\overline{IDS}$ pin sets the IBF flag. Port 6 input buffer is transparent when $\overline{IDS}$ is low, and latched when $\overline{IDS}$ is high.

**CSR.3 Output Buffer Full Flag Clear Mode (OBFC) –** When CSR.3 = 1, the positive edge of the $\overline{ODS}$ input clears the OBF flag. When CSR.3 = 0, the negative edge of the $\overline{ODS}$ input clears the OBF flag.

**CSR.4, CSR.5 AFLAG Mode Select**
**(MA0, MA1) –** Bits 4 and 5 select the mode of operation for the AFLAG pin as follows:

| MA1 | MA0 | AFLAG Function |
|---|---|---|
| 0 | 0 | Logic 0 output |
| 0 | 1 | Logic 1 output |
| 1 | 0 | OBF flag output (CSR.1) |
| 1 | 1 | Select (SEL) input mode |

The select (SEL) input mode is used to determine whether the port 6 data register or the control status register is output on port 6. When the select feature is enabled, the AFLAG input controls the source of port 6 output data. A logic 0 on AFLAG input selects the port 6 data register, and a logic 1 on AFLAG input selects the control status register.

**CSR.6, CSR.7 BFLAG Mode Select**
**(MB0, MB1) –** Bits 6 and 7 select the mode operation as follows:

| MB1 | MB0 | BFLAG Function |
|---|---|---|
| 0 | 0 | Logic 0 output |
| 0 | 1 | Logic 1 output |
| 1 | 0 | IBF flag output (CSR.0) |
| 1 | 1 | Port enable ($\overline{PE}$) |

In the port enable mode, $\overline{IDS}$ and $\overline{ODS}$ inputs are disabled when BFLAG input is high. When the BFLAG input is low, the port is enabled for I/O.

## SPECIAL FUNCTION REGISTER ADDRESSES

The SFRs are identical to those of the standard 80C51 with the exception of four registers that have been added to allow control of the three additional I/O ports P4, P5, and P6. The additional registers are P4, P5, P6, and CSR. Registers P4, P5, and P6 function as port latches for ports 4, 5, and 6, respectively. These registers operate identically to those for ports 0 through 3 of the 80C51.

**Table 1. Special Function Register Addresses**

| REGISTER ADDRESS | | | BIT ADDRESS | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| NAME | SYMBOL | ADDRESS | MSB | | | | | | | LSB |
| Port 4 | P4 | C0 | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 |
| Port 5 | P5 | C8 | CF | CE | CD | CC | CB | CA | C9 | C8 |
| Port 6 data | P6 | D8 | DF | DE | DD | DC | DB | DA | D9 | D8 |
| Port 6 control status | CSR | E8 | EF | EE | ED | EC | EB | EA | E9 | E8 |

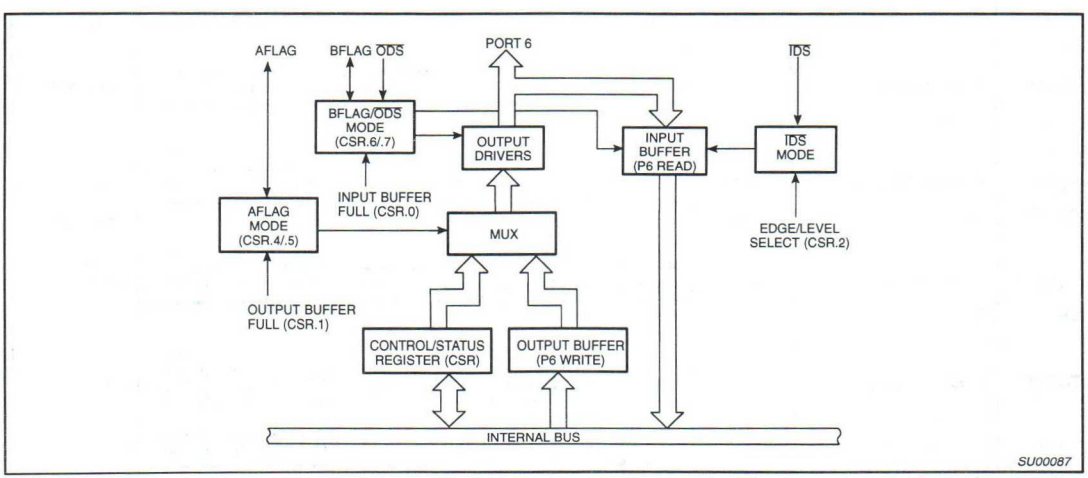



**Figure 1. Port 6 Block Diagram**

**Table 2. 8X451 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT NAMES AND ADDRESSES MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| | | | EF | EE | ED | EC | EB | EA | E9 | E8 | |
| CSR*# | Port 6 command/status | E8H | MB1 | MB0 | MA1 | MA0 | OBFC | IDSM | OBF | IBF | FCH |
| DPTR | Data pointer (2 bytes) | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP* | Interrupt priority | B8H | – | – | – | PS | PT1 | PX1 | PT0 | PX0 | xxx00000B |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE* | Interrupt enable | A8H | EA | – | – | ES | ET1 | EX1 | ET0 | EX0 | 0xx00000B |
| P0* | Port 0 | 80H | 87 | B6 | 85 | 84 | 83 | 82 | 81 | 80 | FFH |
| P1* | Port 1 | 90H | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | FFH |
| P2* | Port 2 | A0H | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | FFH |
| P3* | Port 3 | B0H | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | FFH |
| P4*# | Port 4 | C0H | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 | FFH |
| P5*# | Port 5 | C8H | CF | CE | CD | CC | CB | CA | C9 | C8 | FFH |
| P6*# | Port 6 | D8H | DF | DE | DD | DC | DB | DA | D9 | D8 | FFH |
| PCON | Power control | 87H | SMOD | – | – | – | GF1 | GF0 | PD | IDL | 0xxx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| SBUF | Serial data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial port control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer/counter control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| TMOD | Timer/counter mode | 89H | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 | 00H |
| TH0 | Timer 0 high byte | 8CH | | | | | | | | | 00H |
| TH1 | Timer 1 high byte | 8DH | | | | | | | | | 00H |
| TL0 | Timer 0 low byte | 8AH | | | | | | | | | 00H |
| TL1 | Timer 1 low byte | 8BH | | | | | | | | | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

**Table 3. Control Status Register (CSR)**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| **MB1** | **MB0** | **MA1** | **MA0** | **OBFC** | **IDSM** | **OBF** | **IBF** |
| BFLAG Mode Select | | AFLAG Mode Select | | Output Buffer Flag Clear Mode | Input Data Strobe Mode | Output Buffer Flag Full | Input Buffer Flag Full |
| 0/0 = Logic 0 output*<br>0/1 = Logic 1 output*<br>1/0 = IBF output<br>1/1 = $\overline{PE}$ input<br>(0 = Select)<br>(1 = Disable I/O) | | 0/0 = Logic 0 output*<br>0/1 = Logic 1 output*<br>1/0 = OBF output<br>1/1 = SEL input<br>(0 = Select)<br>(1 = Control/status) | | 0 = Negative edge of $\overline{ODS}$<br>1 = Positive edge o $\overline{ODS}$ | 0 = Positive edge of $\overline{IDS}$<br>1 = Low level of $\overline{IDS}$ | 0 = Output data buffer empty<br>1 = Output data buffer full | 0 = Input data buffer empty<br>1 = Input data buffer full |

**NOTE:**
* Output-always mode: MB1 = 0, MA1 = 1, and MA0 = 0. In this mode, port 6 is always enabled for output. $\overline{ODS}$ only clears the OBF flag.

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | 0 to +70<br>–40 to +85 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.5 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS[1]

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V (87C451, 83C451, 80C451)

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYPICAL[1] | MAX | |
| $V_{IL}$ | Input low voltage; except $\overline{EA}$ | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | | 0 | | $0.2V_{CC}$–0.3 | V |
| $V_{IH}$ | Input high voltage; except XTAL1, RST | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage; XTAL1, RST | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage; ports 1, 2, 3, 4, 5, 6 | $I_{OL}$ = 1.6mA[2] | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage; port 0, ALE, $\overline{PSEN}$ | $I_{OL}$ = 3.2mA[2] | | | 0.45 | V |
| $V_{OH}$ | Output high voltage; ports 1, 2, 3, 4, 5, 6 | $I_{OH}$ = –60µA,<br>$I_{OH}$ = –25µA<br>$I_{OH}$ = –10µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$ | | | V<br>V<br>V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode, ALE, $\overline{PSEN}$)[3] | $I_{OH}$ = –800µA,<br>$I_{OH}$ = –300µA<br>$I_{OH}$ = –80µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$ | | | V<br>V<br>V |
| $I_{IL}$ | Logical 0 input current,; ports 1, 2, 3, 4, 5, 6 | $V_{IN}$ = 0.45V | | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current; ports 1, 2, 3, 4, 5, 6 | See note 4 | | | –650 | µA |
| $I_{LI}$ | Input leakage current; port 0 | $V_{IN} = V_{IL}$ or $V_{IH}$ | | | ±10 | µA |
| $I_{CC}$ | Power supply current:<br>Active mode @ 12MHz[5]<br>Idle mode @ 12MHz[5]<br>Power down mode | See note 6 | | <br>11.5<br>1.3<br>3 | <br>25<br>4<br>50 | <br>mA<br>mA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | 50 | | 300 | kΩ |
| $C_{IO}$ | Pin capacitance[7] | | | | 10 | pF |

**NOTES:**

1. Typical ratings are based on a limited number of samples taken from early manufacturing lots and are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
5. $I_{CCMAX}$ at other frequencies is given by:
   Active mode: $I_{CCMAX}$ = 0.94 X FREQ + 13.71
   Idle mode: $I_{CCMAX}$ = 0.14 X FREQ +2.31
   where FREQ is the external oscillator frequency in MHz. $I_{CCMAX}$ is given in mA. See Figure 13.
6. See Figures 14 through 17 for $I_{CC}$ test conditions.
7. $C_{IO}$ applies to ports 1 through 6, AFLAG, BFLAG, XTAL1, XTAL2.

## AC ELECTRICAL CHARACTERISTICS[1]

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V (87C451, 83C451, 80C451)[2]

| SYMBOL | FIGURE | PARAMETER | 12MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | | Oscillator frequency: **Speed Versions**<br>SC8XC451 C<br>SC8XC451 G | | | <br>3.5<br>3.5 | <br>12<br>16 | <br>MHz<br>MHz |
| $t_{LHLL}$ | 2 | ALE pulse width | 127 | | $2t_{CLCL}$–40 | | ns |
| $t_{AVLL}$ | 2 | Address valid to ALE low | 28 | | $t_{CLCL}$–55 | | ns |
| $t_{LLAX}$ | 2 | Address hold after ALE low | 48 | | $t_{CLCL}$–35 | | ns |
| $t_{LLIV}$ | 2 | ALE low to valid instruction in | | 234 | | $4t_{CLCL}$–100 | ns |
| $t_{LLPL}$ | 2 | ALE low to $\overline{PSEN}$ low | 43 | | $t_{CLCL}$–40 | | ns |
| $t_{PLPH}$ | 2 | $\overline{PSEN}$ pulse width | 205 | | $3t_{CLCL}$–45 | | ns |
| $t_{PLIV}$ | 2 | $\overline{PSEN}$ low to valid instruction in | | 145 | | $3t_{CLCL}$–105 | ns |
| $t_{PXIX}$ | 2 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 2 | Input instruction float after $\overline{PSEN}$ | | 59 | | $t_{CLCL}$–25 | ns |
| $t_{AVIV}$ | 2 | Address to valid instruction in | | 312 | | $5t_{CLCL}$–105 | ns |
| $t_{PLAZ}$ | 2 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 3, 4 | $\overline{RD}$ pulse width | 400 | | $6t_{CLCL}$–100 | | ns |
| $t_{WLWH}$ | 3, 4 | $\overline{WR}$ pulse width | 400 | | $6t_{CLCL}$–100 | | ns |
| $t_{RLDV}$ | 3, 4 | $\overline{RD}$ low to valid data in | | 252 | | $5t_{CLCL}$–165 | ns |
| $t_{RHDX}$ | 3, 4 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 3, 4 | Data float after $\overline{RD}$ | | 97 | | $2t_{CLCL}$–70 | ns |
| $t_{LLDV}$ | 3, 4 | ALE low to valid data in | | 517 | | $8t_{CLCL}$–150 | ns |
| $t_{AVDV}$ | 3, 4 | Address to valid data in | | 585 | | $9t_{CLCL}$–165 | ns |
| $t_{LLWL}$ | 3, 4 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 200 | 300 | $3t_{CLCL}$–50 | $3t_{CLCL}$+50 | ns |
| $t_{AVWL}$ | 3, 4 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 203 | | $4t_{CLCL}$–130 | | ns |
| $t_{QVWX}$ | 3, 4 | Data valid to $\overline{WR}$ transition | 23 | | $t_{CLCL}$–60 | | ns |
| $t_{WHQX}$ | 3, 4 | Data hold after $\overline{WR}$ | 33 | | $t_{CLCL}$–50 | | ns |
| $t_{RLAZ}$ | 3, 4 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 3, 4 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 43 | 123 | $t_{CLCL}$–40 | $t_{CLCL}$+40 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 5 | Serial port clock cycle time | 1.0 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 5 | Output data setup to clock rising edge | 700 | | $10t_{CLCL}$–133 | | ns |
| $t_{XHQX}$ | 5 | Output data hold after clock rising edge | 50 | | $2t_{CLCL}$–117 | | ns |
| $t_{XHDX}$ | 5 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 5 | Clock rising edge to input data valid | | 700 | | $10t_{CLCL}$–133 | ns |

**NOTES: SEE NEXT PAGE**

## AC ELECTRICAL CHARACTERISTICS[1] (continued)

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V (87C451, 83C451, 80C451)[2]

| SYMBOL | FIGURE | PARAMETER | 12MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| **Port 6 input (input rise and fall times = 5ns)** | | | | | | | |
| $t_{FLFH}$ | 8 | $\overline{PE}$ width | 270 | | $3t_{CLCL}$+20 | | ns |
| $t_{ILIH}$ | 8 | $\overline{IDS}$ width | 270 | | $3t_{CLCL}$+20 | | ns |
| $t_{DVIH}$ | 8 | Data setup to $\overline{IDS}$ high or $\overline{PE}$ high | 0 | | 0 | | ns |
| $t_{IHDX}$ | 8 | Data hold after $\overline{IDS}$ high or $\overline{PE}$ high | 30 | | 30 | | ns |
| $t_{IVFV}$ | 9 | $\overline{IDS}$ to BFLAG (IBF) delay | | 130 | | 130 | ns |
| **Port 6 output** | | | | | | | |
| $t_{OLOH}$ | 6 | $\overline{ODS}$ width | 270 | | $3t_{CLCL}$+20 | | ns |
| $t_{FVDV}$ | 7 | SEL to data out delay | | 85 | | 85 | ns |
| $t_{OLDV}$ | 6 | $\overline{ODS}$ to data out delay | | 80 | | 80 | ns |
| $t_{OHDZ}$ | 6 | $\overline{ODS}$ to data float delay | | 35 | | 35 | ns |
| $t_{OVFV}$ | 6 | $\overline{ODS}$ to AFLAG (OBF) delay | | 100 | | 100 | ns |
| $t_{FLDV}$ | 6 | $\overline{PE}$ to data out delay | | 120 | | 120 | ns |
| $t_{OHFH}$ | 7 | $\overline{ODS}$ to AFLAG (SEL) delay | 100 | | 100 | | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 10 | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | 10 | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | 10 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 10 | Fall time | | 20 | | 20 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{PSEN}$ low.

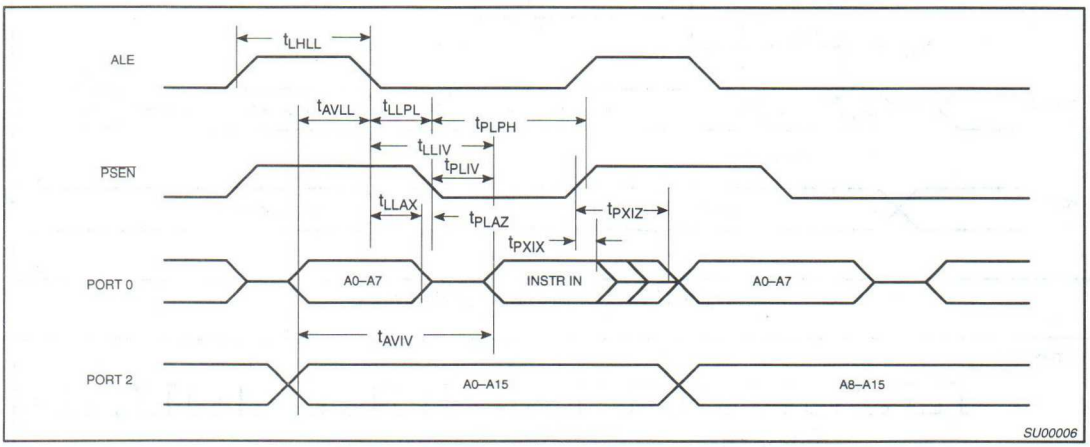



**Figure 2. External Program Memory Read Cycle**

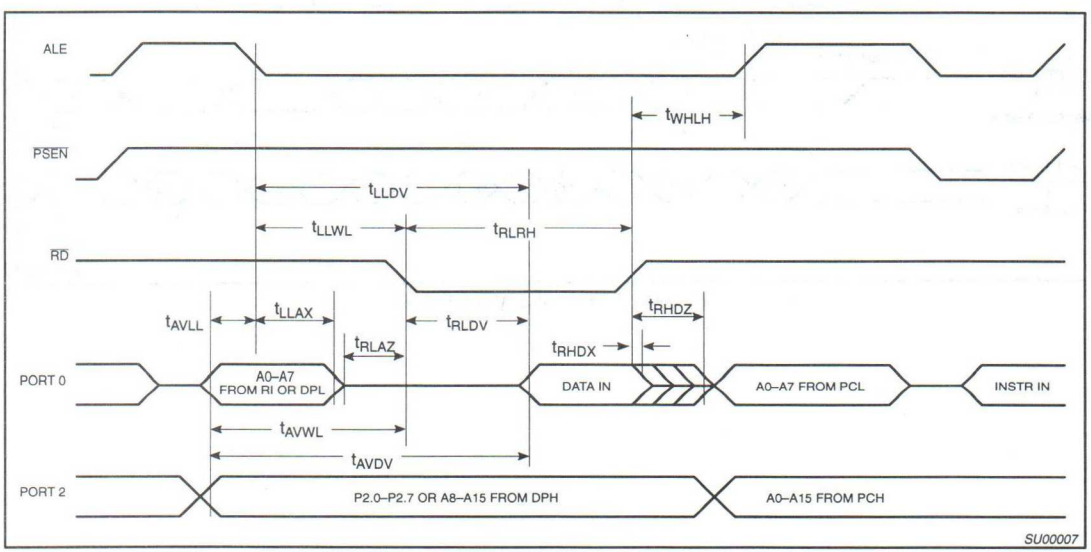



**Figure 3. External Data Memory Read Cycle**

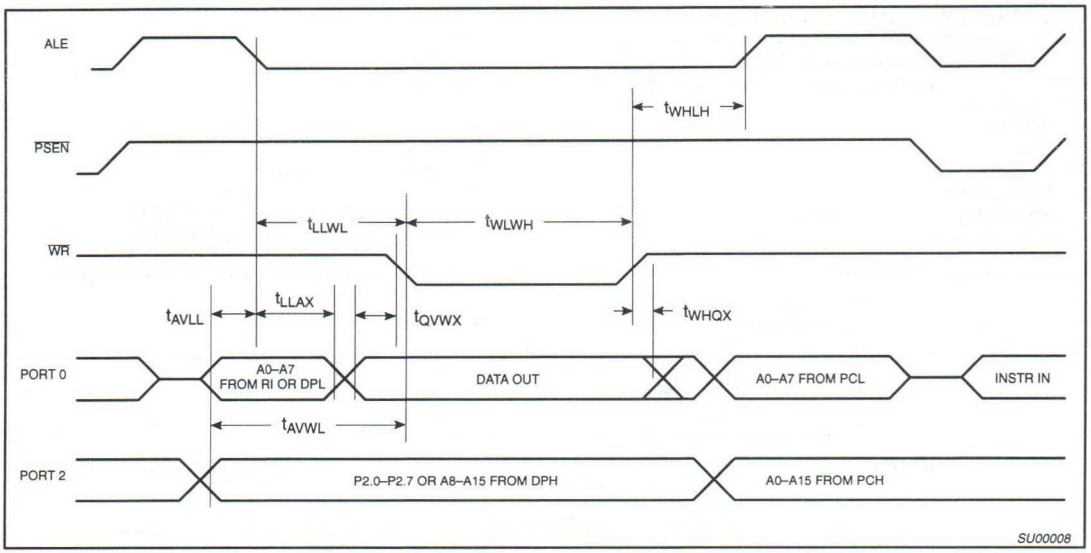



**Figure 4. External Data Memory Write Cycle**

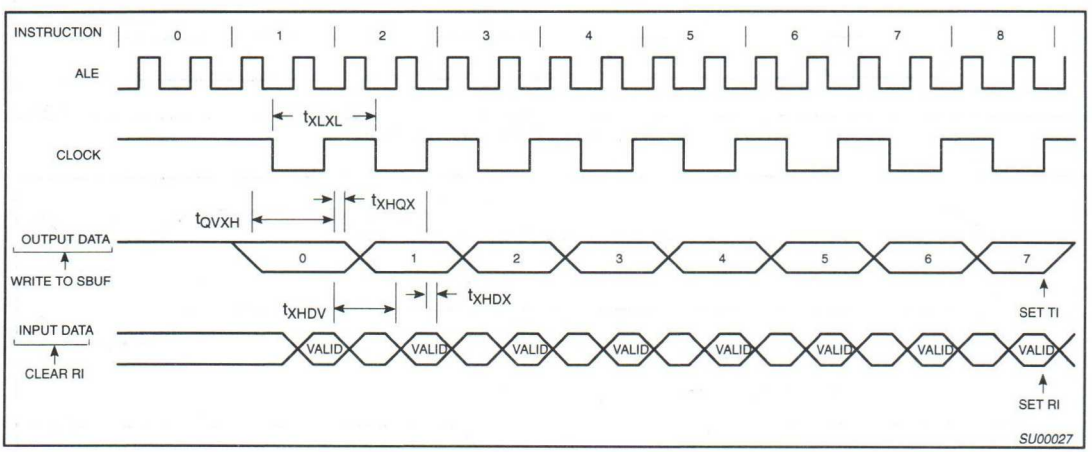



**Figure 5. Shift Register Mode Timing**

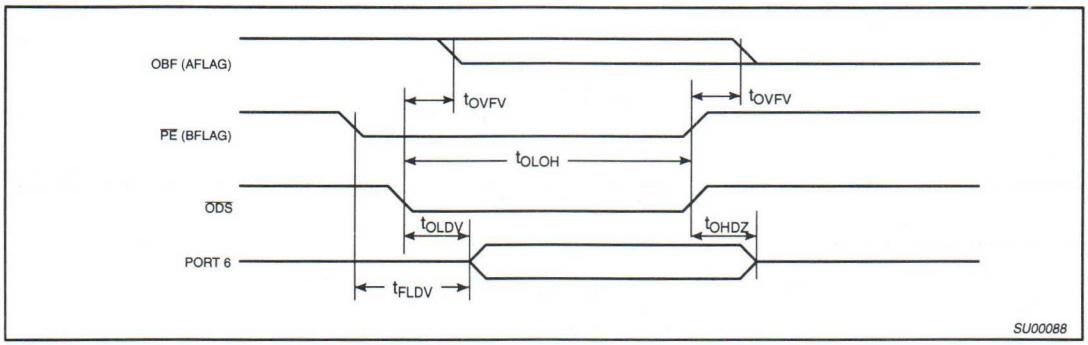



**Figure 6. Port 6 Output**

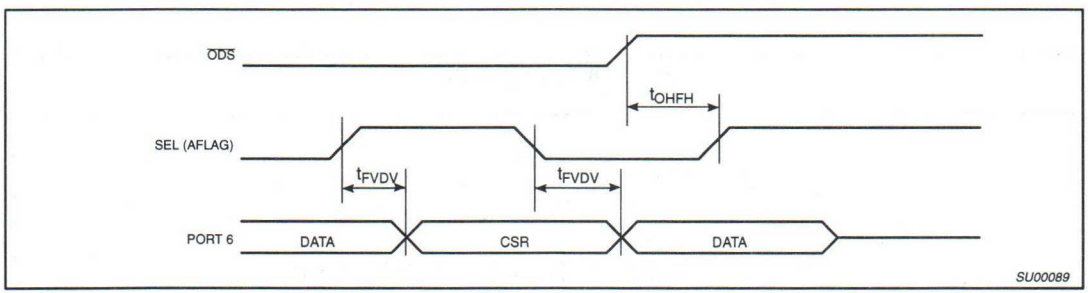



**Figure 7. Port 6 Select Mode**

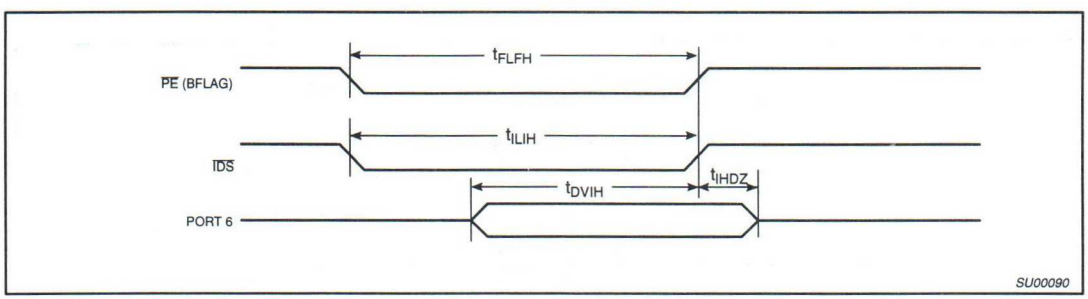



**Figure 8. Port 6 Input**

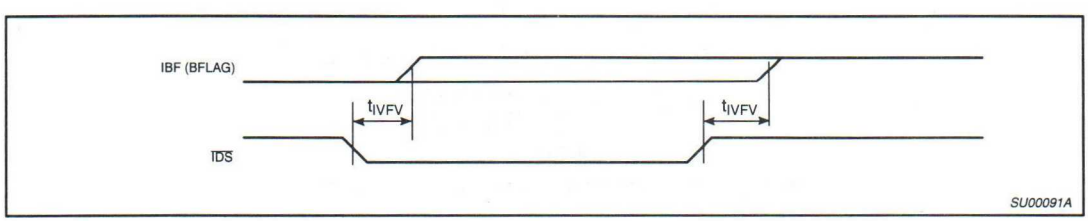



**Figure 9. IBF Flag Output**

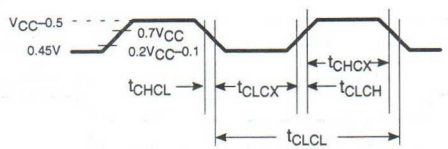
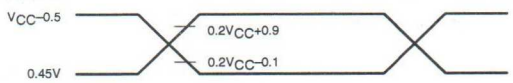


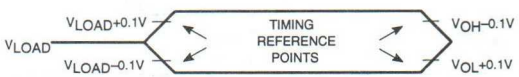

**Figure 10. External Clock Drive**




NOTE:
AC inputs during testing are driven at $V_{CC}$ –0.5 for a logic '1' and 0.45V for a logic '0'. Timing measurements are made at $V_{IH}$ min for a logic '1' and $V_{IL}$ max for a logic '0'.

SU00010

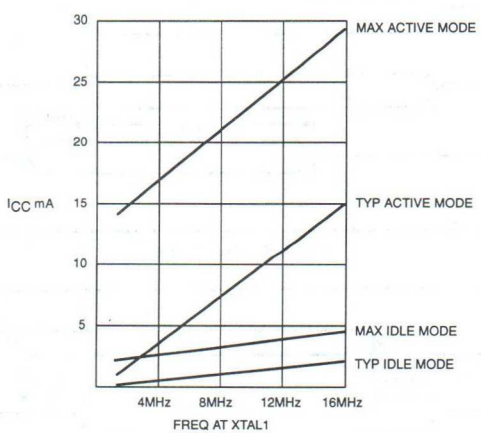
**Figure 11. AC Testing Input/Output**




NOTE:
For timing purposes, a port is no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded $V_{OH}/V_{OL}$ level occurs. $I_{OH}/I_{OL} \geq \pm 20$mA.

SU00011

**Figure 12. Float Waveform**




VALID ONLY WITHIN FREQUENCY SPECIFICATIONS OF THE DEVICE UNDER TEST.

SU00092

**Figure 13. $I_{CC}$ vs. FREQ**

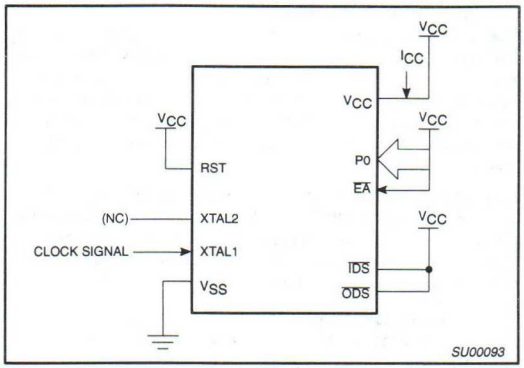



**Figure 14. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

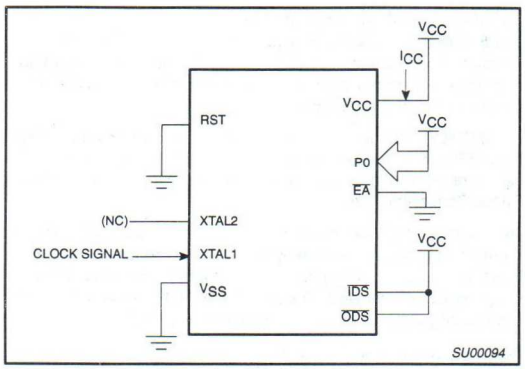



**Figure 15. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

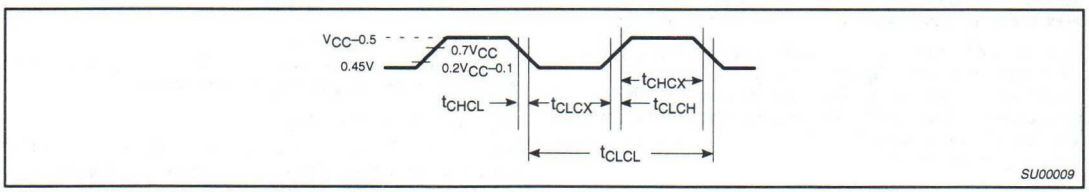



**Figure 16. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

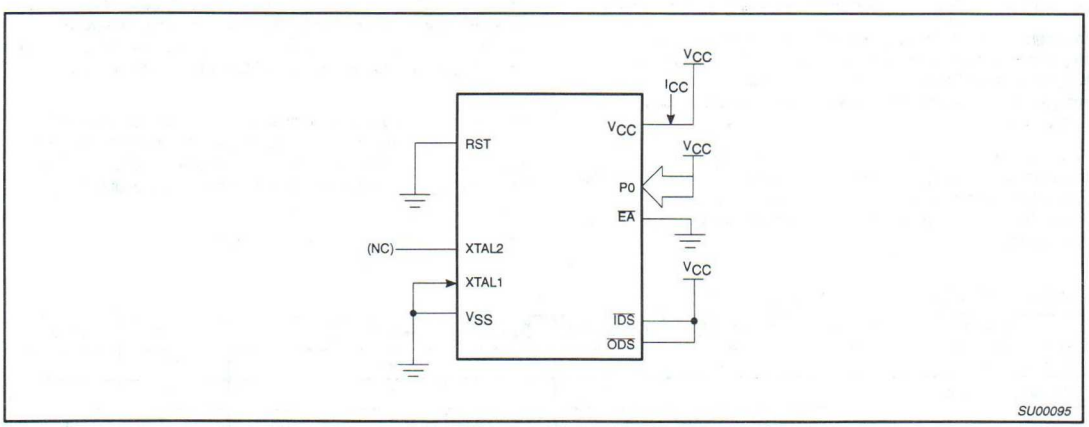



**Figure 17. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2V to 5.5V**

## EPROM CHARACTERISTICS

The 87C451 is programmed by using a modified Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C451 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C451 manufactured by Philips Semiconductors.

Table 4 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the lock bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 18 and 19. Figure 20 shows the circuit configuration for normal program memory verification.

### Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 18. Note that the 87C451 is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1 and 2, as shown in Figure 18. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 3 specified in Table 4 are held at the 'Program Code Data' levels indicated in Table 4. The ALE/$\overline{PROG}$ is pulsed low 25 times as shown in Figure 19.

To program the encryption table, repeat the 25 pulse programming sequence for addresses 0 through 1FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the lock bits, repeat the 25 pulse programming sequence using the 'Pgm Lock Bit' levels. After one lock bit is programmed, further programming of the code memory and encryption table is disabled. However, the other lock bit can still be programmed.

Note that the $\overline{EA}/V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

**Program Verification**

If lock bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 1 and 2 as shown in Figure 20. The other pins are held at the 'Verify Code Data' levels indicated in Table 4. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

**Reading the Signature Bytes**

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:
(030H) = 15H indicates manufactured by Philips
(031H) = 90H indicates 87C451

### Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 4, and which satisfies the timing specifications, is suitable.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-sec/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

**Table 4. EPROM Programming Modes**

| MODE | RST | $\overline{PSEN}$ | ALE/$\overline{PROG}$ | $\overline{EA}/V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm lock bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm lock bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V ±10% during programming and verification.
* ALE/$\overline{PROG}$ receives 25 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

™Trademark phrase of Intel Corporation.

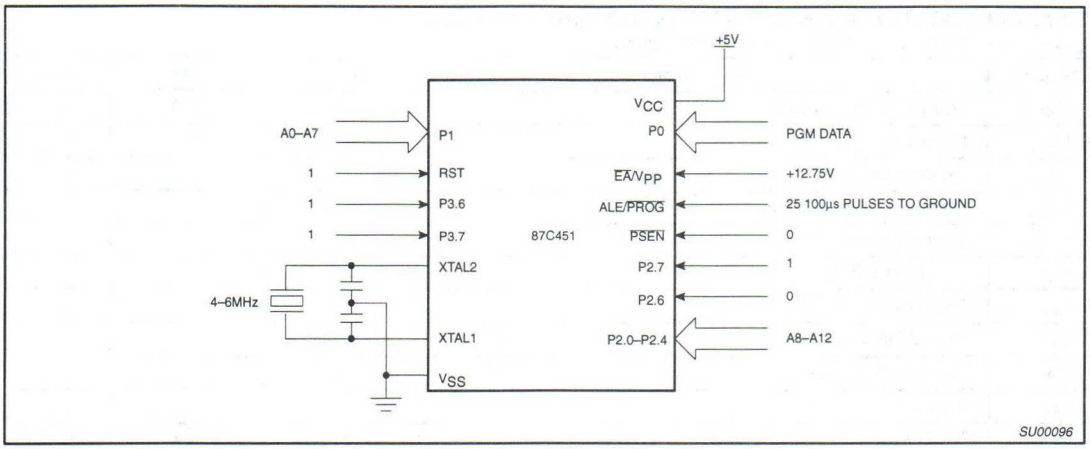



**Figure 18. Programming Configuration**

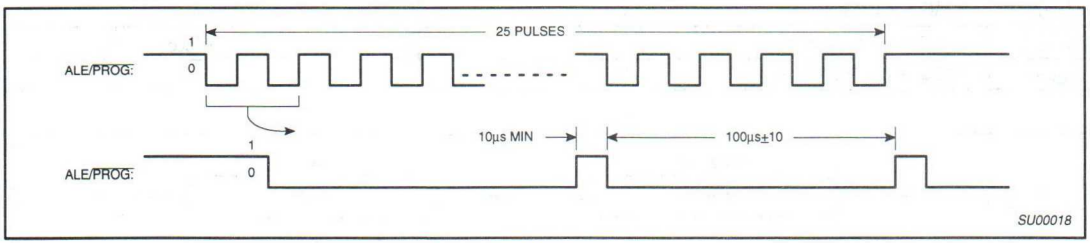



**Figure 19. PROG Waveform**

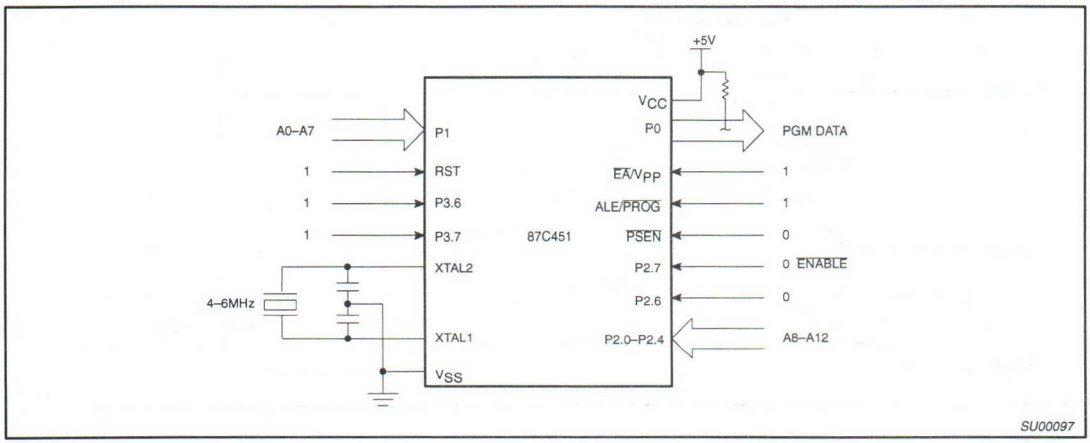
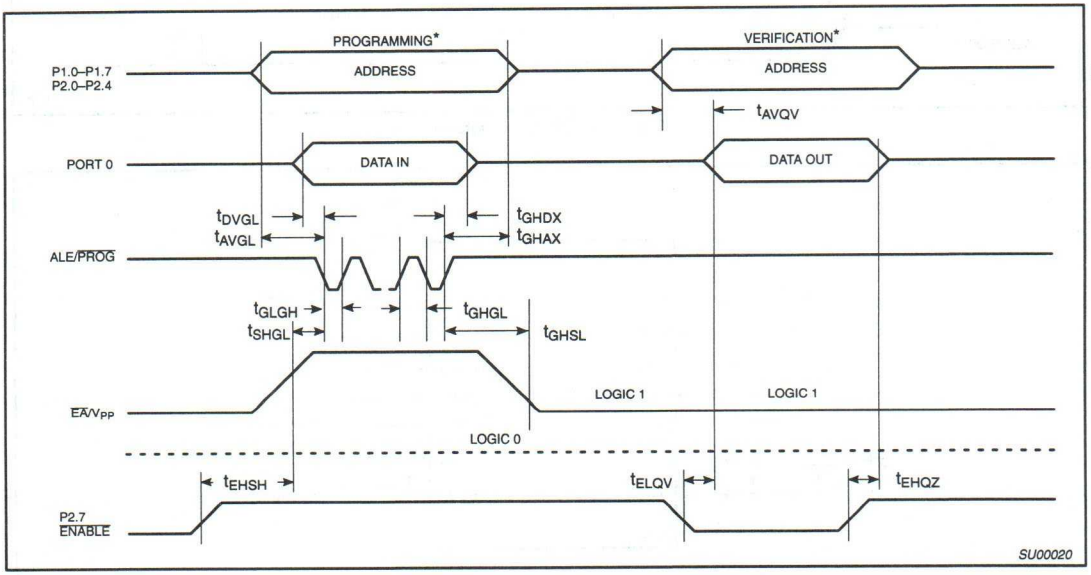



**Figure 20. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (See Figure 21)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




**NOTE:**

* FOR PROGRAMMING VERIFICATION SEE FIGURE 18.
FOR VERIFICATION CONDITIONS SEE FIGURE 20.

**Figure 21. EPROM Programming and Verification**

# CMOS single-chip 8-bit microcontrollers 80C453/83C453/87C453

## DESCRIPTION

The Philips 8XC453 is an I/O expanded single-chip microcontroller fabricated with Philips high-density CMOS technology. Philips epitaxial substrate minimizes latch-up sensitivity.

The 8XC453 is a functional extension of the 87C51 microcontroller with three additional I/O ports and four I/O control lines. The 8XC453 is available in 68-pin LCC packages. Four control lines associated with port 6 facilitate high-speed asynchronous I/O functions.

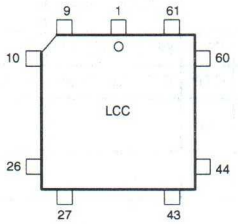
The 87C453 includes an 8k × 8 EPROM, a 256 × 8 RAM, 56 I/O lines, two 16-bit timer/counters, a seven source, two priority level, nested interrupt structure, a serial I/O port for either a full duplex UART, I/O expansion, or multi-processor communications, and on-chip oscillator and clock circuits.

The 87C453 has two software selectable modes of reduced activity for further power reduction; idle mode and power-down mode. Idle mode freezes the CPU while allowing the RAM, timers, serial port, and interrupt system to continue functioning. Power-down mode freezes the oscillator, causing all other chip functions to be inoperative while maintaining the RAM contents.

## FEATURES

- 80C51 based architecture
- Seven 8-bit I/O ports
- Port 6 features:
  - Eight data pins
  - Four control pins
  - Direct MPU bus interface
  - ISA Bus Interface
  - Parallel printer interface
  - IBF and $\overline{OBF}$ interrupts
  - A flag latch on host write
- On the microcontroller:
  - 8k × 8 EPROM
    Quick pulse programming algorithm
    Two-level program security system
  - 256 × 8 RAM
  - Two 16-bit counter/timers
  - Two external interrupts
- External memory addressing capability
  - 64k ROM and 64k RAM
- Low power consumption:
  - Normal operation: less than 24mA at 5V, 16MHz
  - Idle mode
  - Power-down mode
- Reduced EMI
- Full-duplex enhanced UART
  - Framing error detection
  - Automatic address recognition

## LCC PIN FUNCTIONS




| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | $\overline{EA}$/$V_{PP}$ | 24 | P4.2 | 47 | P5.3 |
| 2 | P2.0/A8 | 25 | P4.1 | 48 | P5.4 |
| 3 | P2.1/A9 | 26 | P4.0 | 49 | P5.5 |
| 4 | P2.2/A10 | 27 | P1.0 | 50 | P5.6 |
| 5 | P2.3/A11 | 28 | P1.1 | 51 | P5.7 |
| 6 | P2.4/A12 | 29 | P1.2 | 52 | XTAL2 |
| 7 | P2.5/A13 | 30 | P1.3 | 53 | XTAL1 |
| 8 | P2.6/A14 | 31 | P1.4 | 54 | $V_{SS}$ |
| 9 | P2.7/A15 | 32 | P1.5 | 55 | $\overline{ODS}$ |
| 10 | P0.7/AD7 | 33 | P1.6 | 56 | $\overline{IDS}$ |
| 11 | P0.6/AD6 | 34 | P1.7 | 57 | BFLAG |
| 12 | P0.5/AD5 | 35 | RST | 58 | AFLAG |
| 13 | P0.4/AD4 | 36 | P3.0/RxD | 59 | P6.0 |
| 14 | P0.3/AD3 | 37 | P3.1/TxD | 60 | P6.1 |
| 15 | P0.2/AD2 | 38 | P3.2/$\overline{INT0}$ | 61 | P6.2 |
| 16 | P0.1/AD1 | 39 | P3.3/$\overline{INT1}$ | 62 | P6.3 |
| 17 | P0.0/AD0 | 40 | P3.4/T0 | 63 | P6.4 |
| 18 | $V_{CC}$ | 41 | P3.5/T1 | 64 | P6.5 |
| 19 | P4.7 | 42 | P3.6/$\overline{WR}$ | 65 | P6.6 |
| 20 | P4.6 | 43 | P3.7/$\overline{RD}$ | 66 | P6.7 |
| 21 | P4.5 | 44 | P5.0 | 67 | $\overline{PSEN}$ |
| 22 | P4.4 | 45 | P5.1 | 68 | ALE/$\overline{PROG}$ |
| 23 | P4.3 | 46 | P5.2 | | |

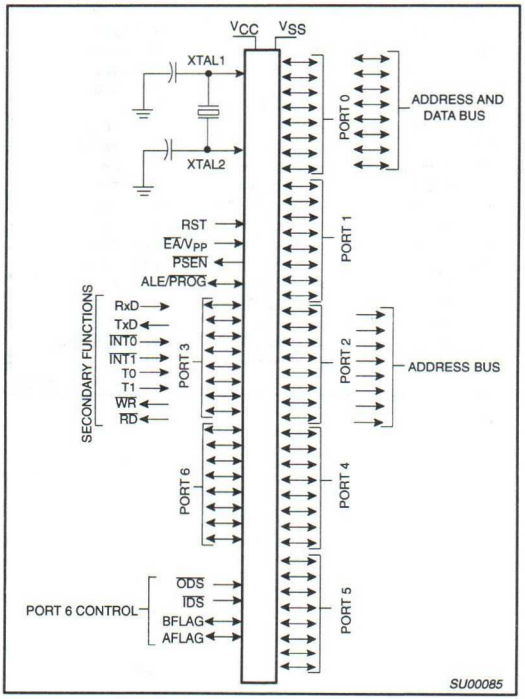
SU00157

## ORDERING INFORMATION

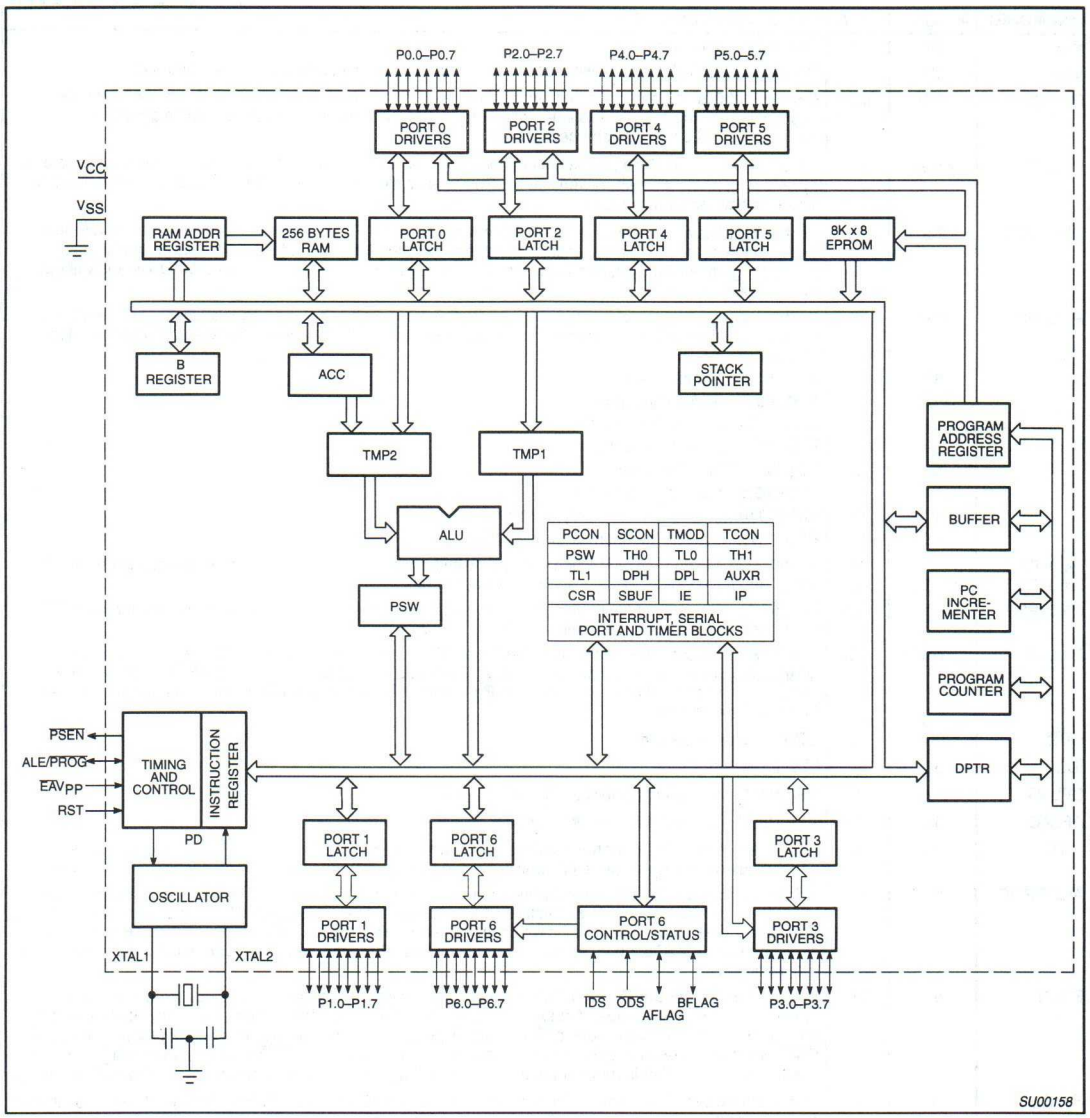
| EPROM[1] | | ROMLESS | ROM | TEMPERATURE °C AND PACKAGE | FREQ. (MHz) | PKG. DWG # |
|---|---|---|---|---|---|---|
| P87C453EBAA | OTP | P80C453EBAA | P83C453EBAA | 68–Pin Plastic Leaded Chip Carrier, 0 to +70 | 3.5 to 16 | SOT188-3 |
| P87C453EFAA | OTP | P80C453EFAA | P83C453EFAA | 68–Pin Plastic Leaded Chip Carrier, –40 to +85 | 3.5 to 16 | SOT188-3 |
| P87C453EBLKA | UV | | | 68-Pin Ceramic Leaded Chip Carrier with window, 0 to +70 | 3.5 to 16 | 1473A |
| P87C453EFLKA | UV | | | 68-Pin Ceramic Leaded Chip Carrier with window, –40 to +85 | 3.5 to 16 | 1473A |

**NOTE:**
1. OTP = One-Time Programmable EPROM.
   UV = Erasable EPROM.

## LOGIC SYMBOL



## BLOCK DIAGRAM



## PIN DESCRIPTION

| MNEMONIC | PIN NO. | TYPE | NAME AND FUNCTION |
|---|---|---|---|
| $V_{SS}$ | 54 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 18 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 17-10 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 is also the multiplexed data and low-order address bus during accesses to external memory. External pull-ups are required during program verification. Port 0 can sink/source eight LS TTL inputs. |
| P1.0–P1.7 | 27-34 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 receives the low-order address bytes during program memory verification. Port 1 can sink/source three LS TTL inputs, and drive CMOS inputs without external pull-ups. |
| P2.0–P2.7 | 2-9 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 emits the high-order address bytes during access to external memory and receives the high-order address bits and control signals during program verification. Port 2 can sink/source three LS TTL inputs, and drive CMOS inputs without external pull-ups. |
| P3.0–P3.7 | 36-43 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 can sink/source three LS TTL inputs, and drive CMOS inputs without external pull-ups. Port 3 also serves the special functions listed below: |
| | 36 | I | **RxD (P3.0):** Serial input port |
| | 37 | O | **TxD (P3.1):** Serial output port |
| | 38 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 39 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 40 | I | **T0 (P3.4):** Timer 0 external input |
| | 41 | I | **T1 (P3.5):** Timer 1 external input |
| | 42 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 43 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| P4.0–P4.3 | | I/O | **Port 4:** Port 4 is an 8-bit bidirectional I/O port with internal pull-ups. Port 4 can sink/source three LS TTL |
| P4.0–P4.7 | 26-19 | I/O | inputs and drive CMOS inputs without external pull-ups. |
| P5.0–P5.7 | 44-51 | I/O | **Port 5:** Port 5 is an 8-bit bidirectional I/O port with internal pull-ups. Port 5 can sink/source three LS TTL inputs and drive CMOS inputs without external pull-ups. |
| P6.0–P6.7 | 59-66 | I/O | **Port 6:** Port 6 is a specialized 8-bit bidirectional I/O port with internal pull-ups. This special port can sink/source three LS TTL inputs and drive CMOS inputs without external pull-ups. Port 6 can be used in a strobed or non-strobed mode of operation. Port 6 works in conjunction with four control pins that serve the functions listed below: |
| $\overline{ODS}$ | 55 | I | **$\overline{ODS}$:** Output data strobe |
| $\overline{IDS}$ | 56 | I | **$\overline{IDS}$:** Input data strobe |
| BFLAG | 57 | I/O | **BFLAG:** Bidirectional I/O pin with internal pull-ups |
| AFLAG | 58 | I/O | **AFLAG:** Bidirectional I/O pin with internal pull-ups |
| RST | 35 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal pull-down resistor permits a power-on reset using only an external capacitor connected to $V_{CC}$. |
| ALE/$\overline{PROG}$ | 68 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. ALE is activated at a constant rate of 1/6 the oscillator frequency except during an external data memory access, at which time one ALE is skipped. ALE can sink/source three LS TTL inputs and drive CMOS inputs without external pull-ups. This pin is also the program pulse during EPROM programming. |
| $\overline{PSEN}$ | 67 | O | **Program Store Enable:** The read strobe to external program memory. $\overline{PSEN}$ is activated twice each machine cycle during fetches from external program memory. However, when executing out of external program memory, two activations of $\overline{PSEN}$ are skipped during each access to external program memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. $\overline{PSEN}$ can sink/source eight LS TTL inputs and drive CMOS inputs without an external pull-up. This pin should be tied low during programming. |
| $\overline{EA}/V_{PP}$ | 1 | I | **Instruction Execution Control/Programming Supply Voltage:** When $\overline{EA}$ is held high, the CPU executes out of internal program memory, unless the program counter exceeds 1FFFH. When $\overline{EA}$ is held low, the CPU executes out of external program memory. $\overline{EA}$ must never be allowed to float. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. |
| XTAL1 | 53 | I | **Crystal 1:** Input to the inverting oscillator amplifier that forms the oscillator. This input receives the external oscillator when an external oscillator is used. |
| XTAL2 | 52 | O | **Crystal 2:** An output of the inverting amplifier that forms the oscillator. This pin should be floated when an external oscillator is used. |

**Table 1. 87C453 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT NAMES AND ADDRESSES MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| | | | EF | EE | ED | EC | EB | EA | E9 | E8 | |
| CSR*# | Port 6 command/status | E8H | MB1 | MB0 | MA1 | MA0 | OBFC | IDSM | OBF | IBF | FCH |
| DPTR | Data pointer (2 bytes) | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP* | Interrupt priority | B8H | – | POB | PIB | PS | PT1 | PX1 | PT0 | PX0 | x0000000B |
| AUXR# | Auxiliary register | 8EH | – | – | – | – | – | – | AF | AO | x0000000B |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE* | Interrupt enable | A8H | EA | IOB | IIB | ES | ET1 | EX1 | ET0 | EX0 | 00000000B |
| P0* | Port 0 | 80H | 87 | B6 | 85 | 84 | 83 | 82 | 81 | 80 | FFH |
| P1* | Port 1 | 90H | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | FFH |
| P2* | Port 2 | A0H | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | FFH |
| P3* | Port 3 | B0H | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | FFH |
| P4*# | Port 4 | C0H | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 | FFH |
| P5*# | Port 5 | C8H | CF | CE | CD | CC | CB | CA | C9 | C8 | FFH |
| P6*# | Port 6 | D8H | DF | DE | DD | DC | DB | DA | D9 | D8 | FFH |
| PCON | Power control | 87H | SMOD1 | SMOD0 | – | POF[1] | GF1 | GF0 | PD | IDL | 00xx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| SADDR# | Slave Address | A9H | | | | | | | | | 00H |
| SADEN# | Slave Address Mask | B9H | | | | | | | | | 00H |
| SBUF | Serial data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial port control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer/counter control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| TMOD | Timer/counter mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |
| TH0 | Timer 0 high byte | 8CH | | | | | | | | | 00H |
| TH1 | Timer 1 high byte | 8DH | | | | | | | | | 00H |
| TL0 | Timer 0 low byte | 8AH | | | | | | | | | 00H |
| TL1 | Timer 1 low byte | 8BH | | | | | | | | | 00H |

**NOTES:**
* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
1. REset value depends on reset source.

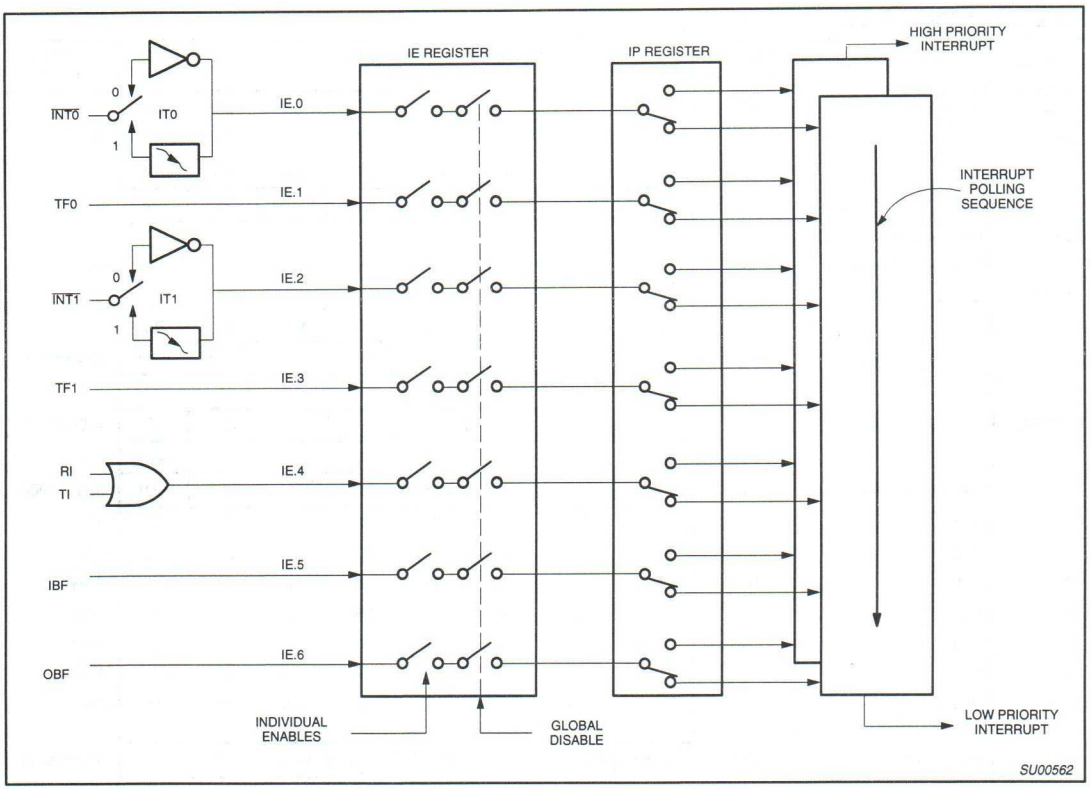



**Figure 1. 8XC453 Interrupt Control System**

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| EA | IOB | IIB | ES | ET1 | EX1 | ET0 | EX0 |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IE.7 | EA | Disables all interrupts. If EA=0, no interrupt will be acknowledged. If EA=1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit. |
| IE.6 | IOB | Enables or disables the Output Buffer Full (OBF) interrupt. If IOB=0, the interrupt is disabled, If IOB=1, an interrupt will occur if EA is set and data has been read from the output buffer register through Port 6 by the external host pulsing ODS low. |
| IE.5 | IIB | Enables or disables the Input Buffer Full (IBF) interrupt. If IIB=0, the interrupt is disabled. If IIB=1, an interrupt will occur if EA is set and data has been written into the Port 6 Input Data Buffer by the host strobing IDS low. |
| IE.4 | ES | Enables or disables the Serial Port Interrupt. If ES=0, the Serial Port Interrupt. If ES=0, the Serial Port interrupt is disabled. |
| IE.3 | ET1 | Enables or disables the Timer 1 Overflow interrupt. If ET1=0, the Timer 1 interrupt is disabled. |
| IE.2 | EX1 | Enables or disables External Interrupt 1. If EX1=0, External Interrupt 1 is disabled. |
| IE.1 | ET0 | Enables or disables the Timer 0 Overflow interrupt. If ET0=0, the Timer 0 interrupt is disabled. |
| IE.0 | EX0 | Enables or disables External Interrupt 0. If EX0=0, external Interrupt 0 is disabled. |

SU00563

**Figure 2. 8XC453 Interrupt Enable (IE) Register**

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| — | POB | PIB | PS | PT1 | PX1 | PT0 | PX0 |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IP.7 | — | Reserved. |
| IP.6 | POB | Defines the Output Buffer Full interrupt (IOB) priority level. POB=1 programs it to the higher priority level. |
| IP.5 | PIB | Defines the Input Buffer Full interrupt (IIB) priority level. PIB=1 programs it to the higher priority level. |
| IP.4 | PS | Defines the Serial Port interrupt priority level. PS=1 programs it to the higher priority level. |
| IP.3 | PT1 | Defines the Timer 1 interrupt priority level. PT1=1 programs it to the higher priority level. |
| IP.2 | PX1 | Defines the External Interrupt 1 priority level. PX1=1 programs it to the higher priority level. |
| IP.1 | PT0 | Enables or disables the Timer 0 interrupt priority level. PT0=1 programs it to the higher priority level. |
| IP.0 | PX0 | Defines the External Interrupt 0 priority level. PX0=1 programs it to the higher priority level. |

SU00564

**Figure 3. 8XC453 Interrupt Priority (IP) Register**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **PCON (87H)** | SMOD1 | SMOD2 | — | POF | GF1 | GF0 | PD | IDL |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| PCON.7 | SMOD1 | Double Baud rate bit. When set to a 1 and Timer 1 is used to generate baud rate, and the Serial Port is used in modes 1, 2, or 3. |
| PCON.6 | SMOD0 | If set to 1, SCON.7 will be the Framing Error bit (FE). If PCON.6 is cleared, SCON.7 will be SM0. |
| PCON.5 | — | Reserved. |
| PCON.4 | POF | Power Off Flag is set during power on of $V_{CC}$. If then cleared by software, it can be used to determine if a warm start has occurred. |
| PCON.3 | GF1 | General-purpose flag bit. |
| PCON.2 | GF0 | General-purpose flag bit. |
| PCON.1 | PD | Power-Down bit. Setting this bit activates power-down mode. It can only be set if input $\overline{EW}$ is high. |
| PCON.0 | IDL | Idle mode bit. Setting this bit activates the idle mode. |

**If logic 1s are written to PD and IDL at the same time, PD takes precedence.**

SU00565

**Figure 4. Power Control Register (PCON)**

**SCON** Address = 98H Reset Value = 0000 0000B

Bit Addressable

| | SM0/FE | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

(SMOD0 = 0/1)*

| Symbol | Function |
|---|---|
| **FE** | Framing Error bit. This bit is set by the receiver when an invalid stop bit is detected. The FE bit is not cleared by valid frames but should be cleared by software. The SMOD0 bit must be set to enable access to the FE bit. |
| **SM0** | Serial Port Mode Bit 0, (SMOD0 must = 0 to access bit SM0) |
| **SM1** | Serial Port Mode Bit 1 |

| SM0 | SM1 | Mode | Description | Baud Rate** |
|---|---|---|---|---|
| 0 | 0 | 0 | shift register | $f_{OSC}/12$ |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{OSC}/64$ or $f_{OSC}/32$ |
| 1 | 1 | 3 | 9-bit UART | variable |

| | |
|---|---|
| **SM2** | Enables the Automatic Address Recognition feature in Modes 2 or 3. If SM2 = 1 then RI will not be set unless the received 9th data bit (RB8) is 1, indicating an address, and the received byte is a Given or Broadcast Address. In Mode 1, if SM2 = 1 then RI will not be activated unless a valid stop bit was received, and the received byte is a Given or Broadcast Address. In Mode 0, SM2 should be 0. |
| **REN** | Enables serial reception. Set by software to enable reception. Clear by software to disable reception. |
| **TB8** | The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired. |
| **RB8** | In modes 2 and 3, the 9th data bit that was received. In Mode 1, if SM2 = 0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used. |
| **TI** | Transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software. |
| **RI** | Receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software. |

**NOTE:**
*SMOD0 is located at PCON6.
**$f_{OSC}$ = oscillator frequency

SU00043

**Figure 5. Serial Port Control Register (SCON)**

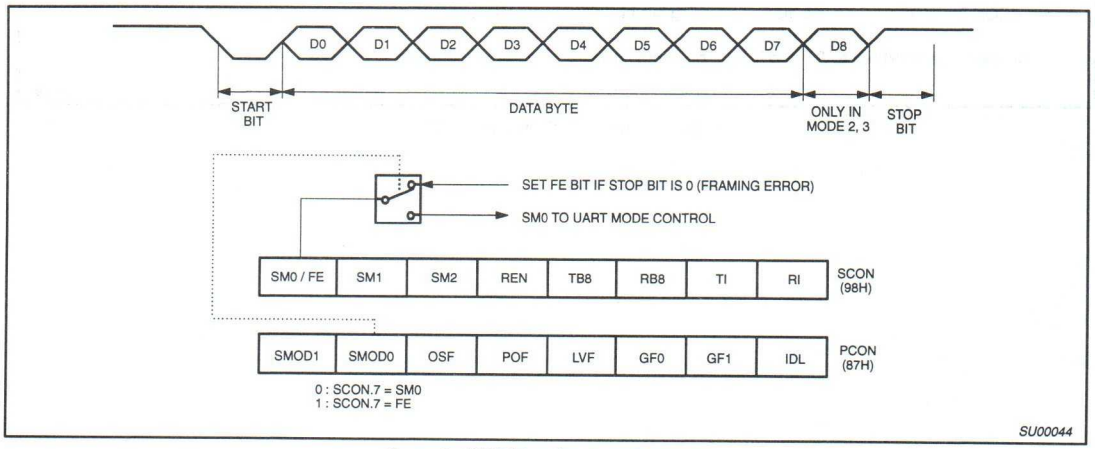



**Figure 6. UART Framing Error Detection**

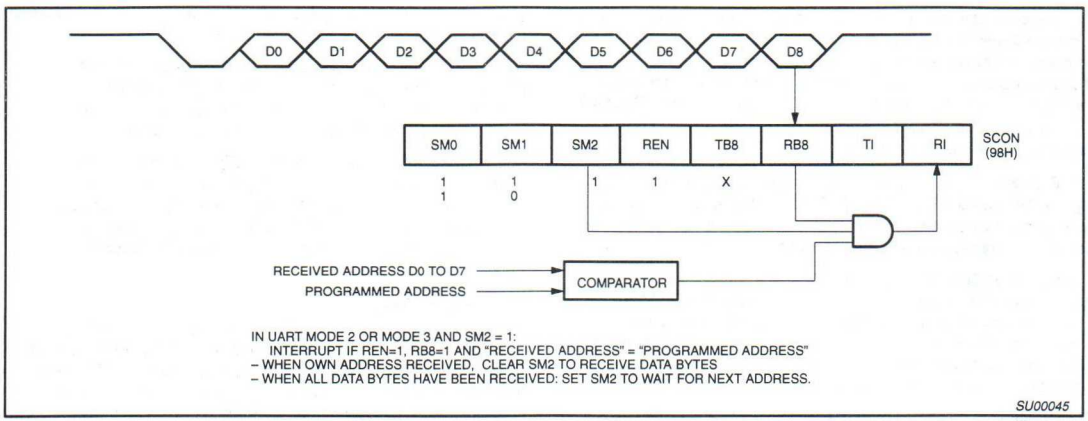



**Figure 7. UART Multiprocessor Communication, Automatic Address Recognition**

## SPECIAL FUNCTION REGISTER ADDRESSES

Special function register addresses for the device are identical to those of the 80C51, except for the additional registers listed in Table 2.

## Enhanced UART

The UART operates in all of the usual modes that are described in the first section of this book for the 80C51. In addition the UART can perform framing error detect by looking for missing stop bits, and automatic address recognition. The 87C453 UART also fully supports multiprocessor communication as does the standard 80C51 UART.

When used for framing error detect the UART looks for missing stop bits in the communication. A missing bit will set the FE bit in the SCON register. The FE bit shares the SCON.7 bit with SM0 and the function of SCON.7 is determined by PCON.6 (SMOD0) (see Figure 5). If SMOD0 is set then SCON.7 functions as FE. SCON.7 functions as SM0 when SMOD0 is cleared. When used as FE SCON.7 can only be cleared by software. Refer to Figure 6.

### Automatic Address Recognition

Automatic Address Recognition is a feature which allows the UART to recognize certain addresses in the serial bit stream by using hardware to make the comparisons. This feature saves a great deal of software overhead by eliminating the need for the software to examine every serial address which passes by the serial port. This feature is enabled by setting the SM2 bit in SCON. In the 9 bit UART modes, mode 2 and mode 3, the Receive Interrupt flag (RI) will be automatically set when the received byte contains either the "Given" address or the "Broadcast" address. The 9 bit mode requires that the 9th information bit is a 1 to indicate that the received information is an address and not data. Automatic address recognition is shown in Figure 7.

The 8 bit mode is called Mode 1. In this mode the RI flag will be set if SM2 is enabled and the information received has a valid stop bit following the 8 address bits and the information is either a Given or Broadcast address.

Mode 0 is the Shift Register mode and SM2 is ignored.

Using the Automatic Address Recognition feature allows a master to selectively communicate with one or more slaves by invoking the Given slave address or addresses. All of the slaves may be contacted by using the Broadcast address. Two special Function Registers are used to define the slave's address, SADDR, and the address mask, SADEN. SADEN is used to define which bits in the SADDR are to b used and which bits are "don't care". The SADEN mask can be logically ANDed with the SADDR to create the "Given" address which the master will use for addressing each of the slaves. Use of the Given address allows multiple slaves to be recognized while excluding others. The following examples will help to show the versatility of this scheme:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1101 |
| | Given | = | 1100 00X0 |
| Slave 1 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1110 |
| | Given | = | 1100 000X |

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires a 0 in bit 0 and it ignores bit 1. Slave 1 requires a 0 in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010 since slave 1 requires a 0 in bit 1. A unique address for slave 1 would be 1100 0001 since a 1 in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 1100 0000.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1001 |
| | Given | = | 1100 0XX0 |
| Slave 1 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1010 |
| | Given | = | 1110 0X0X |
| Slave 2 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1100 |
| | Given | = | 1110 00XX |

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 and 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary t make bit 2 = 1 to exclude slave 2.

The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are teated as don't-cares. In most cases, interpreting the don't-cares as ones, the broadcast address will be FF hexadecimal.

Upon reset SADDR (SFR address 0A9H) and SADEN (SFR address 0B9H) are leaded with 0s. This produces a given address of all "don't cares" as well as a Broadcast address of all "don't cares". this effectively disables the Automatic Addressing mode and allows the microcontroller to use standard 80C51 type UART drivers which do not make use of this feature.

The 87C453 UART has all of the capabilities of the standard 80C51 UART plus Framing Error Detection and Automatic Address Recognition. As in the 80C51, all four modes of operation are supported as well as the 9th bit in modes 2 and 3 that can be used to facilitate multiprocessor communication.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

### Reset

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up.

### Idle Mode

In the idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

### Power-Down Mode

To save even more power, a Power Down mode can be invoked by software. In this mode, the oscillator is stopped and the instruction that invoked Power Down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the Power Down mode is terminated.

On the 87C453 either a hardware reset or external interrupt can cause an exit from Power Down. Reset redefines all the SFRs but does not change the on-chip RAM. An external interrupt allows both the SFRs and the on-chip RAM to retain their values.

To properly terminate Power Down the reset or external interrupt should not be executed before $V_{CC}$ is restored to its normal operating level and must be held active long enough for the oscillator to restart and stabilize (normally less than 10ms).

With an external interrupt, INT0 and INT1 must be enabled and configured as level-sensitive. Holding the pin low restarts the oscillator but bringing the pin back high completes the exit. Once the interrupt is serviced, the next instruction to be executed after RETI will be the one following the instruction that put the device into Power Down.

### Power Off Flag

The Power Off Flag (POF) in PCON is set by on-chip circuitry when the $V_{CC}$ level on the 87C453 rises from 0 to 5V. The POF bit can be set or cleared by software allowing a user to determine if the reset is the result of a power-on or a warm start after powerdown. The $V_{CC}$ level must remain above 3V for the POF to remain unaffected by the $V_{CC}$ level.

### Design Consideration

- When the idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal rest algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

### ONCE™ Mode

The ONCE ("On-Circuit Emulation") Mode facilitates testing and debugging of systems using the 87C453 without having to remove the IC from the circuit. The ONCE Mode is invoked by:

1. Pull ALE low while the device is in reset and $\overline{PSEN}$ is high;
2. Hold ALE low as RST is deactivated.

While the device is in ONCE Mode, the Port 0 pins go into a float state, and the other port pins and ALE and $\overline{PSEN}$ are weakly pulled high. The oscillator circuit remains active. While the 87C453 is in this mode, an emulator or test CPU can be used to drive the circuit. Normal operation is restored when a normal reset is applied.

## PORTS 4 AND 5

Ports 4 and 5 are bidirectional I/O ports with internal pull-ups. Port 4 is an 8-bit port. Port 4 and port 5 pins with ones written to them, are pulled high by the internal pull-ups, and in that state can be used as inputs. Ports 4 and 5 are addressed at the special function register addresses shown in Table 2.

## PORT 6

Port 6 is a special 8-bit bidirectional I/O port with internal pull-ups (see Figure 8). This port can be used as a standard I/O port, or in strobed modes of operation in conjunction with four special control lines: $\overline{ODS}$, $\overline{IDS}$, AFLAG, and BFLAG. Port 6 operating modes are controlled by the port 6 control status register (CSR). Port 6 and the CSR are addressed at the special function register addresses shown in Table 2. The following four control pins are used in conjunction with port 6:

$\overline{ODS}$ – Output data strobe for port 6. $\overline{ODS}$ can be programmed to control the port 6 output drivers and the output buffer full flag (OBF), or to clear only the OBF flag bit in the CSR (output-always mode).

$\overline{ODS}$ is active low for output driver control. The OBF flag can be programmed to be cleared on the negative or positive edge of $\overline{ODS}$. Can produce an IOB interrupt (see Figure 2).

**$\overline{IDS}$** – Input data strobe for port 6. $\overline{IDS}$ is used to control the port 6 input latch and input buffer full flag (IBF) bit in the CSR. The input data latch can be programmed to be transparent when $\overline{IDS}$ is low and latched on the positive transition of $\overline{IDS}$, or to latch only on the positive transition of $\overline{IDS}$. Correspondingly, the IBF flag is set on the negative or positive transition of $\overline{IDS}$. Can produce an IIB interrupt (see Figure 2).

**AFLAG** – AFLAG is a bidirectional I/O pin which can be programmed to be an output set high or low under program control, or to output the state of the output buffer full flag. AFLAG can also be programmed to be an input which selects whether the contents of the output buffer, or the contents of the port 6 control status register will output on port 6. This feature grants complete port 6 status to external devices.

**BFLAG** – BFLAG is a bidirectional I/O pin which can be programmed to be an output, set high or low under program control, or to output the state of the input buffer full flag. BFLAG can also be programmed to input an enable signal for port 6. When BFLAG is used as an enable input, port 6 output drivers are in the high-impedance state, and the input latch does not respond to the $\overline{IDS}$ strobe when BFLAG is high. Both features are enabled when BFLAG is low. This feature facilitates the use of the 87C453 in bused multiprocessor systems.

## CONTROL STATUS REGISTER

The control status register (CSR) establishes the mode of operation for port 6 and indicates the current status of port 6 I/O registers. All control status register bits can be read and written by the CPU, except bits 0 and 1, which are read only. Reset writes ones to bits 2 through 7, and writes zeros to bits 0 and 1 (see Table 3).

**CSR.0 Input Buffer Full Flag (IBF) (Read Only)** – The IBF bit is set to a logic 1 when port 6 data is loaded into the input buffer under control of $\overline{IDS}$. This can occur on the negative or positive edge of $\overline{IDS}$, as determined by CSR.2. When IBF is set, the Interrupt Enable Register bit IIB (IE.5) is set. The Interrupt Service Routine vector address for this interrupt is 002BH. IBF is cleared when the CPU reads the input buffer register.

**CSR.1 Output Buffer Full Flag (OBF) (Read Only)** – The OBF flag is set to a logic 1 when the CPU writes to the port 6 output data buffer. OBF is cleared by the positive or negative edge of $\overline{ODS}$, as determined by CSR.3. When OBF is cleared, the Interrupt Enable Register bit IOB (IE.6) is set. The Interrupt Service Routine vector address for this interrupt is 0033H.

**CSR.2 IDS Mode Select (IDSM)** – When CSR.2 = 0, a low-to-high transition on the $\overline{IDS}$ pin sets the IBF flag. The Port 6 input buffer is loaded on the $\overline{IDS}$ positive edge. When CSR.2 = 1, a high-to-low transition on the $\overline{IDS}$ pin sets the IBF flag. Port 6 input buffer is transparent when $\overline{IDS}$ is low, and latched when $\overline{IDS}$ is high.

**CSR.3 Output Buffer Full Flag Clear Mode (OBFC)** – When CSR.3 = 1, the positive edge of the $\overline{ODS}$ input clears the OBF flag. When CSR.3 = 0, the negative edge of the $\overline{ODS}$ input clears the OBF flag.

**CSR.4, CSR.5 AFLAG Mode Select (MA0, MA1)** – Bits 4 and 5 select the mode of operation for the AFLAG pin as follows:

| MA1 | MA0 | AFLAG Function |
|---|---|---|
| 0 | 0 | Logic 0 output |
| 0 | 1 | Logic 1 output |
| 1 | 0 | OBF flag output (CSR.1) |
| 1 | 1 | Select (SEL) input mode |

The select (SEL) input mode is used to determine whether the port 6 data register or the control status register is output on port 6. When the select feature is enabled, the AFLAG input controls the source of port 6 output data. A logic 0 on AFLAG input selects the port 6 data register, and a logic 1 on AFLAG input selects the control status register.

The value of the AFLAG input is latched into the Auxiliary Register (AUXR) bit 1 (AUXR.1). Checking this bit (AF) will allow the 87C453's program to determine if Port 6 was loaded with data or a UPI command.

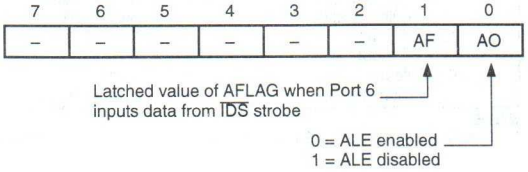
**CSR.6, CSR.7 BFLAG Mode Select (MB0, MB1)** – Bits 6 and 7 select the mode operation as follows:

| MB1 | MB0 | BFLAG Function |
|---|---|---|
| 0 | 0 | Logic 0 output |
| 0 | 1 | Logic 1 output |
| 1 | 0 | IBF flag output (CSR.0) |
| 1 | 1 | Port enable ($\overline{PE}$) |

In the port enable mode, $\overline{IDS}$ and $\overline{ODS}$ inputs are disabled when BFLAG input is high. When the BFLAG input is low, the port is enabled for I/O.

**Reduced EMI Mode** – The on–chip clock distribution drivers have been identified as the cause of most of the EMI emissions from the 80C51 family. By tailoring the clock drivers properly, a compromise between maximum operating speed and minimal EMI emissions can be achieved. Typically, an order in magnitude of reduction is possible over previous designs. This feature has been implemented on this chip along with the additional capability of turning off the ALE output. Setting the AO bit (AUXR.0) in the AUXR special function register will disable the ALE output. Reset forces a 0 into AUXR.0 to enable normal 80C51 type operation.

**Auxiliary Register (AUXR)**



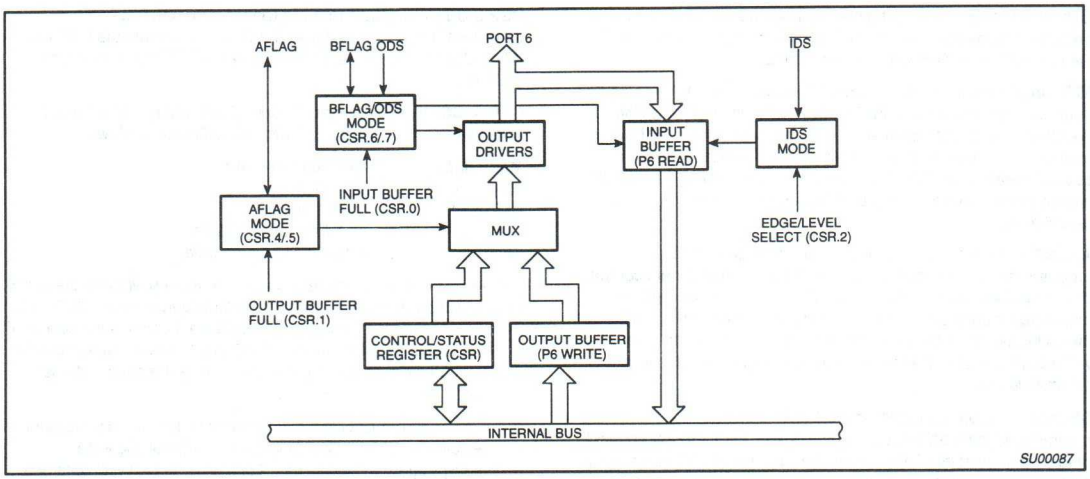



**Figure 8. Port 6 Block Diagram**

## Table 2. Special Function Register Addresses

| REGISTER ADDRESS | | | BIT ADDRESS | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | **Symbol** | **Address** | **MSB** | | | | | | | **LSB** |
| Port 4 | P4 | C0 | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 |
| Port 5 | P5 | C8 | CF | CE | CD | CC | CB | CA | C9 | C8 |
| Port 6 data | P6 | D8 | DF | DE | DD | DC | DB | DA | D9 | D8 |
| Port 6 control status | CSR | E8 | EF | EE | ED | EC | EB | EA | E9 | E8 |
| Slave address | SADDR | A9 | | | | | | | | |
| Slave address mask | SADEN | B9 | | | | | | | | |
| Auxiliary Register | AUXR | 8E | | | | | | | | |

## Table 3. Control Status Register (CSR)

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| **MB1** | **MB0** | **MA1** | **MA0** | **OBFC** | **IDSM** | **OBF** | **IBF** |
| BFLAG Mode Select | | AFLAG Mode Select | | Output Buffer Flag Clear Mode | Input Data Strobe Mode | Output Buffer Flag Full | Input Buffer Flag Full |
| 0/0 = Logic 0 output*<br>0/1 = Logic 1 output*<br>1/0 = IBF output<br>1/1 = PE input<br>(0 = Select)<br>(1 = Disable I/O) | | 0/0 = Logic 0 output*<br>0/1 = Logic 1 output*<br>1/0 = OBF output<br>1/1 = SEL input<br>(0 = Select)<br>(1 = Control/status) | | 0 = Negative edge of ODS<br>1 = Positive edge o ODS | 0 = Positive edge of IDS<br>1 = Low level of IDS | 0 = Output data buffer empty<br>1 = Output data buffer full | 0 = Input data buffer empty<br>1 = Input data buffer full |

**NOTE:**

* Output-always mode: MB1 = 0, MA1 = 1, and MA0 = 0. In this mode, port 6 is always enabled for output. ODS only clears the OBF flag.

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | 0 to +70<br>–40 to +85 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.5 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. Voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $V_{IL}$ | Input low voltage; ports 0, 1, 2, 3, 4, 5, 6, $\overline{IDS}$, $\overline{ODS}$, AFLAG, BFLAG; except $\overline{EA}$ | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | | 0 | | $0.2V_{CC}$–0.3 | V |
| $V_{IH}$ | Input high voltage; except XTAL1, RST | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage; XTAL1, RST | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage; ports 1, 2, 3, 4, 5, 6, AFLAG, BFLAG | $I_{OL}$ = 1.6mA[2] | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage; port 0, ALE, $\overline{PSEN}$ | $I_{OL}$ = 3.2mA[2] | | | 0.45 | V |
| $V_{OH}$ | Output high voltage; ports 1, 2, 3, 4, 5, 6, AFLAG, BFLAG | $I_{OH}$ = –60µA,<br>$I_{OH}$ = –25µA<br>$I_{OH}$ = –10µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$ | | | V<br>V<br>V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode, ALE, $\overline{PSEN}$)[3] | $I_{OH}$ = –800µA,<br>$I_{OH}$ = –300µA<br>$I_{OH}$ = –80µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$ | | | V<br>V<br>V |
| $I_{IL}$ | Logical 0 input current,; ports 1, 2, 3, 4, 5, 6 | $V_{IN}$ = 0.45V | | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current; ports 1, 2, 3, 4, 5, 6 | See note 4 | | | –650 | µA |
| $I_{LI}$ | Input leakage current; port 0 | $V_{IN}$ = $V_{IL}$ or $V_{IH}$ | | | ±10 | µA |
| $I_{CC}$ | Power supply current:<br>Active mode @ 16MHz[5]<br>Idle mode @ 16MHz[5]<br>Power down mode | See note 6 | | 11.5<br>1.3<br>3 | 25<br>4<br>50 | mA<br>mA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | 50 | | 300 | kΩ |
| $C_{IO}$ | Pin capacitance[7] – PLCC package | | | | 10 | pF |

**NOTES:**
1. Typical ratings are based on a limited number of samples from early manufacturing lots, and not guaranteed. Values are room temp., 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and the other ports. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input..
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
4. Pins of ports 1, 2, 3, 4, 5 and 6 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
5. $I_{CC}$MAX at other frequencies is given by:
   Active mode: $I_{CC}$MAX = 0.94 X FREQ + 13.71
   Idle mode: $I_{CC}$MAX = 0.14 X FREQ +2.31
   where FREQ is the external oscillator frequency in MHz. $I_{CC}$MAX is given in mA. See Figure 20.
6. See Figures 21 through 24 for $I_{CC}$ test conditions.
7. $C_{IO}$ applies to ports 1 through 6, $\overline{IDS}$, $\overline{ODS}$, AFLAG, BFLAG, XTAL1, XTAL2.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | | Oscillator frequency | | | 3.5 | 16 | MHz |
| $t_{LHLL}$ | 9 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 9 | Address valid to ALE low | 22 | | $t_{CLCL}-40$ | | ns |
| $t_{LLAX}$ | 9 | Address hold after ALE low | 32 | | $t_{CLCL}-30$ | | ns |
| $t_{LLIV}$ | 9 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 9 | ALE low to $\overline{PSEN}$ low | 32 | | $t_{CLCL}-30$ | | ns |
| $t_{PLPH}$ | 9 | $\overline{PSEN}$ pulse width | 142 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 9 | $\overline{PSEN}$ low to valid instruction in | | 82 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 9 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 9 | Input instruction float after $\overline{PSEN}$ | | 37 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 9 | Address to valid instruction in | | 207 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 9 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 10, 11 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 10, 11 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 10, 11 | $\overline{RD}$ low to valid data in | | 147 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 10, 11 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 10, 11 | Data float after $\overline{RD}$ | | 65 | | $2t_{CLCL}-60$ | ns |
| $t_{LLDV}$ | 10, 11 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 10, 11 | Address to valid data in | | 397 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 10, 11 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 137 | 239 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 10, 11 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 122 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 10, 11 | Data valid to $\overline{WR}$ transition | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{WHQX}$ | 10, 11 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{RLAZ}$ | 10, 11 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 10, 11 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 12 | Serial port clock cycle time | 750 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 12 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 12 | Output data hold after clock rising edge | 8 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 12 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 12 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}-133$ | ns |
| **Port 6 input (input rise and fall times = 5ns)** | | | | | | | |
| $t_{FLFH}$ | 15 | $\overline{PE}$ width | 209 | | $3t_{CLCL}+20$ | | ns |
| $t_{ILIH}$ | 15 | $\overline{IDS}$ width | 209 | | $3t_{CLCL}+20$ | | ns |
| $t_{DVIH}$ | 15 | Data setup to $\overline{IDS}$ high or $\overline{PE}$ high | 0 | | 0 | | ns |
| $t_{IHDZ}$ | 15 | Data hold after $\overline{IDS}$ high or $\overline{PE}$ high | 30 | | 30 | | ns |
| $t_{IVFV}$ | 16 | $\overline{IDS}$ to BFLAG (IBF) delay | | 130 | | 130 | ns |

## AC ELECTRICAL CHARACTERISTICS (Continued)

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| **Port 6 output** | | | | | | | |
| $t_{OLOH}$ | 13 | $\overline{ODS}$ width | 209 | | $3t_{CLCL}+20$ | | ns |
| $t_{FVDV}$ | 14 | SEL to data out delay | | 85 | | 85 | ns |
| $t_{OLDV}$ | 13 | $\overline{ODS}$ to data out delay | | 80 | | 80 | ns |
| $t_{OHDZ}$ | 13 | $\overline{ODS}$ to data float delay | | 35 | | 35 | ns |
| $t_{OVFV}$ | 13 | $\overline{ODS}$ to AFLAG (OBF) delay | | 100 | | 100 | ns |
| $t_{FLDV}$ | 13 | $\overline{PE}$ to data out delay | | 120 | | 120 | ns |
| $t_{OHFH}$ | 14 | $\overline{ODS}$ to AFLAG (SEL) delay | 100 | | 100 | | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 17 | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | 17 | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | 17 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 17 | Fall time | | 20 | | 20 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{\text{PSEN}}$
Q – Output data
R – $\overline{\text{RD}}$ signal
t – Time
V – Valid
W – $\overline{\text{WR}}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{\text{PSEN}}$ low.

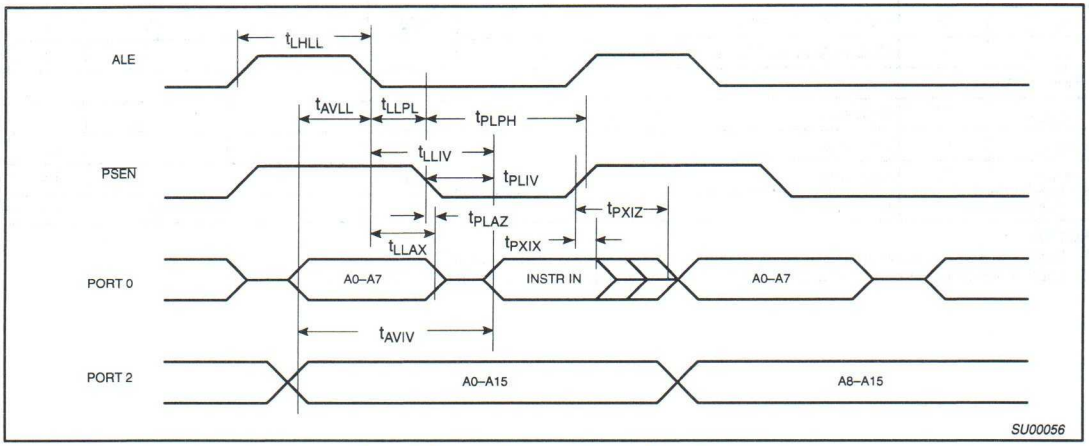



**Figure 9. External Program Memory Read Cycle**

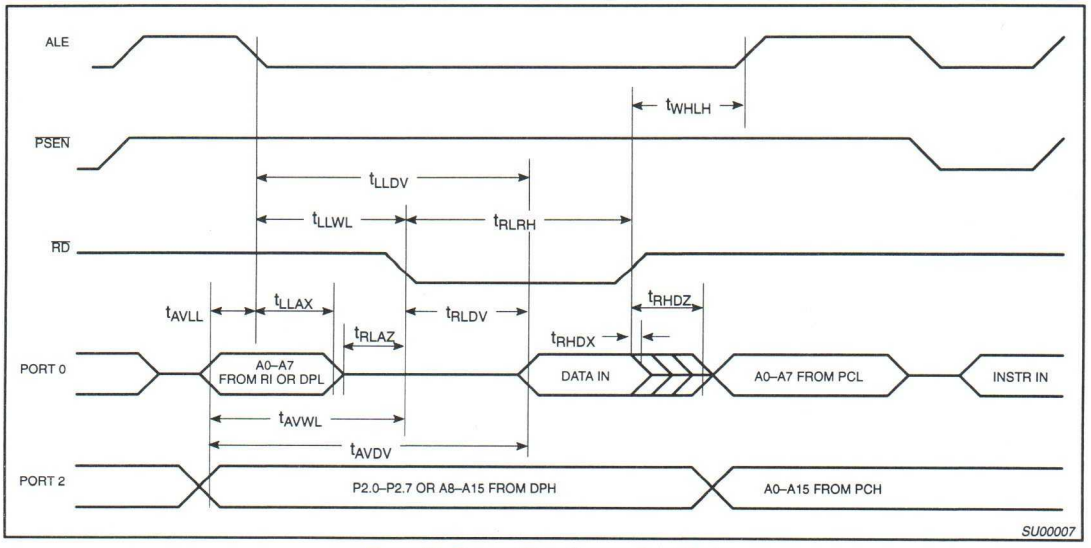



**Figure 10. External Data Memory Read Cycle**

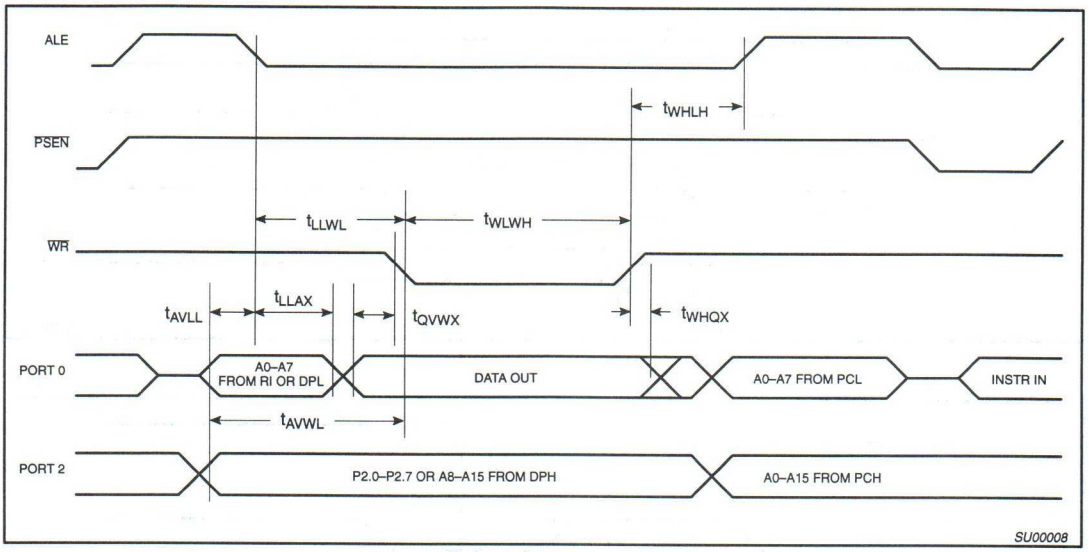



**Figure 11. External Data Memory Write Cycle**

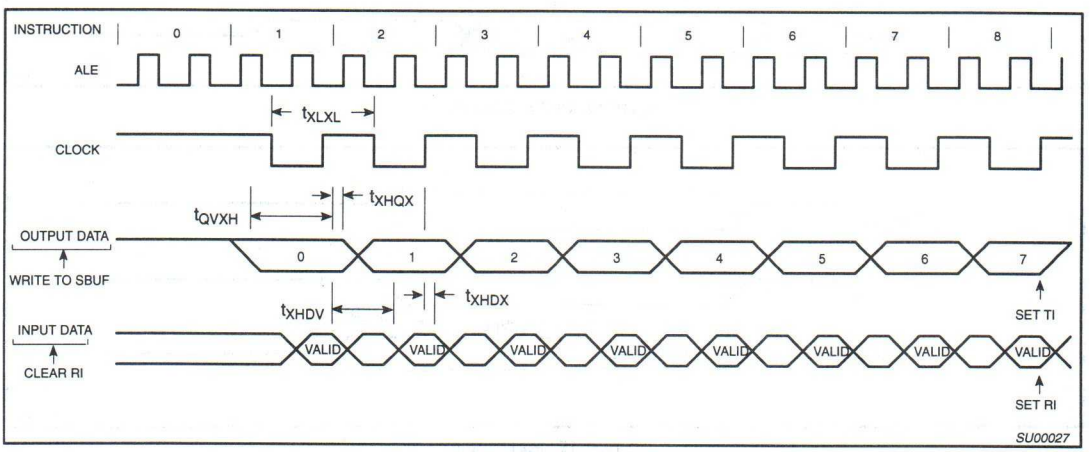



**Figure 12. Shift Register Mode Timing**

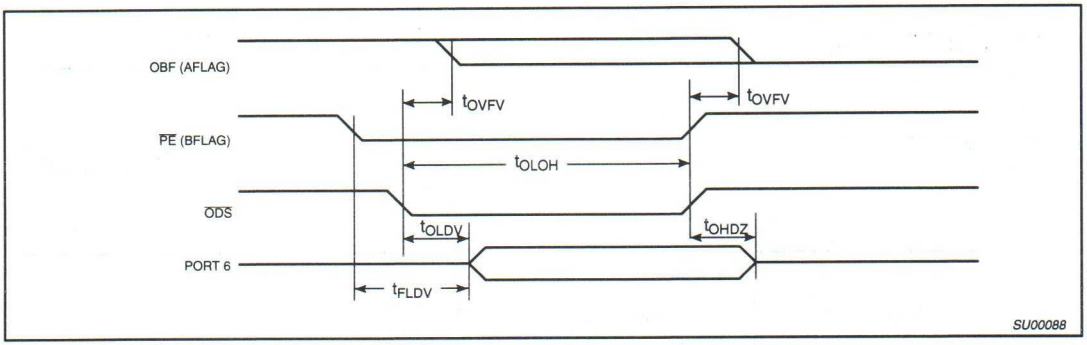



**Figure 13. Port 6 Output**

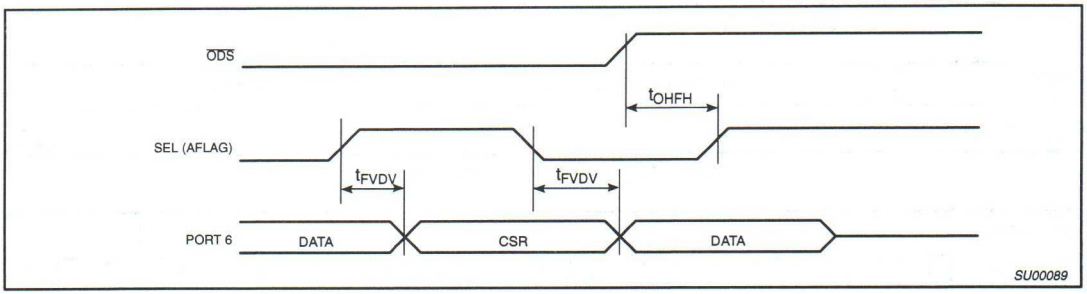



**Figure 14. Port 6 Select Mode**

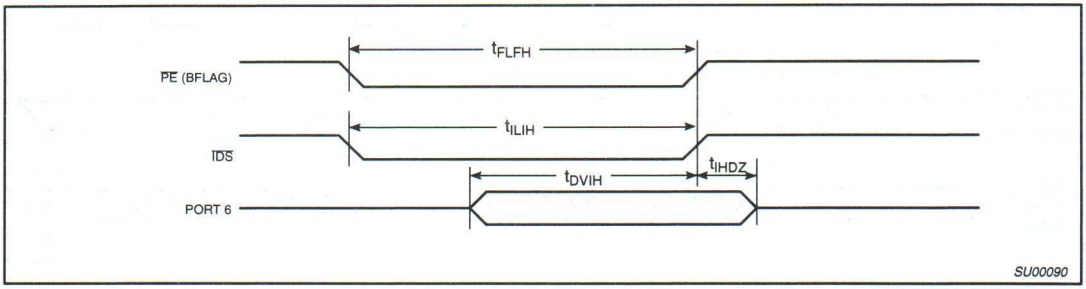



**Figure 15. Port 6 Input**

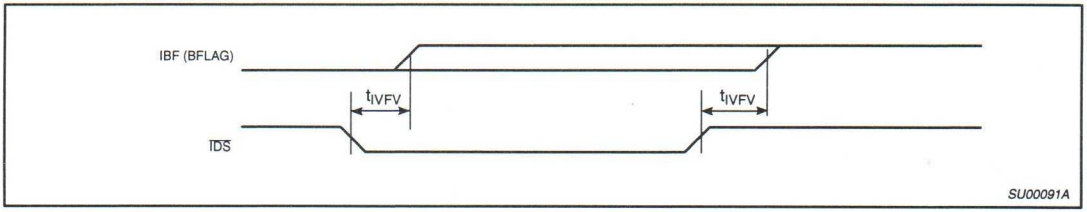



**Figure 16. IBF Flag Output**

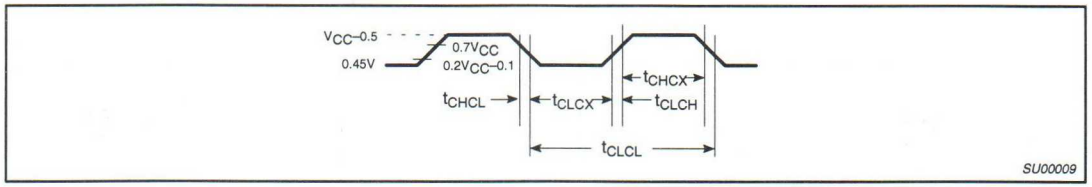



**Figure 17. External Clock Drive**

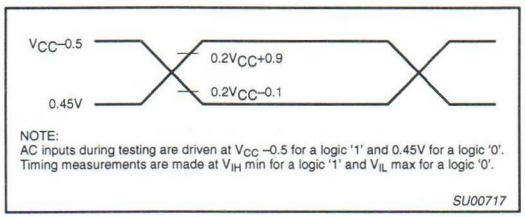



**Figure 18. AC Testing Input/Output**

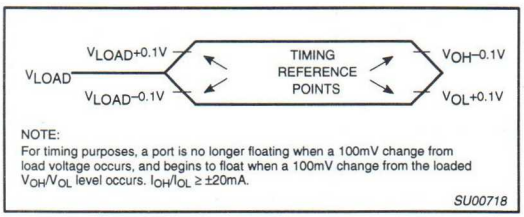



**Figure 19. Float Waveform**

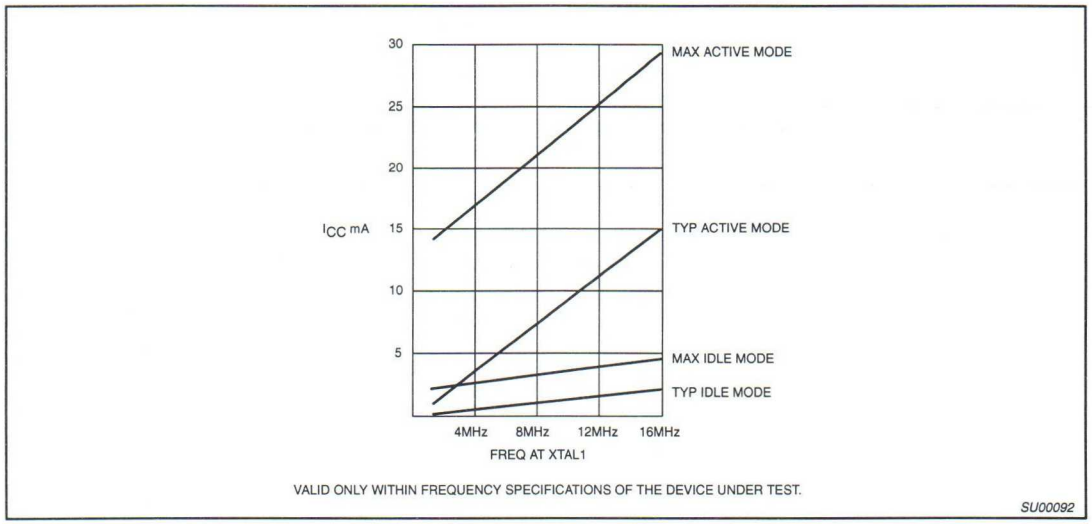



**Figure 20. $I_{CC}$ vs. FREQ**

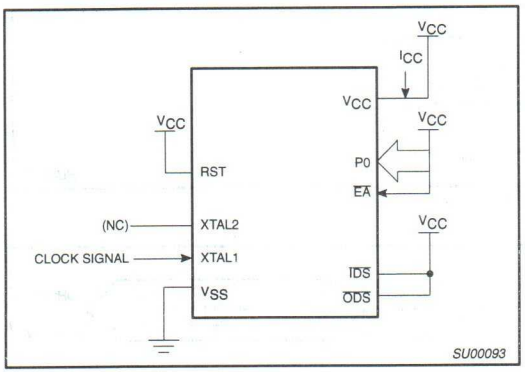



**Figure 21. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

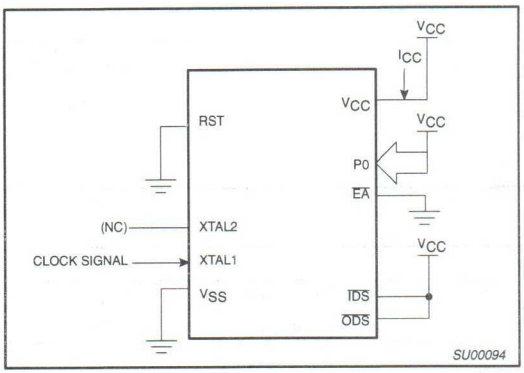



**Figure 22. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

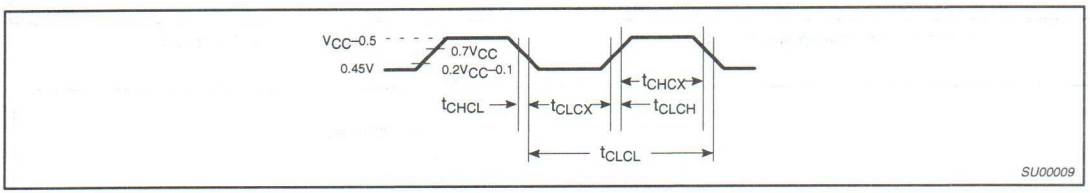



**Figure 23. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

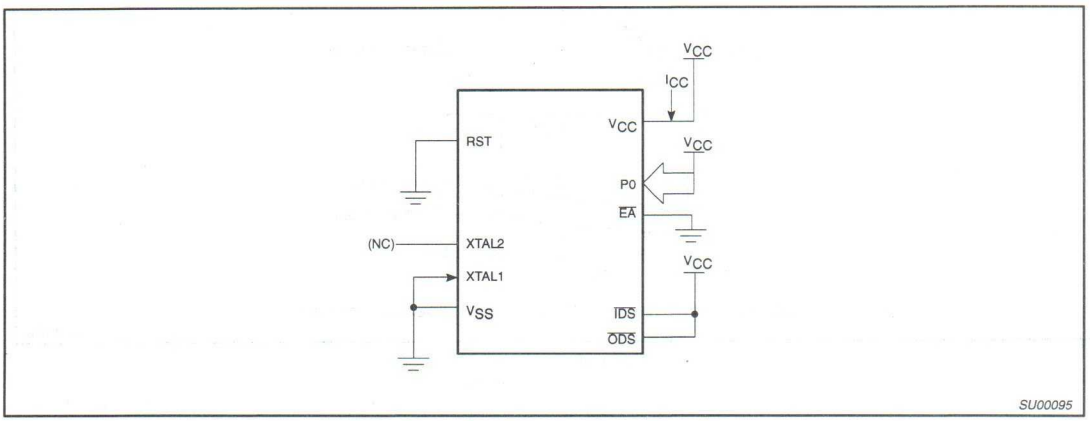



**Figure 24. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2V to 5.5V**

## EPROM CHARACTERISTICS

The 87C453 is programmed by using a modified Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C453 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C453 manufactured by Philips Semiconductors.

Table 4 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the lock bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 25 and 26. Figure 27 shows the circuit configuration for normal program memory verification.

### Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 26. Note that the 87C453 is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1 and 2, as shown in Figure 25. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 3 specified in Table 4 are held at the 'Program Code Data' levels indicated in Table 4. The ALE/$\overline{PROG}$ is pulsed low 15 to 25 times, as shown in Figure 26.

To program the encryption table, repeat the 15 to 25 pulse programming sequence for addresses 0 through 1FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the lock bits, repeat the 15 to 25 pulse programming sequence using the 'Pgm Lock Bit' levels. After one lock bit is programmed, further programming of the code memory and encryption table is disabled. However, the other lock bit can still be programmed.

Note that the $\overline{EA}$/$V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

#### Program Verification

If lock bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 1 and 2 as shown in Figure 27. The other pins are held at the 'Verify Code Data' levels indicated in Table 4. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

#### Reading the Signature Bytes

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:
(030H) = 15H indicates manufactured by Philips
(031H) = B9H indicates 87C453

### Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 4, and which satisfies the timing specifications, is suitable.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-sec/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

**Table 4. EPROM Programming Modes**

| MODE | RST | $\overline{PSEN}$ | ALE/$\overline{PROG}$ | $\overline{EA}$/$V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm lock bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm lock bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V ±10% during programming and verification.
* ALE/$\overline{PROG}$ receives 15 to 25 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

™Trademark phrase of Intel Corporation.

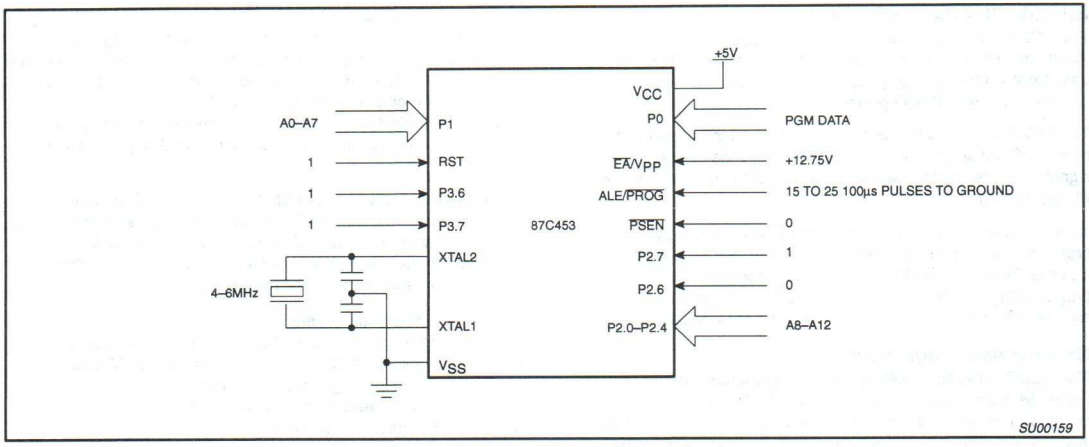



**Figure 25. Programming Configuration**

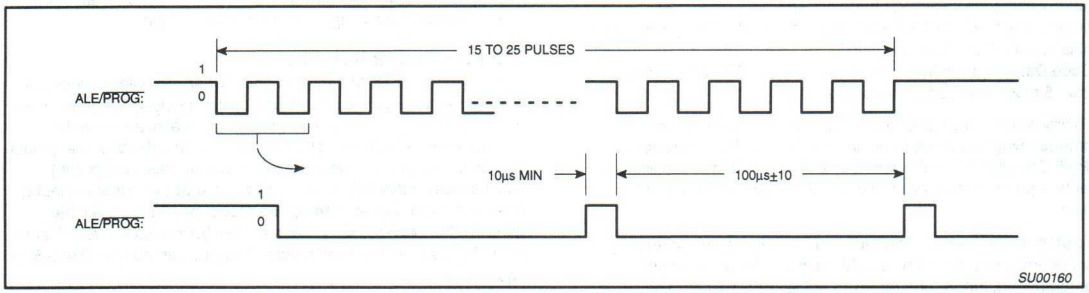



**Figure 26. PROG Waveform**

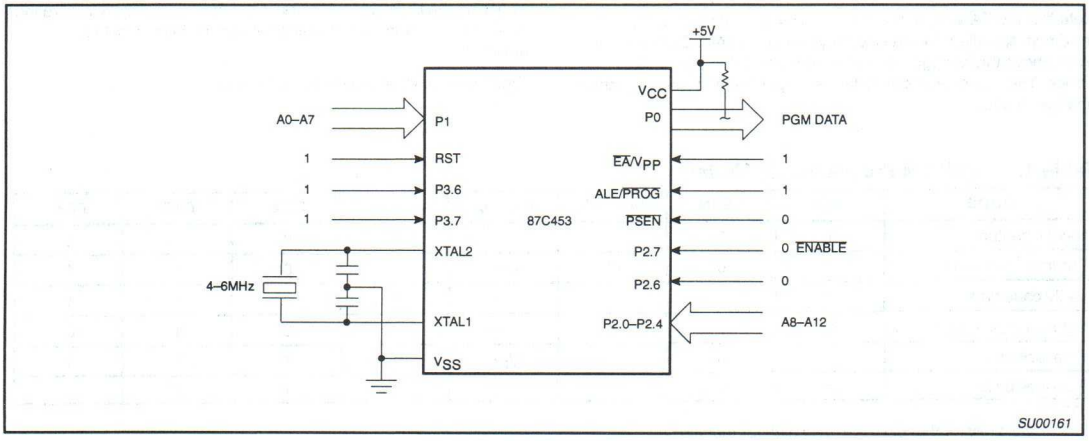
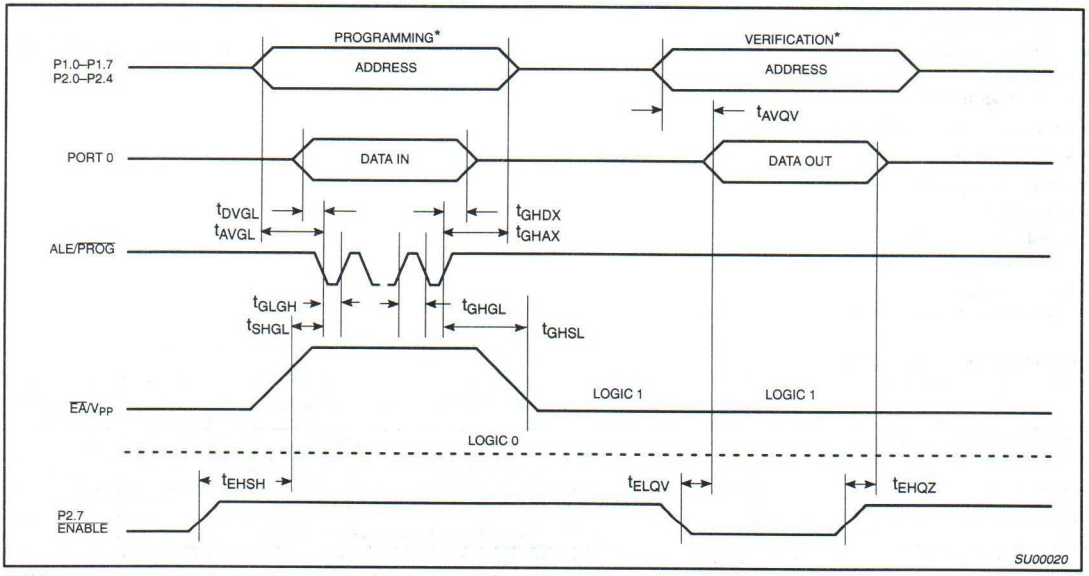



**Figure 27. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (See Figure 28)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




**NOTE:**

* FOR PROGRAMMING VERIFICATION SEE FIGURE 25.
FOR VERIFICATION CONDITIONS SEE FIGURE 27.

**Figure 28. EPROM Programming and Verification**

# CMOS single-chip 8-bit microcontrollers 83C508/87C508

## DESCRIPTION

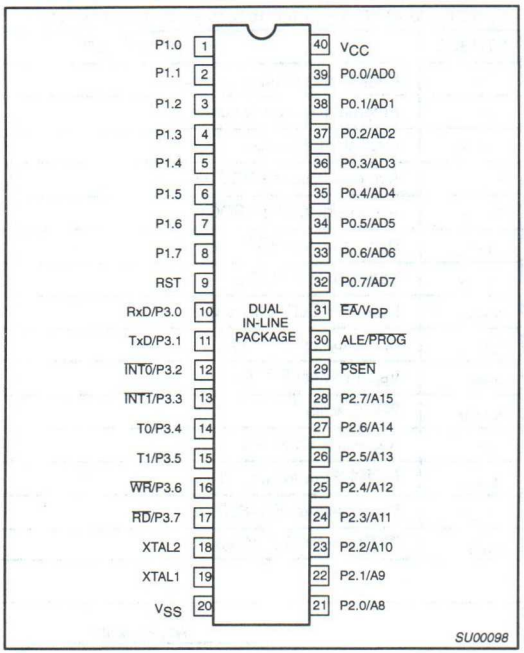
The 83C508 and 87C508 (hereafter referred to as 8XC508) Single-Chip 8-Bit Microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 8XC508 has the same instruction set as the 80C51.

This device provides architectural enhancements that make it applicable in a variety of applications for general control systems. The 8XC508 contains 32k × 8 EPROM memory, the 83C508 contains 32k × 8 ROM memory, a volatile 256 × 8 read/write data memory, four 8-bit I/O ports, two 16-bit timer/event counters, a multi-source, two-priority-level, nested interrupt structure, a 24-by-16 bit unsigned divide, an enhanced UART and on-chip oscillator and timing circuits. For systems that require extra capability, the 8XC508 can be expanded using standard TTL compatible memories and logic.

## FEATURES

- 80C51 central processing unit
- 32k × 8 EPROM expandable externally to 64k bytes
  - Quick Pulse programming algorithm
  - Two level program security system
- 256 × 8 RAM, expandable externally to 64k bytes
- Two 16-bit timer/counters
- Four 8-bit I/O ports
- Full-duplex enhanced UART
  - Framing error detection
  - Automatic address recognition
- Power control modes
  - Idle mode
  - Power-down mode
- Once (On Circuit Emulation) Mode
- Five package styles
- OTP package available
- 24-by-16 bit divide
  - Requires 4 machine cycles
  - 24-bit quotient and 16-bit remainder
- 24-by-16 multiply
  - Requires 4 machine cycles

## PIN CONFIGURATIONS

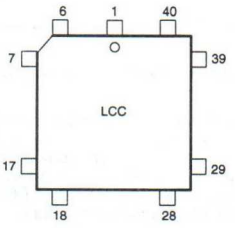



## ORDERING INFORMATION

| ROM | EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQ. (MHz) | DRAWING NUMBER |
|---|---|---|---|---|---|
| P83C508IBP N | P87C508IBP N | OTP | 0 to +70, 40-Pin Plastic Dual In-line Package | 3.5 to 24 | SOT129-1 |
| | P87C508IBF FA | UV | 0 to +70, 40-Pin Ceramic Dual In-line Package w/Window | 3.5 to 24 | 0590B |
| P83C508IBA A | P87C508IBA A | OTP | 0 to +70, 44-Pin Plastic Leaded Chip Carrier | 3.5 to 24 | SOT187-2 |
| | P87C508IBK KA | UV | 0 to +70, 44-Pin Ceramic Leaded Chip Carrier w/Window | 3.5 to 24 | 1472A |
| P83C508IBB BD | P87C508IBB BD | OTP | 0 to +70, 44-Pin Thin Quad Flat Pack | 3.5 to 24 | SOT389-1 |

**NOTE:**
1. OTP = One Time Programmable EPROM. UV = Erasable EPROM.

## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

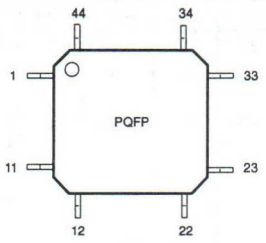



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | NC* | 16 | P3.4/T0 | 31 | P2.7/A15 |
| 2 | P1.0 | 17 | P3.5/T1 | 32 | $\overline{PSEN}$ |
| 3 | P1.1 | 18 | P3.6/$\overline{WR}$ | 33 | ALE/$\overline{PROG}$ |
| 4 | P1.2 | 19 | P3.7/$\overline{RD}$ | 34 | NC* |
| 5 | P1.3 | 20 | XTAL2 | 35 | $\overline{EA}$/$V_{PP}$ |
| 6 | P1.4 | 21 | XTAL1 | 36 | P0.7/AD7 |
| 7 | P1.5 | 22 | $V_{SS}$ | 37 | P0.6/AD6 |
| 8 | P1.6 | 23 | NC* | 38 | P0.5/AD5 |
| 9 | P1.7 | 24 | P2.0/A8 | 39 | P0.4/AD4 |
| 10 | RST | 25 | P2.1/A9 | 40 | P0.3/AD3 |
| 11 | P3.0/RxD | 26 | P2.2/A10 | 41 | P0.2/AD2 |
| 12 | NC* | 27 | P2.3/A11 | 42 | P0.1/AD1 |
| 13 | P3.1/TxD | 28 | P2.4/A12 | 43 | P0.0/AD0 |
| 14 | P3.2/$\overline{INT0}$ | 29 | P2.5/A13 | 44 | $V_{CC}$ |
| 15 | P3.3/$\overline{INT1}$ | 30 | P2.6/A14 | | |

* DO NOT CONNECT

*SU00002*

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS




| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | P1.5 | 16 | $V_{SS}$ | 31 | P0.6/AD6 |
| 2 | P1.6 | 17 | NC* | 32 | P0.5/AD5 |
| 3 | P1.7 | 18 | P2.0/A8 | 33 | P0.4/AD4 |
| 4 | RST | 19 | P2.1/A9 | 34 | P0.3/AD3 |
| 5 | P3.0/RxD | 20 | P2.2/A10 | 35 | P0.2/AD2 |
| 6 | NC* | 21 | P2.3/A11 | 36 | P0.1/AD1 |
| 7 | P3.1/TxD | 22 | P2.4/A12 | 37 | P0.0/AD0 |
| 8 | P3.2/$\overline{INT0}$ | 23 | P2.5/A13 | 38 | $V_{CC}$ |
| 9 | P3.3/$\overline{INT1}$ | 24 | P2.6/A14 | 39 | NC* |
| 10 | P3.4/T0 | 25 | P2.7/A15 | 40 | P1.0 |
| 11 | P3.5/T1 | 26 | $\overline{PSEN}$ | 41 | P1.1 |
| 12 | P3.6/$\overline{WR}$ | 27 | ALE/$\overline{PROG}$ | 42 | P1.2 |
| 13 | P3.7/$\overline{RD}$ | 28 | NC* | 43 | P.13 |
| 14 | XTAL2 | 29 | $\overline{EA}$/$V_{PP}$ | 44 | P1.4 |
| 15 | XTAL1 | 30 | P0.7/AD7 | | |

* DO NOT CONNECT

*SU00003*

## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER DIP | PIN NUMBER LCC | PIN NUMBER QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification and receives code bytes during EPROM programming. External pull-ups are required during program verification. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups, except P1.6 and P1.7 which are open drain. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also receives the low-order address byte during program memory verification. |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. Some Port 2 pins receive the high order address bits during EPROM programming and verification. |
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{CC}$. |
| ALE/$\overline{PROG}$ | 30 | 33 | 27 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the 8XC508 is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}/V_{PP}$ | 31 | 35 | 29 | I | **External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H and 3FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 3FFFH. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. If security bit 1 is programmed, $\overline{EA}$ will be internally latched on Reset. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTE:**
To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher than $V_{CC}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

## BLOCK DIAGRAM

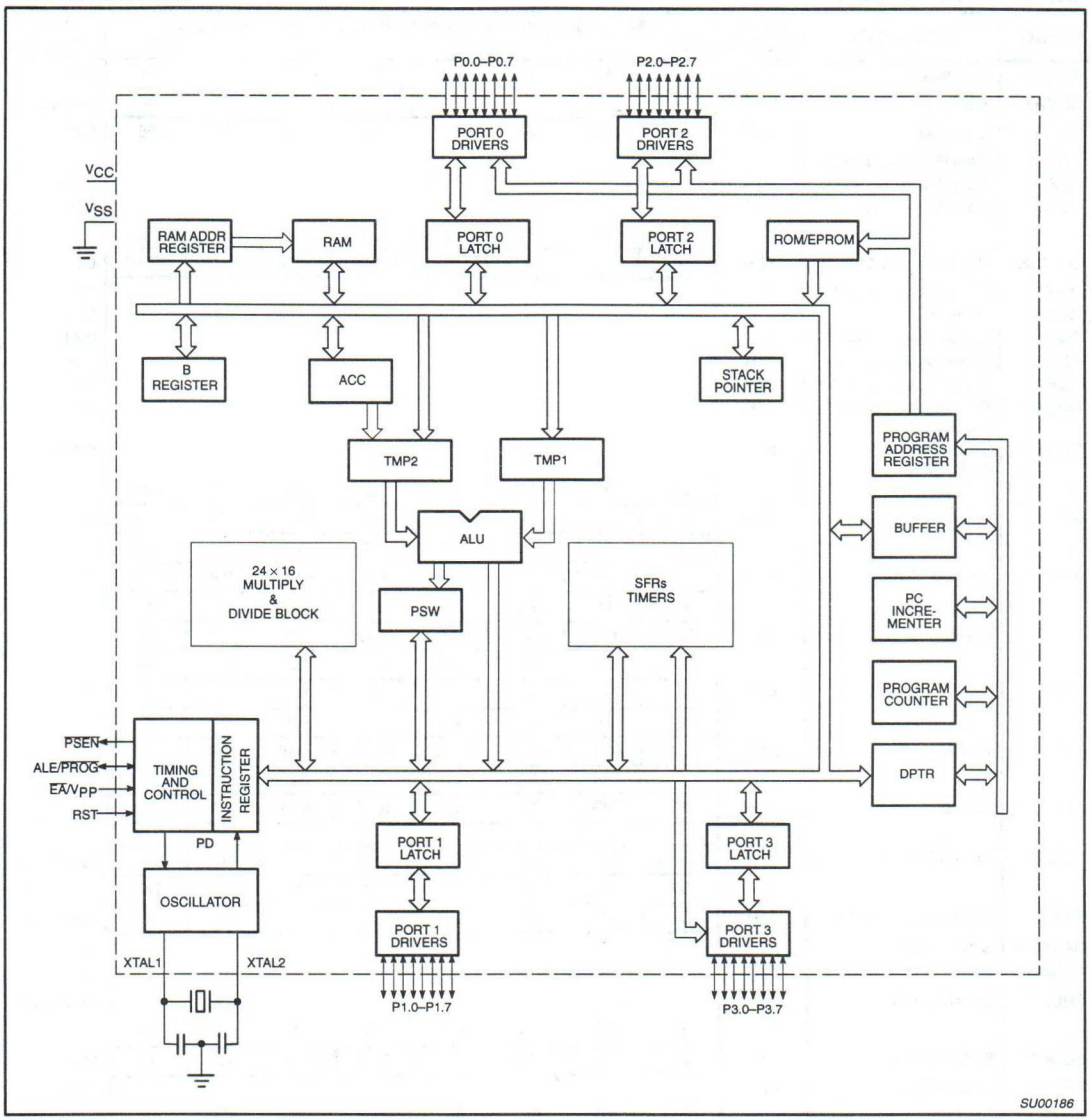


**Table 1. 87C508 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| AUXR# | Auxiliary | 8EH | – | – | – | – | – | – | – | A0 | xxxxxxx0B |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data Pointer (2 bytes) | | | | | | | | | | |
| DPH | Data Pointer High | 83H | | | | | | | | | 00H |
| DPL | Data Pointer Low | 82H | | | | | | | | | 00H |
| | | | FF | FE | FD | FC | FB | FA | F9 | F8 | |
| DMCON* | Divide/Multiply Control | F8H | – | – | – | – | DMMODE | AUTODM | DMSTRT | DMCTRL | 00H |
| DMB0 | Divide/Multiply byte 0 | 91H | | | | | | | | | 00H |
| DMB1 | Divide/Multiply byte 1 | 92H | | | | | | | | | 00H |
| DMB2 | Divide/Multiply byte 2 | 93H | | | | | | | | | 00H |
| DMB3 | Divide/Multiply byte 3 | 94H | | | | | | | | | 00H |
| DMB4 | Divide/Multiply byte 4 | 96H | | | | | | | | | 00H |
| DMOP1 | Divide/Multiply oper-and 1 (LSB) | 95H | | | | | | | | | 00H |
| DMOP2 | Divide/Multiply oper-and 2 (MSB) | 97H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE* | Interrupt Enable | A8H | EA | – | – | ES | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP* | Interrupt Priority | B8H | – | – | – | PS | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | – | – | – | – | – | – | – | – | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | AD15 | AD14 | AD13 | AD12 | AD11 | AD10 | AD9 | AD8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TxD | RxD | FFH |
| | | | | | | | | | | | |
| PCON# | Power Control | 87H | SMOD1 | SMOD0 | – | POF[1] | GF1 | GF0 | PD | IDL | 00xxxx00B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program Status Word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| SADDR# | Slave Address | A9H | | | | | | | | | 00H |
| SADEN# | Slave Address Mask | B9H | | | | | | | | | 00H |
| SBUF | Serial Data Buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial Control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack Pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer Control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| TH0 | Timer High 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer High 1 | 8DH | | | | | | | | | 00H |
| TL0 | Timer Low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer Low 1 | 8BH | | | | | | | | | 00H |
| TMOD | Timer Mode | 89H | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 | 00H |
| | | | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 | |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
1. Reset value depends on reset source.

## ENHANCED UART

The 8XC508 UART has all of the capabilities of the standard 80C51 UART plus Framing Error Detection and Automatic Address Recognition. As in the 80C51, all four modes of operation are supported as well as the 9th bit in modes 2 and 3 that can be used to facilitate multiprocessor communication.

The Framing Error Detection allows the UART to look for missing stop bits. If a Stop bit is missing, the FE bit in the SCON SFR is set. The FE bit can be checked after each transmission to detect communication errors. The FE bit can only be cleared by software and is not affected by a valid stop bit.

Automatic Address Recognition is used to reduce the CPU service time for the serial port. The CPU only needs to service the UART when it is addressed and, with this done by the on-chip circuitry, the need for software overhead is greatly reduced. This mode works similar to the 9th bit communication mode, except that uses only 8 bits and the Stop bit is used to cause the RI bit to be set. There are two SFRs associated with this mode. They are SADDR, which holds the slave address and SADEN, which contains a mask that allows selective masking of the slave address so that broadcast addresses can be used.

## HARDWARE MULTIPLY/DIVIDE UNIT

The 8XC508 contains a 24-by-16 bit hardware divide unit. The 24 bit dividend is stored in special function registers DMB2 – DMB0 and the divisors are in registers DMOP1 (LSB) and DMOP2 (MSB). A division operation returns the 24-bit result in registers DMB2 – DMB0 and a 16-bit remainder in register DMB3 (LSB) and DMB4 (MSB).

The divide unit provides two modes of operation, auto-start and flag-controlled. Auto-start mode is enabled by setting the AUTODM (auto divide/multiply) bit in the DMCON (divide/multiply control) register. If auto-start mode is enabled, writing to the divisor (DMOP1) will automatically start a division operation and will set the DMCTRL (divide/multiply control) and DMSTRT bits in the DMCON register. DMCTRL will automatically be cleared by the divide hardware when the division operation has been completed.

Flag controlled operation is initiated by setting the DMCTRL bit in the DMCON register which will start the division operation and also set the DMSTRT bit. The DMCTRL bit will automatically be cleared by the divide hardware when the division operation has been completed. DMSTRT can only be cleared by software.

The hardware divide unit of the 87C508 can also be used to perform a 24-by-16 bit multiply operation. Multiplication operation is selected by the DMMODE bit in the DMCON register (1 = multiply, 0 = divide). The 24-bit multiplicand is stored in the DMB2 – DMB0 registers. The multiplier is stored in the DMOP1 (LSB) and DMOP2 (MSB) registers. The 40 bit result of the multiply operation is stored in the DMB4 – DMB0 registers. Like division, the multiplication can be automatically started or software started. Autostart mode is enabled by setting the AUTODM bit in the DMCON register. Writing a multiplier to the DMOP1 register will automatically start a multiplication operation and will set the DMCTRL and DMSTRT bits in the DMCON register. DMCTRL will automatically be cleared by the multiply hardware when the multiplication operation is complete.

Flag controlled operation is initiated by setting the DMCTRL bit in the DMCON register, which will start the multiplication operation and also set the DMSTRT bit. The DMCTRL bit will automatically be cleared by the multiply hardware when the multiplication operation has been completed. DMSTRT can only be cleared by software.

## POWER OFF FLAG

The Power Off Flag (POF) is set by on-chip circuitry when the $V_{CC}$ level on the 8XC508 rises from 0 to 5V. The POF bit can be set or cleared by software allowing a user to determine if the reset is the result of a power-on or a warm start after powerdown. The $V_{CC}$ level must remain above 3V for the POF to remain unaffected by the $V_{CC}$ level.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

### Reset

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up.

### Idle Mode

In the idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

### Power-Down Mode

To save even more power, a Power Down mode can be invoked by software. In this mode, the oscillator is stopped and the instruction that invoked Power Down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the Power Down mode is terminated.

On the 8XC508 either a hardware reset or external interrupt can use an exit from Power Down. Reset redefines all the SFRs but does not change the on-chip RAM. An external interrupt allows both the SFRs and the on-chip RAM to retain their values.

To properly terminate Power Down the reset or external interrupt should not be executed before $V_{CC}$ is restored to its normal operating level and must be held active long enough for the oscillator to restart and stabilize (normally less than 10ms).

With an external interrupt, INT0 and INT1 must be enabled and configured as level-sensitive. Holding the pin low restarts the oscillator but bringing the pin back high completes the exit. Once the interrupt is serviced, the next instruction to be executed after RETI will be the one following the instruction that put the device into Power Down.

## Design Consideration

- When the idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal rest algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

## ONCE™ Mode

The ONCE ("On-Circuit Emulation") Mode facilitates testing and debugging of systems using the 8XC508 without the 8XC508 having to be removed from the circuit. The ONCE Mode is invoked by:

1. Pull ALE low while the device is in reset and $\overline{PSEN}$ is high;
2. Hold ALE low as RST is deactivated.

While the device is in ONCE Mode, the Port 0 pins go into a float state, and the other port pins and ALE and $\overline{PSEN}$ are weakly pulled high. The oscillator circuit remains active. While the 8XC508 is in this mode, an emulator or test CPU can be used to drive the circuit. Normal operation is restored when a normal reset is applied.

**Table 2. External Pin Status During Idle and Power-Down Mode**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | 0 to +70 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Maximum $I_{OL}$ per I/O pin | 15 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.5 | W |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$ | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | | 0 | | $0.2V_{CC}$–0.3 | V |
| $V_{IH}$ | Input high voltage | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3[8] | $I_{OL}$ = 1.6mA[2] | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$[8] | $I_{OL}$ = 3.2mA[2] | | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3, ALE, $\overline{PSEN}$[3] | $I_{OH}$ = –30µA | $V_{CC}$ – 0.7 | | | V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode), ALE[9], $\overline{PSEN}$[3] | $I_{OH}$ = –3.2mA | $V_{CC}$ – 0.7 | | | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3 | $V_{IN}$ = 0.45V | | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3 | See note 4 | | | –650 | µA |
| $I_{LI}$ | Input leakage current, port 0 | 0.45 $V_{IN}$ < $V_{CC}$ – 0.3 | | | ±10 | µA |
| $I_{CC}$ | Power supply current:<br>Active mode @ 16MHz[5]<br>Idle mode @ 16MHz<br>Power-down mode | See note 6 | | <br>15<br>3<br>10 | <br>32<br>5<br>50 | <br>mA<br>mA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | 50 | | 225 | kΩ |
| $C_{IO}$ | Pin capacitance[10] (except EA) | | | | 15 | pF |

**NOTES:**

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
5. $I_{CCMAX}$ at other frequencies is given by: Active mode: $I_{CCMAX}$ = 1.50 × FREQ + 8: Idle mode: $I_{CCMAX}$ = 0.14 × FREQ +2.31, where FREQ is the external oscillator frequency in MHz. $I_{CCMAX}$ is given in mA. See Figure 8.
6. See Figures 9 through 12 for $I_{CC}$ test conditions.
7. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
8. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 15mA (*NOTE: This is 85°C specification.)
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 71mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
9. ALE is tested to $V_{OH1}$, except when ALE is off then $V_{OH}$ is the voltage specification.
10. Pin capacitance is less than 25pF. Pin capacitance of ceramic package is less than 15pF (except EA is 25pF).

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2, 3]

| SYMBOL | FIGURE | PARAMETER | 24MHz CLOCK | | VARIABLE CLOCK[4] | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 1 | Oscillator frequency<br>Speed versions : I | <br>3.5 | <br>24 | 3.5 | 24 | MHz<br>MHz |
| $t_{LHLL}$ | 1 | ALE pulse width | 43 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 1 | Address valid to ALE low | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{LLAX}$ | 1 | Address hold after ALE low | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{LLIV}$ | 1 | ALE low to valid instruction in | | 102 | | $4t_{CLCL}-65$ | ns |
| $t_{LLPL}$ | 1 | ALE low to $\overline{PSEN}$ low | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{PLPH}$ | 1 | $\overline{PSEN}$ pulse width | 80 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 1 | $\overline{PSEN}$ low to valid instruction in | | 65 | | $3t_{CLCL}-60$ | ns |
| $t_{PXIX}$ | 1 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 1 | Input instruction float after $\overline{PSEN}$ | | 17 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 1 | Address to valid instruction in | | 128 | | $5t_{CLCL}-80$ | ns |
| $t_{PLAZ}$ | 1 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 2, 3 | $\overline{RD}$ pulse width | 150 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 2, 3 | $\overline{WR}$ pulse width | 150 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 2, 3 | $\overline{RD}$ low to valid data in | | 118 | | $5t_{CLCL}-90$ | ns |
| $t_{RHDX}$ | 2, 3 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 2, 3 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-28$ | ns |
| $t_{LLDV}$ | 2, 3 | ALE low to valid data in | | 183 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 2, 3 | Address to valid data in | | 210 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 2, 3 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 75 | 175 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 2, 3 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 92 | | $4t_{CLCL}-75$ | | ns |
| $t_{QVWX}$ | 2, 3 | Data valid to $\overline{WR}$ transition | 12 | | $t_{CLCL}-30$ | | ns |
| $t_{WHQX}$ | 2, 3 | Data hold after $\overline{WR}$ | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{QVWH}$ | 3 | Data valid to $\overline{WR}$ high | 162 | | $7t_{CLCL}-130$ | | ns |
| $t_{RLAZ}$ | 2, 3 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 2, 3 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 17 | 67 | $t_{CLCL}-25$ | $t_{CLCL}+25$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 5 | High time | 17 | | 10 | $t_{CLCL}-t_{CLCX}$ | ns |
| $t_{CLCX}$ | 5 | Low time | 17 | | 10 | $t_{CLCL}-t_{CHCX}$ | ns |
| $t_{CLCH}$ | 5 | Rise time | | 5 | | 5 | ns |
| $t_{CHCL}$ | 5 | Fall time | | 5 | | 5 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 4 | Serial port clock cycle time | 505 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 4 | Output data setup to clock rising edge | 283 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 4 | Output data hold after clock rising edge | 3 | | $2t_{CLCL}-80$ | | ns |
| $t_{XHDX}$ | 4 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 4 | Clock rising edge to input data valid | | 283 | | $10t_{CLCL}-133$ | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 8XC508 to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.
4. Variable clock is spcified for oscillator frequencies greater than 3.5MHz to 24MHz for "I" version.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{\text{PSEN}}$
Q – Output data
R – $\overline{\text{RD}}$ signal
t – Time
V – Valid
W – $\overline{\text{WR}}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{\text{PSEN}}$ low.

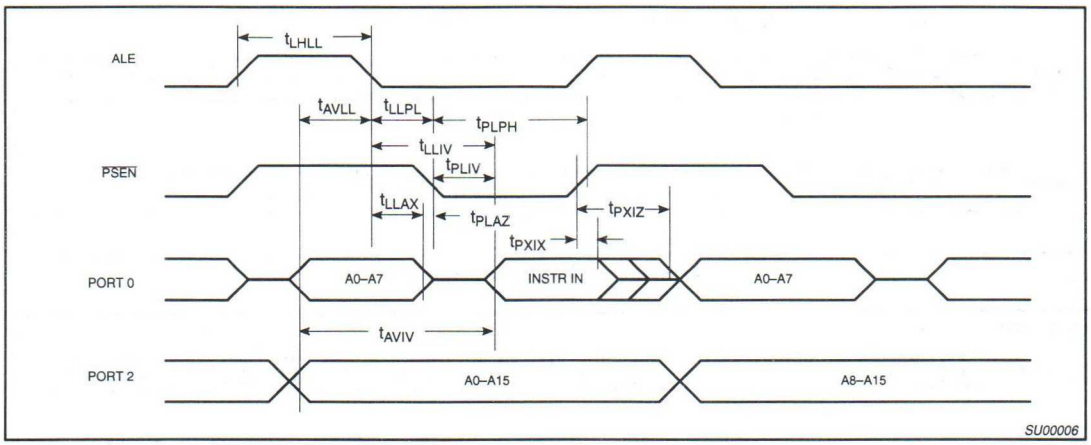



**Figure 1. External Program Memory Read Cycle**

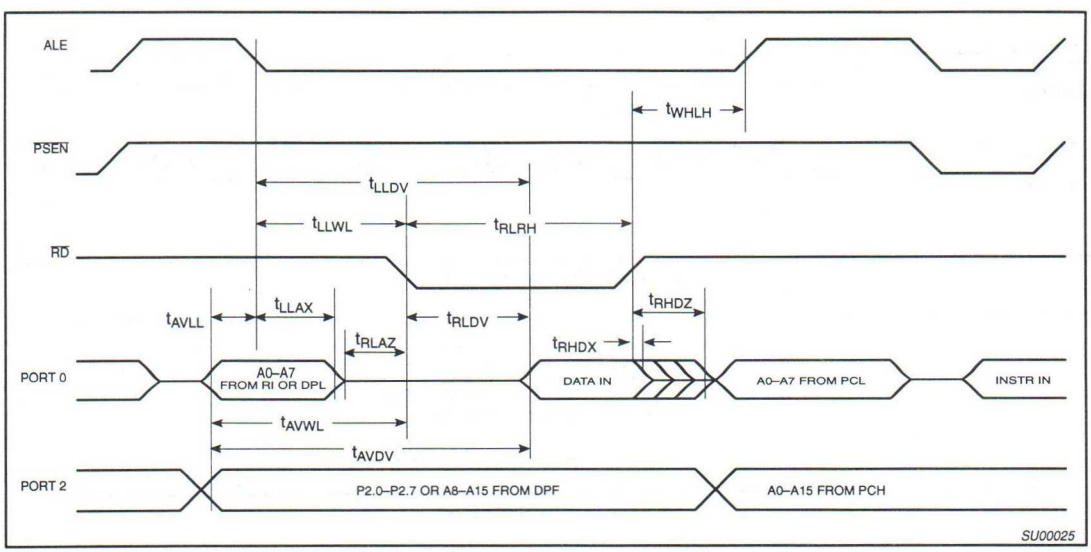



**Figure 2. External Data Memory Read Cycle**

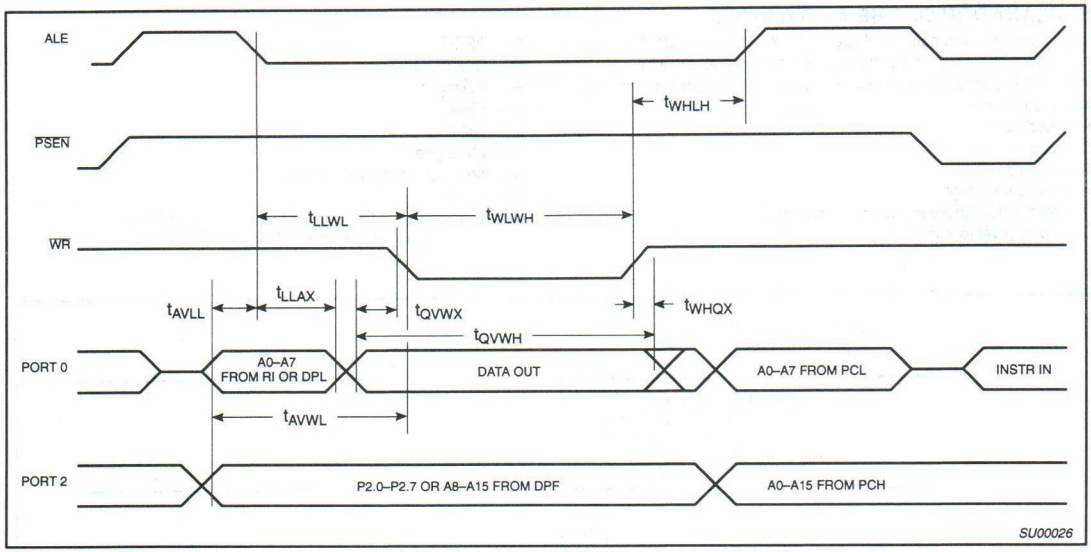



**Figure 3. External Data Memory Write Cycle**

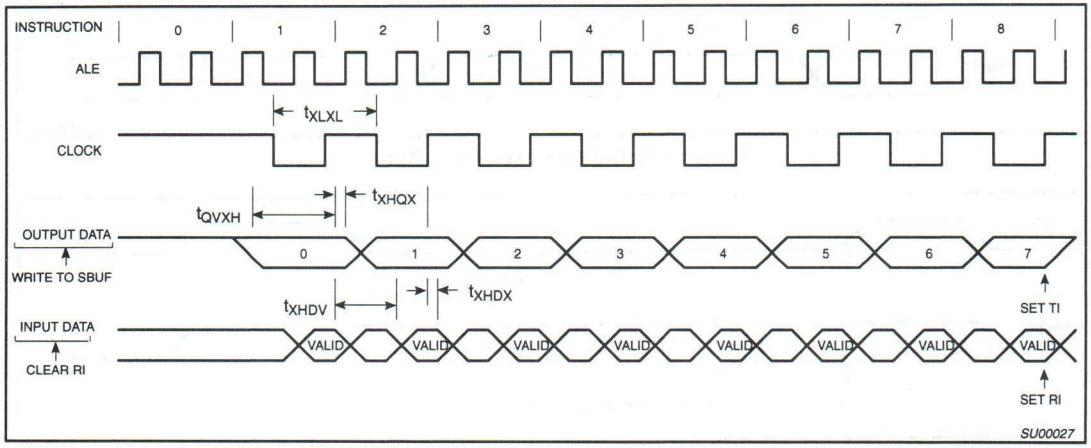



**Figure 4. Shift Register Mode Timing**

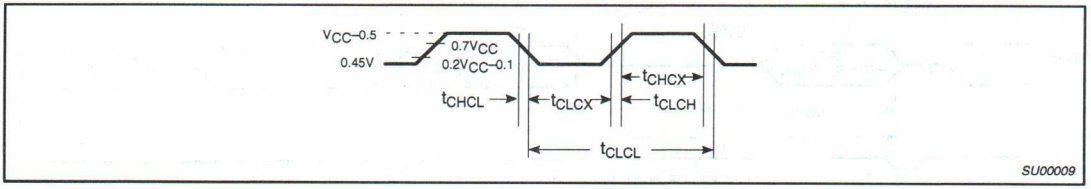
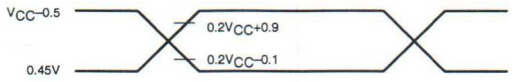


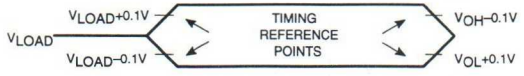

**Figure 5. External Clock Drive**




NOTE:
AC inputs during testing are driven at $V_{CC}$ –0.5 for a logic '1' and 0.45V for a logic '0'. Timing measurements are made at $V_{IH}$ min for a logic '1' and $V_{IL}$ max for a logic '0'.

SU00010

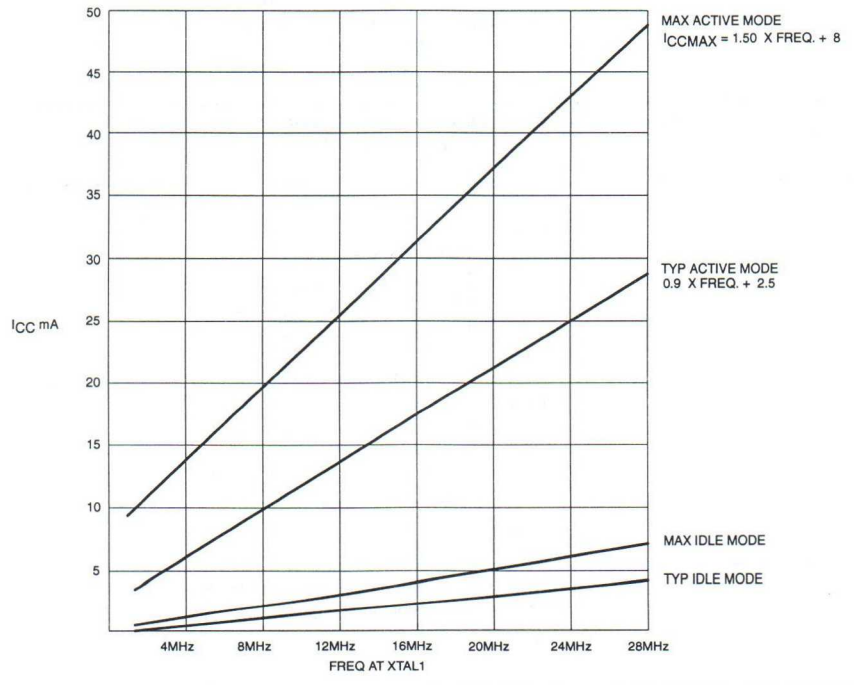
**Figure 6. AC Testing Input/Output**




NOTE:
For timing purposes, a port is no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded $V_{OH}/V_{OL}$ level occurs. $I_{OH}/I_{OL} \geq \pm 20$mA.

SU00011

**Figure 7. Float Waveform**




SU00713

**Figure 8. $I_{CC}$ vs. FREQ Valid only within frequency specifications of the device under test**

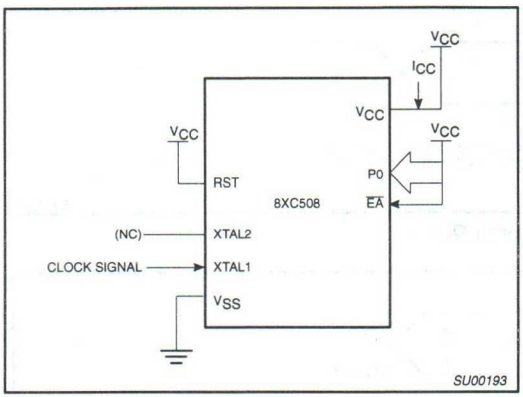



**Figure 9. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

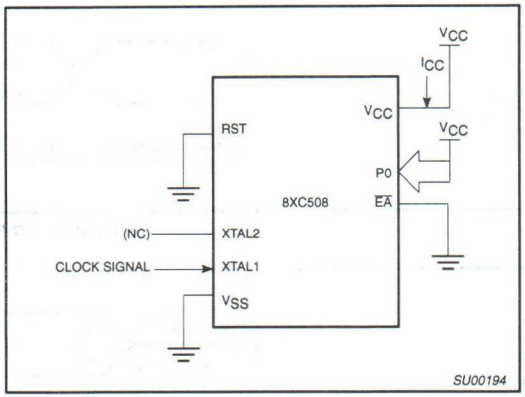



**Figure 10. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

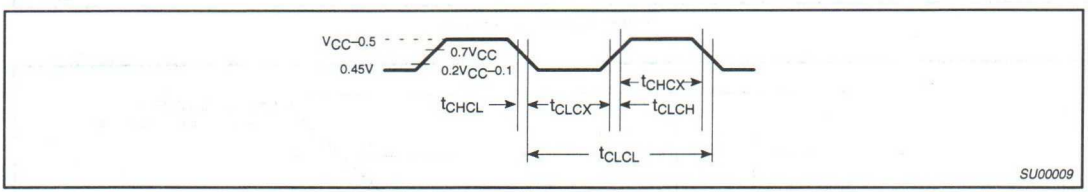



**Figure 11. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

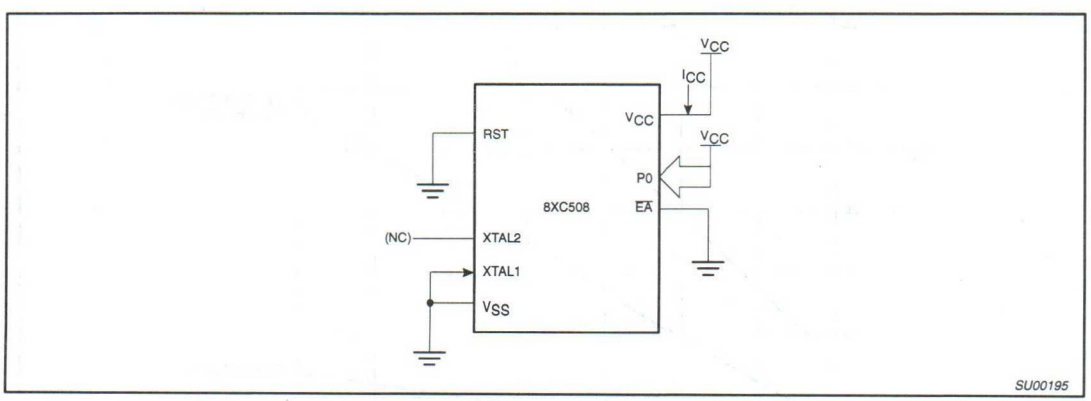



**Figure 12. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2V to 5.5V**

## EPROM CHARACTERISTICS

The 87C508 is programmed by using a modified Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C508 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C508 manufactured by Philips.

Table 3 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the security bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 13 and 14. Figure 15 shows the circuit configuration for normal program memory verification.

### Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 13. Note that the 87C508 is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1 and 2, as shown in Figure 13. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 3 specified in Table 3 are held at the 'Program Code Data' levels indicated in Table 3. The ALE/$\overline{PROG}$ is pulsed low 25 times as shown in Figure 14.

To program the encryption table, repeat the 25 pulse programming sequence for addresses 0 through 1FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the security bits, repeat the 25 pulse programming sequence using the 'Pgm Security Bit' levels. After one security bit is programmed, further programming of the code memory and encryption table is disabled. However, the other security bit can still be programmed.

Note that the $\overline{EA}/V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

#### Program Verification

If security bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 1 and 2 as shown in Figure 15. The other pins are held at the 'Verify Code Data' levels indicated in Table 3. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

#### Reading the Signature Bytes

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:
(030H) = 15H indicates manufactured by Philips
(031H) = BDH indicates 87C508

### Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 3, and which satisfies the timing specifications, is suitable.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-s/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

**Table 3. EPROM Programming Modes**

| MODE | RST | $\overline{PSEN}$ | ALE/$\overline{PROG}$ | $\overline{EA}/V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 | P3.3 |
|---|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 | 1 |
| Pgm security bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 | 1 |
| Pgm security bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 | 1 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V±10% during programming and verification.

* ALE/$\overline{PROG}$ receives 5 programming pulses (only for user array; 25 pulses for encryption or security bits) while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

™Trademark phrase of Intel Corporation.

### Table 4. Program Security Bits

| PROGRAM LOCK BITS[1, 2] | | | |
|---|---|---|---|
| | SB1 | SB2 | PROTECTION DESCRIPTION |
| 1 | U | U | No Program Security features enabled. (Code verify will still be encrypted by the Encryption Array if programmed.) |
| 2 | P | U | MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory, $\overline{EA}$ is sampled and latched on Reset, and further programming of the EPROM is disabled. |
| 3 | P | P | Same as 2, also verify is disabled. |

**NOTES:**
1. P – programmed. U – unprogrammed.
2. Any other combination of the security bits is not defined.

## ROM CODE SUBMISSION

When submitting ROM code for the 83C508, the following must be specified:
1. 16k byte user ROM data
2. 32 byte ROM encryption key
3. ROM security bits.

If submitting a file, the format is as follows:

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 7FFFH | DATA | 7:0 | User ROM Data |
| 8000H to 801FH | KEY | 7:0 | ROM Encryption Key<br>FFH = no encryption |
| 8020H | SEC | 0 | ROM Security Bit 1<br>0 = enable security<br>1 = disable security |
| 8020H | SEC | 1 | ROM Security Bit 2<br>0 = enable security<br>1 = disable security |

**Security Bit 1:** When programmed, this bit has two effects on masked ROM parts:
1. External MOVC is disabled, and
2. $\overline{EA}$ is latched on Reset.

**Security Bit 2:** When programmed, this bit inhibits Verify User ROM.

If the file does not include the options, the following information must be included with the ROM code.

for each of the following, check the appropriate state and send to Philips along with the code:

Security Bit #1: ☐ Enabled ☐ Disabled

Security Bit #2: ☐ Enabled ☐ Disabled

Encryption: ☐ No ☐ Yes If Yes, must send key file.

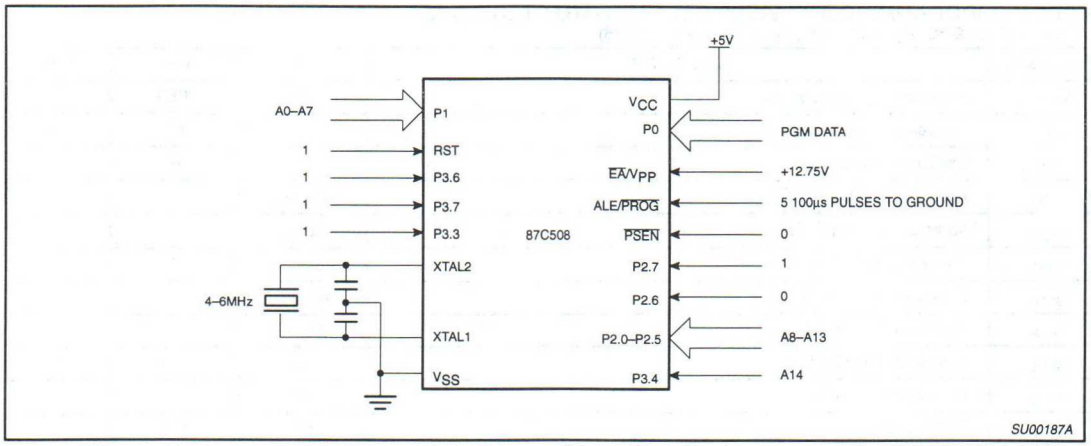



**Figure 13. Programming Configuration**

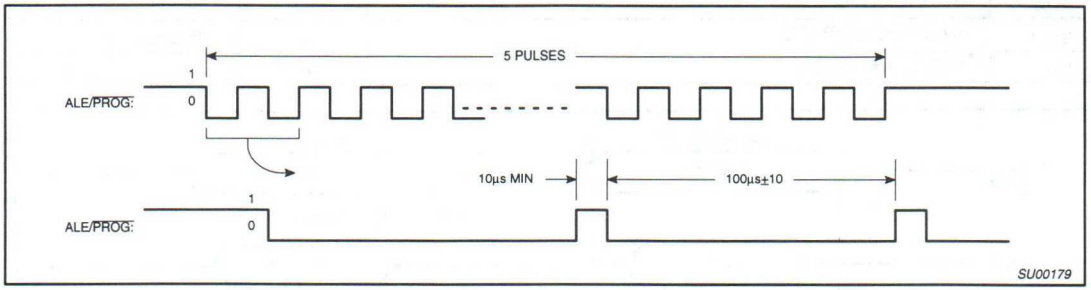



**Figure 14. PROG Waveform**

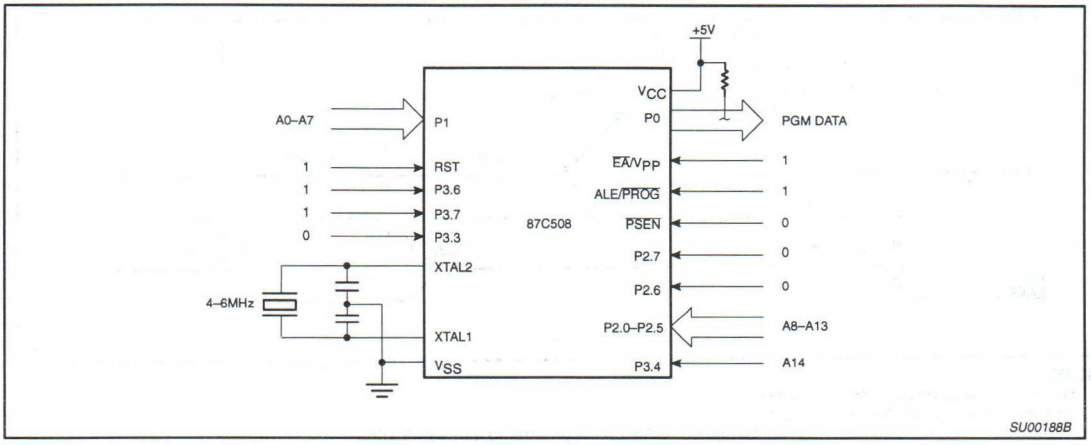
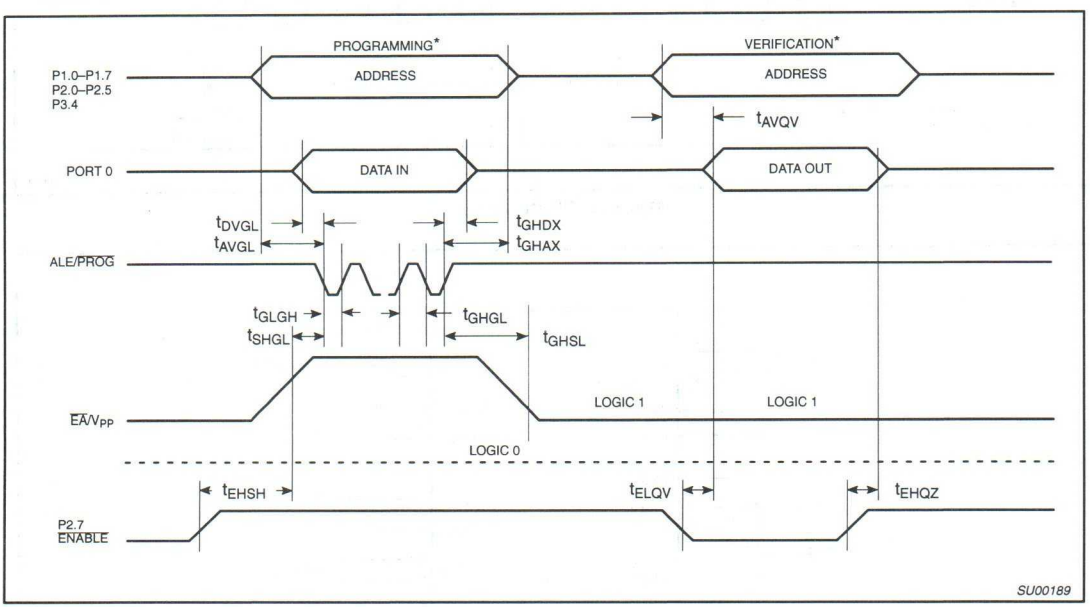



**Figure 15. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (See Figure 16)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




**NOTE:**

* FOR PROGRAMMING VERIFICATION SEE FIGURE 13.
FOR VERIFICATION CONDITIONS SEE FIGURE 15.

**Figure 16. EPROM Programming and Verification**

# 8-bit microcontroller P83C524

## FEATURES

- 80C51 CPU
- 16 kbytes on-chip ROM, expandable externally to 64 kbytes Program Memory address space
- 512 bytes on-chip RAM, expandable externally to 64 kbytes Data Memory address space
- Four 8-bit I/O ports
- Full-duplex UART compatible with the standard 80C51 and the 8052
- Two standard 16-bit timer/counters
- An additional 16-bit timer (functionally equivalent to the timer 2 of the 8052)
- On-chip Watchdog Timer (WDT) with a separate on-chip oscillator
- Bit-level I²C-bus hardware serial I/O Port
- 7-source and 7-vector interrupt structure with 2 prioirty levels
- Up to 3 external interrupt request inputs
- Two programmable power reduction modes (Idle and Power-down)
- Termination of Idle mode by any interrupt, external or WDT (watchdog) reset
- Wake-up from Power-down by external interrupt, external or WDT reset
- ROM code protection
- XTAL frequency range: 1.2MHz to 16MHz
- All packaging pin-outs fully compatible to the standard 8051/8052.

## GENERAL DESCRIPTION

The P83C524 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The P83C524 is a stand-alone high-performance microcontroller designed for use in real time applications such as instrumentation, industrial control, medium to high-end consumer applications and specific automotive control applications.

The P83C524 contains a non-volatile 16K × 8 read-only program memory, a volatile 512 × 8 read/write data memory, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a 16-bit timer (identical to the timer 2 of the 8052), a multi-source, two-priority-level, nested interrupt structure, two serial interfaces (UART and bit-level I²C-bus), an on-chip oscillator and timing circuits, a watchdog timer (WDT) with a separate on-chip oscillator. For systems that require extra capability, the P83C524 can be expanded using standard TTL compatible memories and logic.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The P83C524 has the same instruction set as the PCB80C51 which consists of over 100 instructions: 49 one-byte, 46 two-byte and 16 three-byte. With a 16MHz crystal, 58% of the instructions are executed in 750ns and 40% in 1.5µs. Multiply and divide instructions require 3µs.

## ORDERING INFORMATION

| EXTENDED TYPE NUMBER | PACKAGE | | | | TEMPERATURE RANGE (°C) | FREQUENCY (MHz) |
|---|---|---|---|---|---|---|
| | PINS | PIN POSITION | MATERIAL | CODE | | |
| **ROM (Note 1)** | | | | | | |
| P83C524FBP | 40 | DIL | plastic | SOT129-1 | 0 to +70 | 1.2 to 16 |
| P83C524FFP | 40 | DIL | plastic | SOT129-1 | –40 to +85 | 1.2 to 16 |
| P83C524FHP | 40 | DIL | plastic | SOT129-1 | –40 to +125 | 1.2 to 16 |
| P83C524FBA | 44 | PLCC | plastic | SOT187-2 | 0 to +70 | 1.2 to 16 |
| P83C524FFA | 44 | PLCC | plastic | SOT187-2 | –40 to +85 | 1.2 to 16 |
| P83C524FHA | 44 | PLCC | plastic | SOT187-2 | –40 to +125 | 1.2 to 16 |

**NOTE:**
1. For EPROM types, refer to the 87C524 data sheet.

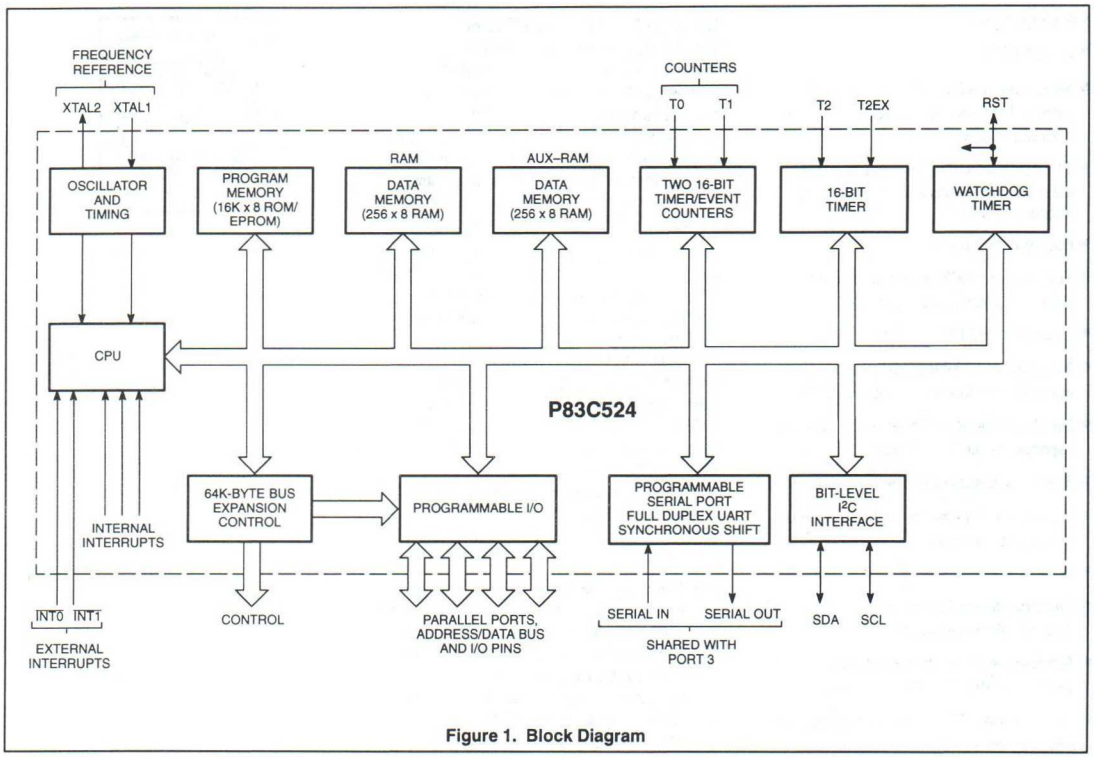



**Figure 1. Block Diagram**

## PINNING INFORMATION

### Pinning

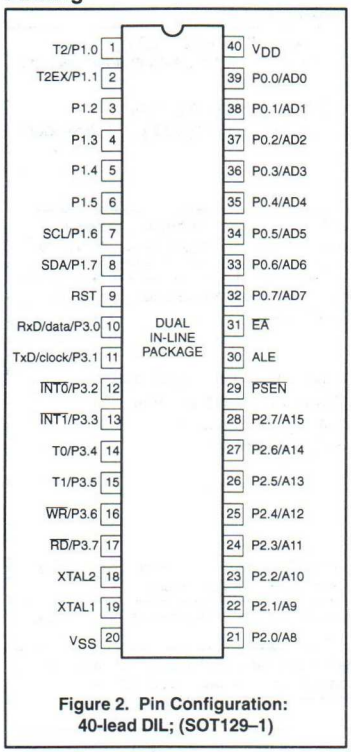



**Figure 2. Pin Configuration: 40-lead DIL; (SOT129–1)**

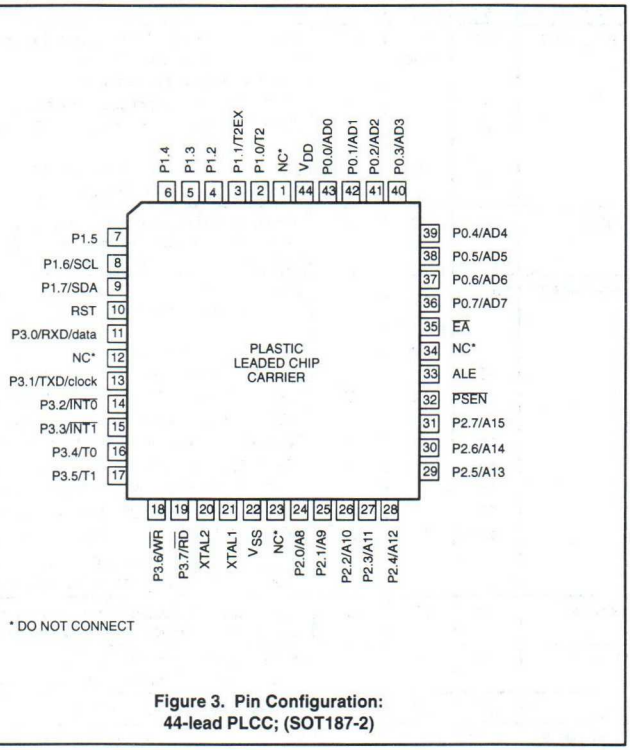



**Figure 3. Pin Configuration: 44-lead PLCC; (SOT187-2)**

## PIN DESCRIPTION

| MNEMONIC | PIN NO. DIP | PIN NO. PLCC | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|
| P1.0–P1.7 | 1–8 | 2–9 (1 NC) | I/O | **Port 1:** 8-bit quasi-bidirectional I/O port. Port 1 can sink/source one TTL (4 LS TTL) input. It can drive CMOS inputs without external pull-ups, except P1.6 and P1.7 which have open drain outputs. |
| | | | | **Port 1 alternative functions:** |
| | 1 | 2 | I | **T2 (P1.0):** Timer/event counter 2 external event counter input (falling edge triggered). |
| | 2 | 3 | I | **T2EX (P1.1):** Timer/event counter 2 capture/reload trigger or external interrupt 2 input (falling edge triggered) |
| | 7 | 8 | I/O | **SCL (P1.6):** $I^2C$ serial port clock line. |
| | 8 | 9 | I/O | **SDA (P1.7):** $I^2C$ serial port data line. |
| RST | 9 | 10 | I/O | **Reset:** A HIGH level on this pin for two machine cycles while the oscillator is running, resets the device. An internal pull-down resistor permits power-on reset using only a capacitor connected to $V_{DD}$. After a WDT overflow, this pin is pulled HIGH while the internal reset signal is active. |
| P3.0–P3.7 | 10–17 | 11, 13–19 (12 NC) | I/O | **Port 3:** 8-bit quasi-bidirectional I/O port with internal pull-ups. Port 3 can sink/source one TTL (=4 LS TTL) input. It can drive CMOS inputs without external pull-ups. |
| | | | | **Port 3 alternative functions:** |
| | 10 | 11 | I | **RxD/data (P3.0):** Serial Port data input (asynchonous) or data input/output (synchronous). |
| | 11 | 13 | O | **TxD/clock (P3.1):** Serial Port data output (asynchronous) or clock output (synchronous). |
| | 12 | 14 | I | **$\overline{INT0}$ (P3.2):** External interrupt 0 or gate control input for timer/event counter 0. |
| | 13 | 15 | I | **$\overline{INT1}$ (P3.3):** External interrupt 1 or gate control input for timer/event counter 1. |
| | 14 | 16 | I | **T0 (P3.4):** External input for timer/event counter 0. |
| | 15 | 17 | I | **T1 (P3.5):** External input for timer/event counter 1. |
| | 16 | 18 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe. |
| | 17 | 19 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe. |
| | | | | The generation or use of a Port 3 pin as an alternative function is carried out automatically by the P83C524 provided the associated Special Function Register (SFR) bit is set HIGH. |
| XTAL2 | 18 | 20 | O | **Crystal input 2:** Output of the inverting amplifier that forms the oscillator. This pin left open-circuit when an external oscillator clock is used. |
| XTAL1 | 19 | 21 | I | **Crystal input 1:** Input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external oscillator clock signal when an external oscillator is used. |
| $V_{SS}$ | 20 | 22 | I | **Ground:** Circuit ground potential. |
| P2.0–P2.7 (A8 to A15) | 21–28 | 24–31 (23 NC) | I/O | **Port 2:** 8-bit quasi-bidirectional I/O port with internal pull-ups. During access to external memories (RAM/ROM) that use 16-bit addresses (MOVX @DPTR) Port 2 emits the high-order address byte (A8 to A15). Port 2 can sink/source one TTL (=4 LS TTL) input. It can drive CMOS inputs without external pull-ups. |
| $\overline{PSEN}$ | 29 | 32 | O | **Program Store Enable output:** Read strobe to external program memory via Port 0 and Port 2. It is activated twice each machine cycle during fetches from external program memory. When executing out of external program memory two activations of $\overline{PSEN}$ are skipped during each access to external data memory. $\overline{PSEN}$ is not activated (remains HIGH) during fetches from external program memory. $\overline{PSEN}$ can sink source 8 LS TTL inputs. It can drive CMOS inputs without external pull-ups. |
| ALE | 30 | 33 | I/O | **Address Latch Enable output:** Latches the LOW byte of the address during access to external memory in normal operation. It is activated every six oscillator periods except during an external data memory access. ALE can sink/source 8 LS TTL inputs. It can drive CMOS inputs without an external pull-up. |
| $\overline{EA}$ | 31 | 35 (34 NC) | I | **External Access input:** When during RESET, $\overline{EA}$ is held at a TTL HIGH level, the CPU executes out of the internal program ROM, provided the program counter is less than 32768. When $\overline{EA}$ is held at a TTL LOW level during RESET, the CPU executes out of external program memory via Port 0 and Port 2. $\overline{EA}$ is not allowed to float. |
| P0.0–0.7 (AD0 to AD7) | 32–39 | 36–43 | I/O | **Port 0:** 8-bit open drain bidirectional I/O Port. It is also the multiplexed low-order address and data bus during accesses to external memory (AD0 to AD7). During these accesses internal pull-ups are activated. Port 0 can sink/source 8 LS TTL inputs. |
| $V_{DD}$ | 40 | 44 | I | **Power Supply:** +5V power supply pin during normal operation, Idle mode and Power-down mode. |
| | | | | To avoid a "latch-up" effect at power-on, the voltage on any pin (at any time) must not be higher than $V_{DD}$ +0.5V or lower than $V_{SS}$ –0.5V respectively. |
| NC | – | 1, 12, 23, 34 | – | No connection (PLCC only). |

**Table 1. 8XC524/8XC528 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes): | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | ES1 | ET2 | ES0 | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP*# | Interrupt priority | B8H | – | PS1 | PT2 | PS0 | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | SDA | SCL | – | – | – | – | T2EX | T2 | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | RD | WR | T1 | T0 | INT1 | INT0 | TxD | RxD | FFH |
| PCON | Power control | 87H | SMOD | – | – | – | GF1 | GF0 | PD | IDL | 0xxx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | F1 | P | 00H |
| RCAP2H# | Capture high | CBH | | | | | | | | | 00H |
| RCAP2L# | Capture low | CAH | | | | | | | | | 00H |
| SBUF | Serial data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial controller | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| S1BIT# | Serial $I^2C$ data | D9H/RD | SDI | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x0000000B |
| | | WR | SD0 | X | X | X | X | X | X | X | 0xxxxxxxB |
| S1INT# | Serial $I^2C$ interrupt | DAH | INT | X | X | X | X | X | X | X | 0xxxxxxxB |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| S1SCS*# | Serial $I^2C$ control | D8H/RD | SDI | SCI | CLH | BB | RBF | WBF | STR | ENS | xxxx0000B |
| | | WR | SD0 | SC0 | CLH | X | X | X | STR | ENS | 00xxxx00B |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| T2CON*# | Timer 2 control | C8H | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/T2 | CP/RL2 | 00H |
| TH0 | Timer high 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer high 1 | 8DH | | | | | | | | | 00H |
| TH2# | Timer high 2 | CDH | | | | | | | | | 00H |
| TL0 | Timer low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer low 1 | 8BH | | | | | | | | | 00H |
| TL2# | Timer low 2 | CCH | | | | | | | | | 00H |
| T3# | Watchdog timer | FFH | | | | | | | | | 00H |
| TMOD | Timer mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |
| WDCON# | Watchdog control | A5H | | | | | | | | | A5H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

**Table 2. Internal and External Program Memory Access with Security Bit Set**

| INSTRUCTION | ACCESS TO INTERNAL PROGRAM MEMORY | ACCESS TO EXTERNAL PROGRAM MEMORY |
|---|---|---|
| MOVC in internal program memory | YES | YES |
| MOVC in external program memory | NO | YES |

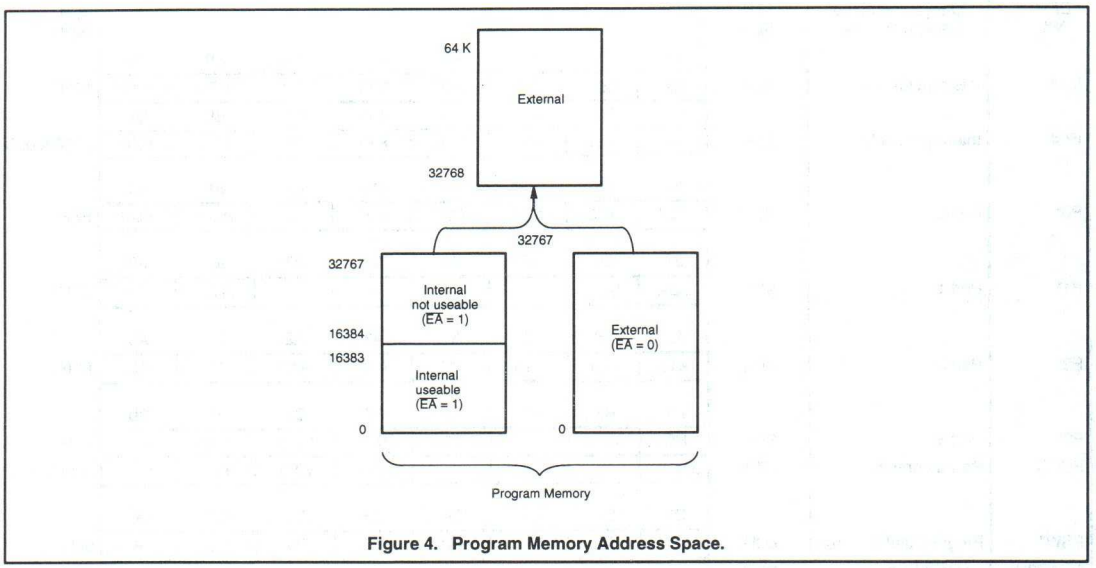



**Figure 4. Program Memory Address Space.**

## PROGRAM MEMORY

The program memory address space of the P83C524 comprises an internal and an external memory portion. The P83C524 has 16 kbyte of usable program memory on–chip. The program memory can be externally expanded up to 64 kbyte. If the $\overline{EA}$ pin is held HIGH during RESET, the P83C524 executes out of the internal program memory unless the address exceeds 32767 (notice: Only address 0 to 16383 of the internal program memory can be used). Locations 32768 through 64K are then fetched from the external program memory. If the $\overline{EA}$ pin is held LOW during RESET, the P83C524 fetches all instructions from the external program memory. Figure 4 illustrates the program memory address space.

## ROM CODE PROTECTION

By setting a mask programmable security bit, the ROM content in the 83C524 is protected, i.e., it cannot be read out by any test mode or by any instruction in the external program memory space. The MOVC instructions are the only ones which have access to program code in the internal or external program memory. The $\overline{EA}$ input is latched during RESET and is 'don't care' after RESET (also if security bit is not set). This implementation prevents reading from internal program code by switching from external program memory to internal program memory during MOVC instruction or an instruction that handles immediate data. Table 2 lists the access to the internal and external program memory by the MOVC instructions when the security bit has been set to logical one. If the security bit has been set to a logical 0 there are no restrictions for the MOVC instructions.

## INTERNAL DATA MEMORY

The internal data memory is divided into three physically separated segments: 256 bytes of RAM, 256 bytes of AUX-RAM, and a 128 bytes special function area. These can be addressed each in a different way.

- RAM 0 to 127 can be addressed directly and indirectly as in the 80C51. Address pointers are R0 and R1 of the selected register bank.
- RAM 128 to 255 can only be addressed indirectly as in the 80C51. Address pointers are R0 and R1 of the selected register bank.
- AUX-RAM 0 to 255 is indirectly addressed in the same way as external data memory with the MOVX instructions. Address pointers are R0, R1 of the selected register bank and DPTR. An access to AUX-RAM 0 to 255 will not affect ports P0, P2, P3.6 and P3.7.

An access to external data memory locations higher than 255 will be performed with the MOVX DPTR instructions in the same way as in the 8051 structure, so with P0 and P2 as data/address bus and P3.6 and P3.7 as write and read timing signals. Note that these external data memory cannot be accessed with R0 and R1 as address pointer.

## TIMER 2

Timer 2 is functionally equal to the Timer 2 of the 8052AH. Timer 2 is a 16-bit timer/counter. These 16 bits are formed by two special function registers TL2 and TH2. Another pair of special function register RCAP2L and RCAP2H form a 16-bit capture register or a 16-bit reload register. Like Timer 0 and 1, it can operate either as a timer or as an event counter. This is selected by bit C/T2N in the special function register T2CON. It has three operating modes: capture, autoload, and baud rate generator mode which are selected by bits in T2CON.

## WATCHDOG TIMER T3

The watchdog timer (WDT) (see Figure 5) consists of an 11-bit prescaler and an 8-bit timer formed by special function register T3. The prescaler is incremented by an on-chip oscillator with a fixed frequency of 1MHz. The maximum tolerance on this frequency is −50% and +100%. The 8-bit timer increments every 2048 cycles of the on-chip oscillator. When a timer overflow occurs, the microcontroller is reset and a reset output pulse of 16 × 2048 cycles of the on-chip oscillator is generated at pin RST. The internal RESET signal is not inhibited when the external RST pin is kept LOW by, for example, an external reset circuit. The RESET signal drives Ports 1, 2 and 3 outputs into the HIGH state and port 0 into high impedance, whether the XTAL-clock is running or not.

The watchdog timer is controlled by special function register WDCON with the direct address location A5H. WDCON can be read and written by software. A value of A5H in WDCON halts the on-chip oscillator and clears both the prescaler and timer T3. After the RESET signal, WDCON contains A5H.

Every value other than A5H in WDCON enables the watchdog timer. When the watchdog timer is enabled, it runs independently of the XTAL-clock.

Timer T3 can be read on the fly. Timer T3 can be written to only if WDCON has previously been loaded with 5AH, otherwise T# and the prescaler are not affected. A successful write operation to T3 also clears the prescaler and WDCON. During a read or write operation addressing T3, the output of the on-chip oscillator is inhibited to prevent timing problems due to asynchronous increments of T3. To prevent an overflow of the watchdog timer, the user program has to reload T3 within periods that are shorter than the programmed watchdog timer internal. This time interval is determined by the 8-bit reload value that is written into register T3.

Watchdog timer interval =

$$\frac{[256 - (T3)] \times 2048}{\text{on - chip oscillator frequency}}$$

The advantages of this implementation are:

- Only an internal reset connection to the microcontroller core.
- The Power-down mode and the Watchdog (WDT) function can be used concurrently.
- The WDT also monitors the XTAL oscillator. In case of a failure the port outputs are forced to a defined HIGH state.
- Interference will not disable the WDT because it is unlikely that it will force WDCON to A5H.
- Tolerances of the on-chip oscillator can be adjusted by testing the T3 value and adapting the reload value.
- The WDT can be enabled and disabled under control of the user software. This gives the possibility to use both the Watchdog function and the Power-down function.
- The direct address A5H of WDCON and its disable value A5H will not unintentionally be present at a random location in the field of program code, except for immediate data, because the opcode A5H is not used in the instruction set.

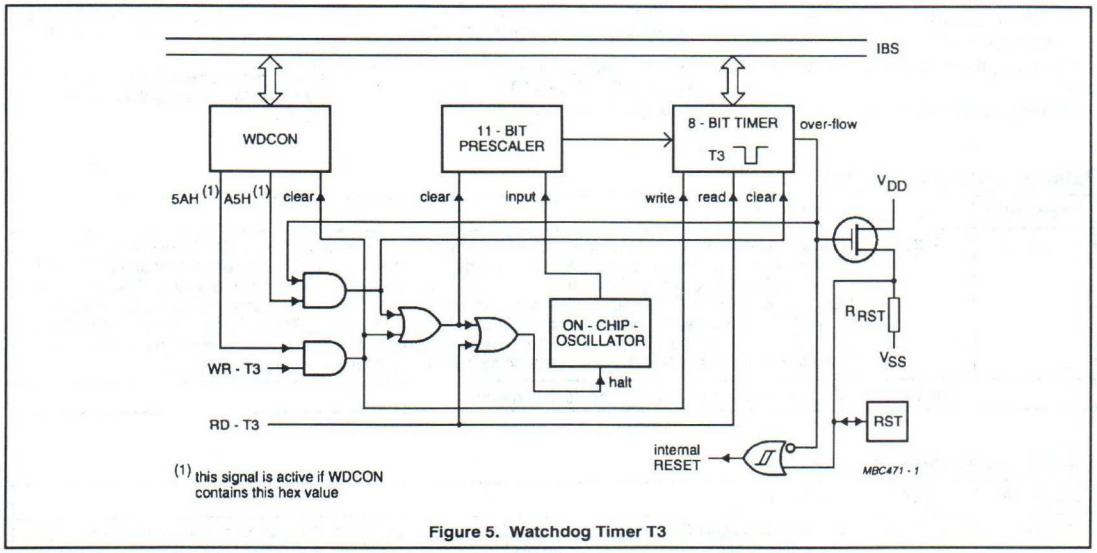



**Figure 5. Watchdog Timer T3**

## BIT-LEVEL I²C INTERFACE

This bit-level serial I/O interface supports the I²C-bus (see Figure 6). P1.6/SCL and P1.7/SDA are the serial I/O pins. These two pins meet the I²C specification concerning the input levels and output drive capability. Consequently, these pins have an open drain output configuration. All the four modes of the I²C-bus are supported:

- master transmitter
- master receiver
- slave transmitter
- slave receiver

The advantages of the bit-level I²C hardware compared with a full software I²C implementation are:

- The hardware can generate the SCL pulse
- Testing a single bit (RBF respectively, WBF) is sufficient as a check for error free transmission.

The bit-level I²C hardware operates on serial bit level and performs the following functions:

- filtering the incoming serial data and clock signals
- recognizing the START condition
- generating a serial interrupt request SI after reception of a START condition and the first falling edge of the serial clock
- recognizing the STOP condition
- recognizing a serial clock pulse on the SCL line
- latching a serial bit on the SDA line (SDI)
- stretching the SCL LOW period of the serial clock to suspend the transfer of the next serial data bit
- setting Read Bit Finished (RBF) when the SCL clock pulse has finished and Write Bit Finished (WBF) if there is no arbitration loss detected (i.e., SDA = 0 while SDO = 1)
- setting a serial clock Low-to-High detected (CLH) flag
- setting a Bus Busy (BB) flag on a START condition and clearing this flag on a STOP condition
- releasing the SCL line and clearing the CLH, RBF and WBF flags to resume transfer of the next serial data bit
- generating an automatic clock if the single bit data register S1BIT is used in master mode.

The following functions must be done in software:

- handling the I²C START interrupts
- converting serial to parallel data when receiving
- converting parallel to serial data when transmitting
- comparing the received slave address with its own
- interpreting the acknowledge information
- guarding the I²C status if RBF or WBF = 0.

Additionally, if acting as master:

- generating START and STOP conditions
- handling bus arbitration
- generating serial clock pulses if S1BIT is not used.

Three SFRs control the bit-level I²C interface: S1INT, S1BIT and S1SCS.

### S1INT: I²C Interrupt Register

This register is located at address DAH. Refer to Table 3.

**S1INT SFR (DAH)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SI | X | X | X | X | X | X | X |

**NOTE:**

1. **SI bit**: Writing a logic 0 clears this bit, writing a logic 1 has no effect.

### S1BIT: Single-bit Data Register with I²C Auto-Clock

This register is located at address D9H. Refer to Table 4.

**S1BIT SFR (D9H)**

**READ**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SDI | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**WRITE**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SDO | X | X | X | X | X | X | X |

**NOTE:**

1. Access of the **S1BIT** SFR clears SI, CLH, RBF and WBF. It starts the auto-clock if SCO = 0.

**Table 3. Description of S1INT Bits**

| MNEMONIC | BIT | FUNCTION |
|---|---|---|
| SI | S1INT.7 | **Serial Interrupt request (SI) flag:** If a START condition occurs the SI flag in the S1INT SFR is set on the falling edge of the filtered serial clock. If SI = 1 is detected during a transfer this can be a "spurious START" error condition. If no transfer is taking place the SI = 1 is a START from an external master. Provided the bits EA and ES1 in IE SFR are set, SI then generates an interrupt so that a slave address receive routine can be started. SI can be cleared by accessing the S1BIT register or by writing "00" to S1INT. Also after reception of a START condition, the LOW period of the clock pulse is stretched, suspending the serial transfer to allow the software to take action. This clock stretching is ended by a read or write access to S1BIT. |
| – | S1INT.6 to 0 | X = undefined during read, don't care during write. |

**Table 4. Description of S1BIT Bits**

| MNEMONIC | BIT | FUNCTION |
|---|---|---|
| SDO/SDI | S1BIT.7 | **Serial Data Output (SDO) and the filtered Serial Data Input (SDI).** SDI data is latched on the rising edge of the filtered serial clock. S1BIT.7 accesses the same memory locations as S1SCS.7. S1BIT SFR is not bit-addressable. |
| – | S1BIT.6 to 0 | X = don't care. |

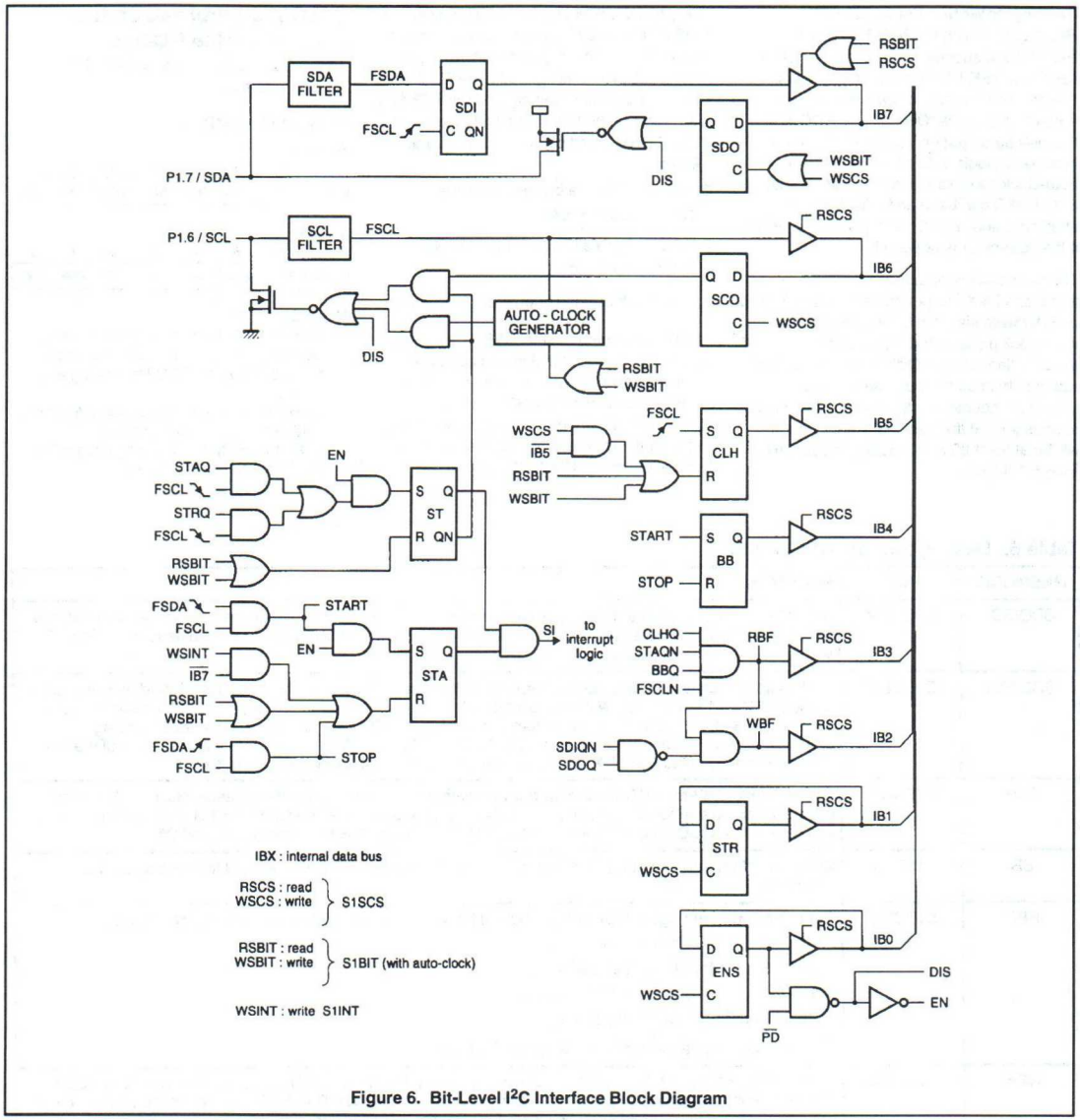



**Figure 6. Bit-Level I²C Interface Block Diagram**

**Reading or Writing the S1BIT SFR**

Reading or writing the S1BIT SFR starts an $I^2C$ bit I/O sequence: some flags are cleared (SI, CLH, RBF, WBF), clock stretching is finished and the auto-clock is started. An auto-clock pulse is "OR-ed" with SCO and thus will be output only if the SCO flag has been set to logic 0. SCO = 1 inhibits the auto-clock start, so a dummy read or write of S1BIT SFR can be used to finish clock stretching and clear SI, SLH, RBF and WBF if the auto-clock is not used.

The auto-clock is an active HIGH SCL pulse that starts 28 XTAL periods after the SDI read or SDO write via S1BIT. The duration of the auto-clock pulse is 100 XTAL clock periods. If the SCL line is kept LOW by any device that wants to hold up the bus transfer, the auto-clock counter waits after 20 XTAL clock periods so that the auto-clock pulse length will be at least 80 XTAL clock periods (5µs at $f_{OSC}$ = 16MHz).

Every bit I/O should be followed by a RBF or WBF bit test. A bit transfer has been finished successfully if after reading a bit the RBF flag is logic 1 or after writing a bit the WBF flag is logic 1. When after reading a bit the RBF flag is still logic 0, the bus status just before the S1SCS status read can be determined as follows:

- If CLH = 0 then a bus device is still stretching the clock.
- If SCI = 1 while CLH = 1 then the SCL pulse is not finished.
- If BB = 0 there has been a STOP condition.

When after writing a bit the WBF flag is still logic 0 and none of the 3 status conditions mentioned for RBF are found then a "bus arbitration lost" condition will be the cause. This can be determined also from the states of the received bit and the last transmitted bit: "arbitration loss" if SDO = 1 and SDI = 0.

## S1SCS: Control and Status Register for the $I^2C$-bus

This register is located at address D8H. Refer to Table 5.

**S1SCS SFR (D8H)**

**READ**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SDI | SCI | CLH | BB | RBF | WBF | STR | ENS |

**WRITE**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SDO | SCO | SLH | X | X | X | STR | ENS |

**NOTES:**

1. **SDI and SCI bits:** read-modify-write operations like "SETB bit" or "CLR bit" access SDO and SCO for reading and writing.
2. **CLH bit:** writing a logic 0 clears this bit, writing a logic 1 has no effect.
3. **RBF and WBF bits:** writing a logic 0 to CLH also clears these bits.
4. X = don't care.

## Table 5. Description of S1SCS Bits

| MNEMONIC | BIT | FUNCTION |
|---|---|---|
| SDO/SDI | S1SCS.7 | **Serial Data Output and the filtered Serial Data Input.** SDI data is latched on the rising edge of the filtered serial clock. S1SCS.7 accesses the same memory locations as S1BIT.7. Access of the data bit via S1SCS will not start an auto-clock pulse. |
| SCO/SCI | S1SCS.6 | **Serial Clock Output and the filtered Serial Clock Input.** Serial clock output SCO is "OR-ed" with the auto-clock. If SCO = 1 the auto-clock output is inhibited. The internal clock stretching logic and external devices can pull the SCL line LOW. If the auto-clock is not used, the SCL line has to be controlled by setting SCO = 1, waiting for CLH = 1 and setting SCO = 0 after the specified SCL HIGH time. (Because of the input filter, CLH will be set at least 8 XTAL clock periods after the SCL LOW-to-HIGH transition. |
| CLH | S1SCS.5 | **Serial Clock LOW-to-HIGH transition flag:** set with a rising edge of the filtered serial clock. CLH = 1 indicates that, since the last CLH reset, a new valid data bit has been latched in SDI. CLH can be reset by writing a logic 0 to S1SCS.5 or by a read/write of S1BIT. Clearing CLH also clears RBF and WBF. |
| BB | S1SCS.4 | **Bus Busy flag:** indicating that there has been a START condition that was not yet followed by a STOP condition. |
| RBF | S1SCS.3 | **Read Bit Finished flag:** indicating a successful bit read. RBF = 1 implies the following conditions:<br>– CLH = 1: SCL had a rising edge<br>– SCI = 0: the SCL pulse has finished<br>– SI = 0: no START condition occurred<br>– BB = 1: no STOP condition occurred<br>The RBF flag can be cleared by clearing the CLH flag. |
| WBF | S1SCS.2 | **Write Bit Finished flag:** indicating a successful bit write. the same conditions as for RBF are true and also no "arbitration loss" condition occurred. Arbitration is lost if a 1 data bit in SDO was over-ruled on SDA by an external device. The WBF flag can be cleared by clearing the CLH flag. |
| STR | S1SCS.1 | **STRetch control flag.** STR = 1 enabled stretching of all SCL LOW periods. This allows the processor in $I^2C$ slave mode to react on a fast master. The STR flag remains set until cleared by writing a logic 0 to S1SCS.1.<br>**The STretch (ST)** flag (not readable) pulls the serial clock LOW while ST = 1. The ST flag is set on the falling edge of the filtered serial clock if STR = 1. It is also set after reception of a START condition, regardless of the STR contents. ST is cleared with a read or write of S1BIT. |
| ENS | S1SCS.0 | **ENable Serial I/O flag.** ENS = 1 enables the START detection and clock stretching logic. ENS = 0 can be used to switch off the $I^2C$ bus hardware. Note that the SDO and SCO control flags must be set to logic 1 before ENS is set to avoid pulling SCL or SDA lines to logic 0. |

## INTERRUPT SYSTEM

The interrupt structure of the 8XC524 is the same as that used in the 80C51, but includes two additional interrupt sources: one for the third timer/counter, T2, and one for the $I^2C$ interface. The interrupt enable and interrupt priority registers are IE and IP.

### IE: Interrupt Enable Register

This register is located at address A8H. Refer to Table 6.

**IE SFR (A8H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| EA | ES1 | ET2 | ES | ET1 | EX1 | ET0 | EX0 |

### IP: Interrupt Priority Register

This register is located at address B8H. Refer to Table 7.

**IP SFR (B8H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | PS1 | PT2 | PS | PT1 | PX1 | PT0 | PX0 |

**Table 6. Description of IE Bits**

| MNEMONIC | BIT | FUNCTION |
|---|---|---|
| EA | IE.7 | General enable/disable control:<br>0 = **NO** interrupt is enabled.<br>1 = **ANY** individually enabled interrupt will be accepted. |
| ES1 | IE.6 | Enable **bit-level $I_2C$ I/O** interrupt |
| ET2 | IE.5 | Enable **Timer 2** interrupt |
| ES | IE.4 | Enable **Serial Port** interrupt |
| ET1 | IE.3 | Enable **Timer 1** interrupt |
| EX1 | IE.2 | Enable **External** interrupt 1 |
| ET0 | IE.1 | Enable **Timer 0** interrupt |
| EX0 | IE.0 | Enable **External** interrupt 0 |

**Table 7. Description of IP Bits**

| MNEMONIC | BIT | FUNCTION |
|---|---|---|
| – | IP.7 | **Reserved.** |
| PS1 | IP.6 | **Bit-level $I^2C$** interrupt priority level |
| PT2 | IP.5 | **Timer 2** interrupt priority level |
| PS | IP.4 | **Serial Port** interrupt priority level |
| PT1 | IP.3 | **Timer 1** interrupt priority level |
| PX1 | IP.2 | **External Interrupt 1** priority level |
| PT0 | IP.1 | **Timer 0** interrupt priority level |
| PX0 | IP.0 | **External Interrupt 0** priority level |

The interrupt vector locations and the interrupt priorities are:

| Source Vector | Address | Priority within Level |
|---|---|---|
| 0003H | IE0 | Highest |
| 002BH | TF2+EXF2 | |
| 0053H | SI ($I^2C$) | |
| 000BH | TF0 | |
| 0013H | IE1 | |
| 001BH | TF1 | |
| 0023H | R1+T1 | Lowest |

## IDLE AND POWER-DOWN OPERATION

Idle mode operation permits the interrupt, serial ports and timer blocks to function while the CPU is halted. The following functions remain active during Idle mode. These functions may generate an interrupt or reset and thus end the Idle mode:

- Timer 0, Timer 1, Timer 2, Watchdog Timer
- UART, I²C-Interface
- External interrupt.

The Power-down operation stops the oscillator. The Power-down mode can only be activated by setting the PD bit in the PCON register.

### Idle Mode

The instruction that sets PCON.0 is the last instruction executed in the normal operating mode before Idle mode is activated. Once in the Idle mode, the CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator, RAM and all other registers maintain their data during Idle mode. The status of external pins during Idle mode is shown in Table 9.

### Power-down Mode

The instruction that sets PCON.1 is the last executed prior to going into the Power-down mode. The oscillator is stopped. Note that the Power-down mode also can be entered when the watchdog has been enabled. The Power-down mode can be terminated by an external RESET in the same way as in the 80C51 or in addition by any one of the two external interrupts, IE0 or IE1. A reset generated by the WDT terminates the Power-down mode in the same way as an external RESET.

The status of the external pins during Power-down mode is shown in Table 9. If the Power-down mode is activated while in external program memory, the port data that is held in the P2 SFR is restored to Port 2. If the data is a logic 1, the port pin is held HIGH during the Power-down mode by the strong pull-up transistor p1.

### Wake-up from Power-down Mode

The Power-down mode of the P83C524 can also be terminated by any one of the two external interrupts, IE0 or IE1. A termination with an external interrupt does not affect the internal data memory and does not affect the Special Function Registers (SFRs).

In order to prevent any interrupt priority problems during wake-up, the prioirty of the desired wake-up interrupt should be higher than the priorities of all other enabled interrupt sources. The instruction following the one that put the device into the Power-down mode will be the first one which will be executed after an interrupt has been serviced.

**Table 8. Internal Registers Status after a RESET**

| REGISTER | CONTENTS |
|---|---|
| ACC | 00H |
| B | 00H |
| DPH, DPL | 00H |
| IE | 00H |
| IP | X000 0000B |
| PCH, PCL | 00H |
| PCON | 0XX 0000B |
| PSW | 00H |
| P0 to P3 | FFH |
| SBUF | Indeterminate |
| SCON | 00H |
| SP | 07H |
| TCON | 00H |
| TMOD | 00H |
| TH0, TL0 | 00H |
| TH1, TL1 | 00H |
| T2CON | 00H |
| TH2, TL2 | 00H |
| RCAP2H, RCAP2L | 00H |
| S1BIT | X000 0000B |
| S1INT | 0XXX XXXXB |
| S1SCS | XXX0 0000B |
| T3 | 00H |
| WDCON | A5H |

**Table 9. Status of the External Pins During Idle and Power-down Modes**

| MODE | MEMORY | ALE | PSEN | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | internal | 1 | 1 | port data | port data | port data | port data |
| Idle | external | 1 | 1 | floating | port data | address | port data |
| Power-down | internal | 0 | 0 | port data | port data | port data | port data |
| Power-down | external | 0 | 0 | floating | port data | port data | port data |

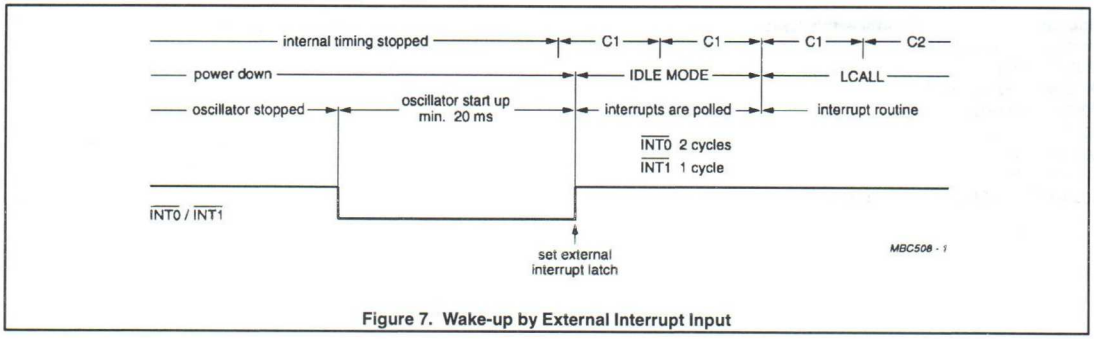



**Figure 7. Wake-up by External Interrupt Input**

## ABSOLUTE MAXIMUM RATINGS

In accordance with the Absolute Maximum Rating System (IEC 134).

| SYMBOL | PARAMETER | MIN. | MAX. | UNIT |
|---|---|---|---|---|
| $V_{DD}$ | Supply voltage range | –0.5 | +6.5 | V |
| $V_I$ | All Input voltages | –0.5 | $V_{DD}$+0.5 | V |
| $P_{TOT}$ | Total power dissipation | – | 1 | W |
| $T_{STG}$ | Storage temperature range | –65 | +150 | °C |
| $T_{AMB}$ | Operating ambient temperature range: | | | |
| | version FBx | 0 | +70 | °C |
| | version FFx | –40 | +85 | °C |
| | version FHx | –40 | +125 | °C |

## DC CHARACTERISTICS FBx (0 to +70°C)

$V_{DD}$ = 5V ±20%; $V_{SS}$ = 0V; $T_{AMB}$ = 0 to +70°C. All voltages with respect to $V_{SS}$ unless otherwise specified.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| **Supply** | | | | | |
| $V_{DD}$ | Supply voltage range | | 4.0 | 6.0 | V |
| | Supply current: | | | | |
| $I_{DD}$ | Operating modes | $V_{DD}$ = 6V; $f_{CLK}$ = 16MHz (notes 1 and 8) | – | 35 | mA |
| $I_{ID}$ | Idle mode | $V_{DD}$ = 5V ±20%; $f_{CLK}$= 16MHz (notes 2 and 8) | – | 6 | mA |
| $I_{PD}$ | Power-down mode | $2 \le V_{PD} \le V_{DDMAX}$ (note 3) | – | 100 | µA |
| **Inputs** | | | | | |
| $V_{IL}$ | LOW level input voltage (except $\overline{EA}$, P1.6, P1.7) | | –0.5 | $0.2V_{DD}$–0.1 | V |
| $V_{IL1}$ | LOW level input voltage $\overline{EA}$ | | –0.5 | $0.2V_{DD}$–0.3 | V |
| $V_{IL2}$ | LOW level input voltage P1.6, P1.7 | note 6 | –0.5 | $0.3V_{DD}$ | V |
| $V_{IH}$ | HIGH level input voltage (except RST, XTAL1, P1.6, P1.7) | | $0.2V_{DD}$+0.9 | $V_{DD}$+0.5 | V |
| $V_{IH1}$ | HIGH level input voltage RST, XTAL1 | | $0.7V_{DD}$ | $V_{DD}$+0.5 | V |
| $V_{IH2}$ | HIGH level input voltage P1.6, P1.7 | note 6 | $0.7V_{DD}$ | 6.0 | V |
| $I_{IL}$ | Input current logic 0 Ports 1, 2, and 3 (except P1.6 and P1.7) | $V_I$ = 0.45V | – | –50 | µA |
| $I_{TL}$ | Input current logic 1 to logic 0 transition Ports 1, 2, and 3 (except P1.6 and P1.7) | $V_I$ = 2.0V | – | –650 | µA |
| $I_{LI1}$ | Input leakage current Port 0, $\overline{EA}$ | $0.45 < V_I < V_{DD}$ | – | ±10 | µA |
| $I_{LI2}$ | Input leakage current P1.6 and P1.7 | $0V < V_I < 6V$<br>$0V < V_{DD} < 6V$ | – | ±10 | µA |

## DC CHARACTERISTICS FBx (0 to +70°C) (Continued)

$V_{DD}$ = 5V ±20%; $V_{SS}$ = 0V; $T_{AMB}$ = 0 to +70°C. All voltages with respect to $V_{SS}$ unless otherwise specified.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| **Outputs** | | | | | |
| $V_{OL}$ | Low level output voltage<br>Ports 1, 2, and 3 (except P1.6 and P1.7) | $I_{OL}$ = 1.6mA<br>notes 4 and 7 | – | 0.45 | V |
| $V_{OL1}$ | LOW level output voltage<br>Port 0, ALE, $\overline{PSEN}$ | $I_{OL}$ = 3.2mA<br>notes 4 and 7 | – | 0.45 | V |
| $V_{OL2}$ | LOW level output voltage<br>P1.6 and P1.7 | $I_{OL}$ = 3.0mA<br>note 7 | – | 0.40 | V |
| $V_{OH}$ | HIGH level output voltage<br>Ports 1, 2, and 3 | $I_{OH}$ = –60µA; $V_{DD}$ = 5V ±10%<br>$I_{OH}$ = –25µA<br>$I_{OH}$ = –10µA | 2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | –<br>–<br>– | V<br>V<br>V |
| $V_{OH1}$ | HIGH level output voltage<br>Port 0 in external Bus mode,<br>ALE, $\overline{PSEN}$, RST | $I_{OH}$ = –800µA; $V_{DD}$ = 5V ±10%<br>$I_{OH}$ = –300µA<br>$I_{OH}$ = –80µA<br>note 5 | 2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | –<br>–<br>– | V<br>V<br>V |
| $R_{RST}$ | RST pull-down resistor | | 50 | 150 | kΩ |
| $C_{I/O}$ | Capacitance of input buffer | Test frequency = 1MHz; $T_{AMB}$ = 25°C | – | 10 | pF |

**NOTES TO DC CHARACTERISTICS (FBx):**

1. The operating supply current is measures with all output pins disconnected; XTAL1 driven with $t_R = t_F$ = 5ns; $V_{IL} = V_{SS}$ + 0.5V; $V_{IH} = V_{DD}$ – 0.5V; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 + P1.6 + P1.7 = $V_{DD}$; the WDT is disabled (by the external RESET).
2. The Idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_R = t_F$ = 5ns; $V_{IL} = V_{SS}$ + 0.5V; $V_{IH} = V_{DD}$ – 0.5V; XTAL2 not connected; the WDT is disabled; $\overline{EA}$ = RST = $V_{SS}$; Port 0 = P1.6 = P1.7 = $V_{DD}$.
3. The Power-down current is measured with all output pins disconnected; XTAL2 not connected; WDT is disabled; $\overline{EA}$ = RST = XTAL1 = $V_{SS}$; Port 0 = P1.6 = P1.7 = $V_{DD}$.
4. Capacitive loading on Port 0 and Port 2 may cause spurious noise pulses to be superimposed on the LOW level output voltage of ALE, Port 1 and Port 3. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make a 1-to-0 transition during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE line may exceed 0.8V. In such cases it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.
5. Capacitive loading on Port 0 and Port 2 may cause the HIGH level output voltage on ALE $\overline{PSEN}$ to momentarily fall below the 0.9V $V_{DD}$ specification when the address bits are stabilizing.
6. The input threshold voltage of P1.6 and P1.7 (SIO1) meets the $I^2C$ specification, so a voltage below 0.3 $V_{DD}$ will be recognized as a logic 0 while an input above 0.7 $V_{DD}$ will be recognized as a logic 1.
7. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 10mA
   Maximum $I_{OL}$ per 8-bit port:
   Port 0: 26mA
   Ports 1, 2, and 3: 15mA
   Maximum total $I_{OL}$ for all output pins: 71mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
8. $I_{DDMAX}$ at other frequencies can be derived from Figure 8, where f is the external oscillator frequency in MHz; $I_{DDMAX}$ is given in mA.

## DC CHARACTERISTICS FFx (–40 to +85°C)

$V_{DD}$ = 5V ±20%; $V_{SS}$ = 0V; $T_{AMB}$ = –40 to +85°C (extended temperature range). All voltages with respect to $V_{SS}$ unless otherwise specified. DC parameters not included here are the same as for the FBx temperature range data.

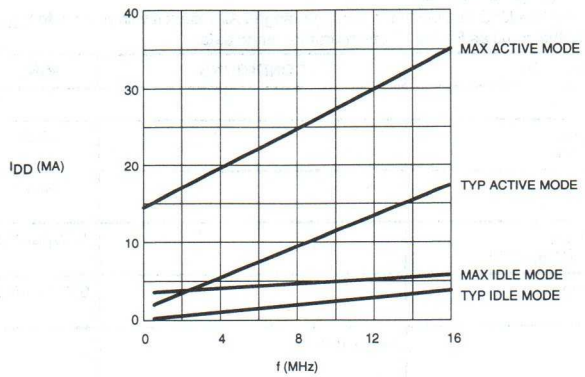
| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| **Inputs** | | | | | |
| $V_{IL}$ | LOW level input voltage (except $\overline{EA}$, P1.6, P1.7) | | –0.5 | $0.2V_{DD}$–0.15 | V |
| $V_{IL1}$ | LOW level input voltage $\overline{EA}$ | | –0.5 | $0.2V_{DD}$–0.35 | V |
| $V_{IH}$ | HIGH level input voltage (except RST, XTAL1, P1.6, P1.7) | | $0.2V_{DD}$+1.0 | $V_{DD}$+0.5 | V |
| $V_{IH1}$ | HIGH level input voltage RST, XTAL1 | | $0.7V_{DD}$ +0.1 | $V_{DD}$+0.5 | V |
| $I_{IL}$ | Input current logic 0 Ports 1, 2, and 3 (except P1.6 and P1.7) | $V_I$ = 0.45V | – | –75 | µA |
| $I_{TL}$ | Input current logic 1 to logic 0 transition Ports 1, 2, and 3 (except P1.6 and P1.7) | $V_I$ = 2.0V | – | –750 | µA |

## DC CHARACTERISTICS FHx (–40 to +125°C)

$V_{DD}$ = 5V ±10%; $V_{SS}$ = 0V; $T_{AMB}$ = –40 to +125°C (extended temperature range). All voltages with respect to $V_{SS}$ unless otherwise specified. DC parameters not included here are the same as for the FBx temperature range data.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| **Supply** | | | | | |
| $V_{DD}$ | Supply voltage range | | 4.5 | 5.5 | V |
| | Supply current: | | | | |
| $I_{DD}$ | Operating modes | $V_{DD}$ = 5.5V; $f_{CLK}$ = 16MHz (note 1) | – | 35 | mA |
| $I_{ID}$ | Idle mode | $V_{DD}$ = 5V ±10%; $f_{CLK}$= 16MHz (note 2) | – | 6 | mA |
| $I_{PD}$ | Power-down mode | $2 \le V_{PD} \le V_{DDMAX}$ (note 3) | – | 150 | µA |
| **Inputs** | | | | | |
| $V_{IL}$ | LOW level input voltage (except $\overline{EA}$, P1.6, P1.7) | | –0.5 | $0.2V_{DD}$–0.25 | V |
| $V_{IL1}$ | LOW level input voltage $\overline{EA}$ | | –0.5 | $0.2V_{DD}$–0.45 | V |
| $V_{IH}$ | HIGH level input voltage (except RST, XTAL1, P1.6, P1.7) | | $0.2V_{DD}$+1.0 | $V_{DD}$+0.5 | V |
| $V_{IH1}$ | HIGH level input voltage RST, XTAL1 | | $0.7V_{DD}$ +0.1 | $V_{DD}$+0.5 | V |
| $I_{IL}$ | Input current LOW Ports 1, 2, and 3 (except P1.6 and P1.7) | $V_I$ = 0.45V | – | –75 | µA |
| $I_{TL}$ | Input current logic 1 to logic 0 transition Ports 1, 2, and 3 (except P1.6 and P1.7) | $V_I$ = 2.0V | – | –750 | µA |

**NOTES:**
1. See notes 1 and 8 of the FBx DC characteristics table.
2. See notes 2 and 8 of the FBx DC characteristics table.
3. See note 3 of the FBx DC characteristics table.




Valid only within frequency specifications of the device under test.

**Figure 8. $I_{DD}$ as a Function of Frequency**

## AC CHARACTERISTICS

FBx: $V_{DD}$ = 5V ±20%; $V_{SS}$ = 0V; $T_{AMB}$ = 0 to +70°C; $t_{CK}$ min. = 63ns.
FFx: $V_{DD}$ = 5V ±20%; $V_{SS}$ = 0V; $T_{AMB}$ = –40 to +85°C; $t_{CK}$ min. = 63ns.
FHx: $V_{DD}$ = 5V ±10%; $V_{SS}$ = 0V; $T_{AMB}$ = 0 to +125°C; $t_{CK}$ min. = 63ns.
All versions Fxx: $C_L$ = 100pF for Port 0, ALE and $\overline{PSEN}$; $C_L$ = 80pF for all other outputs unless otherwise specified.
$t_{CK}$ min. = 1/f max. (maximum operating frequency); $t_{CK}$ = clock period.

| SYMBOL | PARAMETER | 16 MHz | | 12 MHz | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|---|
| | | MIN. | MAX. | MIN. | MAX. | MIN. | MAX. | |
| **External program memory** | | | | | | | | |
| $t_{LHLL}$ | ALE pulse duration | 85 | – | 127 | – | 2 $t_{CK}$–40 | – | ns |
| $t_{AVLL}$ | Address set-up time to ALE | 8 | – | 28 | – | $t_{CK}$–55 | – | ns |
| $t_{LLAX}$ | Address hold time after ALE | 28 | – | 48 | – | $t_{CK}$–35 | – | ns |
| $t_{LLIV}$ | Time from ALE to valid instruction input | – | 150 | – | 233 | – | 4 $t_{CK}$–100 | ns |
| $t_{LLPL}$ | Time from ALE to control pulse $\overline{PSEN}$ | 23 | – | 43 | – | $t_{CK}$–40 | – | ns |
| $t_{PLPH}$ | Control pulse duration $\overline{PSEN}$ | 143 | – | 205 | – | 3 $t_{CK}$–45 | – | ns |
| $t_{PLIV}$ | Time from $\overline{PSEN}$ to valid instruction input | – | 83 | – | 145 | – | 3 $t_{CK}$–105 | ns |
| $t_{PXIX}$ | Input instruction hold time after $\overline{PSEN}$ | 0 | – | 0 | – | 0 | – | ns |
| $t_{PXIZ}$ | Input instruction float delay after $\overline{PSEN}$ | – | 38 | – | 59 | – | $t_{CK}$–25 | ns |
| $t_{AVIV}$ | Address to valid instruction input | – | 208 | – | 312 | – | 5 $t_{CK}$–105 | ns |
| $t_{PLAZ}$ | Address float time to $\overline{PSEN}$ | – | 10 | – | 10 | – | 10 | ns |
| **External data memory** | | | | | | | | |
| $t_{LHLL}$ | ALE pulse duration | 85 | – | 127 | – | 2 $t_{CK}$–40 | – | ns |
| $t_{AVLL}$ | Address set-up time to ALE | 8 | – | 28 | – | $t_{CK}$–55 | – | ns |
| $t_{LLAX}$ | Address hold time after ALE | 28 | – | 48 | – | $t_{CK}$–35 | – | ns |
| $t_{RLRH}$ | $\overline{RD}$ pulse duration | 275 | – | 400 | – | 6 $t_{CK}$–100 | – | ns |
| $t_{WLWH}$ | $\overline{WR}$ pulse duration | 275 | – | 400 | – | 6 $t_{CK}$–100 | – | ns |
| $t_{RLDV}$ | $\overline{RD}$ to valid data input | – | 148 | – | 252 | – | 5 $t_{CK}$–165 | ns |
| $t_{RHDX}$ | Data hold time after $\overline{RD}$ | 0 | – | 0 | – | 0 | – | ns |
| $t_{RHDZ}$ | Data float delay after $\overline{RD}$ | – | 55 | – | 97 | – | 2 $t_{CK}$–70 | ns |
| $t_{LLDZ}$ | Time from ALE to valid data input | – | 350 | – | 517 | – | 8 $t_{CK}$–150 | ns |
| $t_{AVDV}$ | Address to valid data input | – | 398 | – | 585 | – | 9 $t_{CK}$–165 | ns |
| $t_{LLWL}$ | Time from ALE to $\overline{RD}$ or $\overline{WR}$ | 138 | 238 | 200 | 300 | 3 $t_{CK}$–50 | 3 $t_{CK}$+50 | ns |
| $t_{AVWL}$ | Time from address to $\overline{RD}$ or $\overline{WR}$ | 120 | – | 203 | – | 4 $t_{CK}$–130 | – | ns |
| $t_{WHWL}$ | Time from $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH | 23 | 103 | 43 | 123 | $t_{CK}$–40 | $t_{CK}$+40 | ns |
| $t_{QVWX}$ | Data valid to $\overline{WR}$ transition | 3 | – | 23 | – | $t_{CK}$–60 | – | ns |
| $t_{QVWH}$ | Data set-up time before $\overline{WR}$ | 288 | – | 433 | – | 7 $t_{CK}$–150 | – | ns |
| $t_{WHQX}$ | Data hold time after $\overline{WR}$ | 13 | – | 33 | – | $t_{CK}$–50 | – | ns |
| $t_{RLAZ}$ | Address float delay after $\overline{RD}$ | – | 0 | – | 0 | – | 0 | ns |

**NOTE:**
1. The maximum operating frequency is limited to 16MHz and the minimum to 1.2MHz (all versions Fxx).

## I²C CHARACTERISTICS (bit-level)

| SYMBOL | PARAMETER | INPUT | OUTPUT | I²C SPEC. | UNIT |
|---|---|---|---|---|---|
| **SCL Timing** | | | | | |
| $t_{HD;STA}$ | START condition hold time | ≥ 14 $t_{CK}$; note 1 | note 2 | ≥ 4.0 | µs |
| $t_{LOW}$ | SCL LOW time | ≥ 16 $t_{CK}$ | note 2 | ≥ 4.7 | µs |
| $t_{HIGH}$ | SCL HIGH time | ≥ 14 $t_{CK}$; note 1 | ≥ 80 $t_{CK}$; note 3 | ≥ 4.0 | µs |
| $t_{RC}$ | SCL RISE time | ≤ 1; note 4 | note 5 | ≤ 1.0 | µs |
| $t_{FC}$ | SCL FALL time | ≤ 0.3; note 4 | ≤ 0.3; note 6 | ≤ 0.3 | µs |
| **SDA Timing** | | | | | |
| $t_{SU;DAT}$ | Data set-up time | ≥ 250 ns | note 2 | ≥ 250 | ns |
| $t_{HD;DAT}$ | Data hold time | ≥ 0 ns | note 2 | ≥ 0 | ns |
| $t_{SU;STA}$ | Repeated START set-up time | ≥ 14 $t_{CK}$; note 1 | note 2 | ≥ 4.7 | µs |
| $t_{SU;STO}$ | STOP condition set-up time | ≥ 14 $t_{CK}$; note 1 | note 2 | ≥ 4.0 | µs |
| $t_{BUF}$ | Bus free time | ≥ 14 $t_{CK}$; note 1 | note 2 | ≥ 4.7 | µs |
| $t_{RD}$ | SDA RISE time | ≤ 1; note 4 | note 5 | ≤ 1.0 | µs |
| $t_{FD}$ | SDA FALL time | ≤ 300ns; note 4 | ≤ 0.3; note 6 | ≤ 0.3 | µs |

**NOTES:**
1. At $f_{CLK}$ = 3.5MHz, this evaluates to 14 × 286ns = 4µs, i.e., the bit-level I²C interface can respond to the I²C protocol for $f_{CLK}$ ≥ 3.5MHz.
2. This parameter is determined by the user software, it has to comply with the I²C specification.
3. This value gives the auto-clock pulse length which meets the I²C specification for the specified XTAL1 clock frequency range. Alternatively, the SCL pulse may be timed by software.
4. Spikes on SDA and SCL lines with a duration of less than 4 × $f_{CLK}$ will be filtered out.
5. The RISE time is determined by the external bus line capacitance and pull-up resistor, it must be ≤ 1 µs.
6. The maximum capacitance on bus lines SDA and SCL is 400 pF.

## XTAL1 CHARACTERISTICS

Oscillator circuitry: crystal capacitors: C1 = C2 = 20pF (see Figure 14).

### Table 10. External Clock Drive XTAL

| SYMBOL | PARAMETER | VARIABLE CLOCK F = 1.2 TO 16 MHz | | UNIT |
|---|---|---|---|---|
| | | MIN. | MAX. | |
| $f_{CLK}$ | Clock frequency | 1.2 | 16 | MHz |
| $t_{CK}$ | Clock period | 63 | 833 | ns |
| $t_{HIGH}$ | HIGH time | 20 | $t_{CK} - t_{LOW}$ | ns |
| $t_{LOW}$ | LOW time | 20 | $t_{CK} - t_{HIGH}$ | ns |
| $t_R$ | RISE time | – | 20 | ns |
| $t_F$ | FALL time | – | 20 | ns |
| $t_{CY}$ | Cycle time ($t_{CY}$ = 12 $t_{CK}$) | 0.75 | 10 | ns |

## SERIAL PORT CHARACTERISTICS

See Table 11 and Figure 15.

### Table 11. External Clock Drive XTAL

$V_{DD}$ = 5V ±20%; $V_{SS}$ = 0V; $T_{AMB}$ = 0 to +70°C; Load Capacitance = 80pF.

| SYMBOL | PARAMETER | 12MHz OSCILLATOR | | VARIABLE OSCILLATOR | | UNIT |
|---|---|---|---|---|---|---|
| | | MIN. | MAX. | MIN. | MAX. | |
| $t_{XLXL}$ | Serial Port clock cycle time | 1 | – | 12 $t_{CK}$ | – | µs |
| $t_{QVXH}$ | Output data setup to clock rising edge | 700 | – | 10 $t_{CK}$–133 | – | ns |
| $t_{XHQX}$ | Output data hold after clock rising edge | 50 | – | 2 $t_{CK}$–117 | – | ns |
| $t_{XHDX}$ | Input data hold after clock rising edge | 0 | – | 0 | – | ns |
| $t_{XHDV}$ | Clock rising edge to input data valid | – | 700 | – | 10 $t_{CK}$–133 | ns |

## TIMING DIAGRAMS

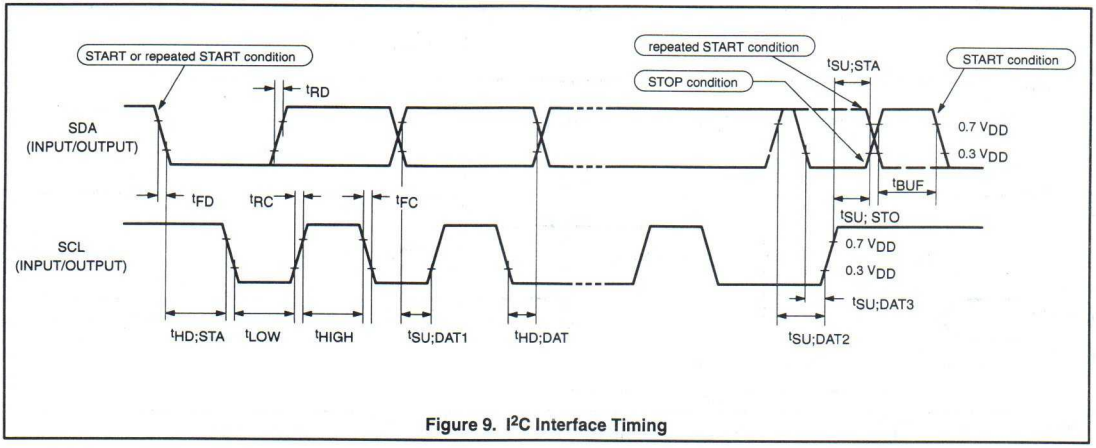



**Figure 9. I²C Interface Timing**

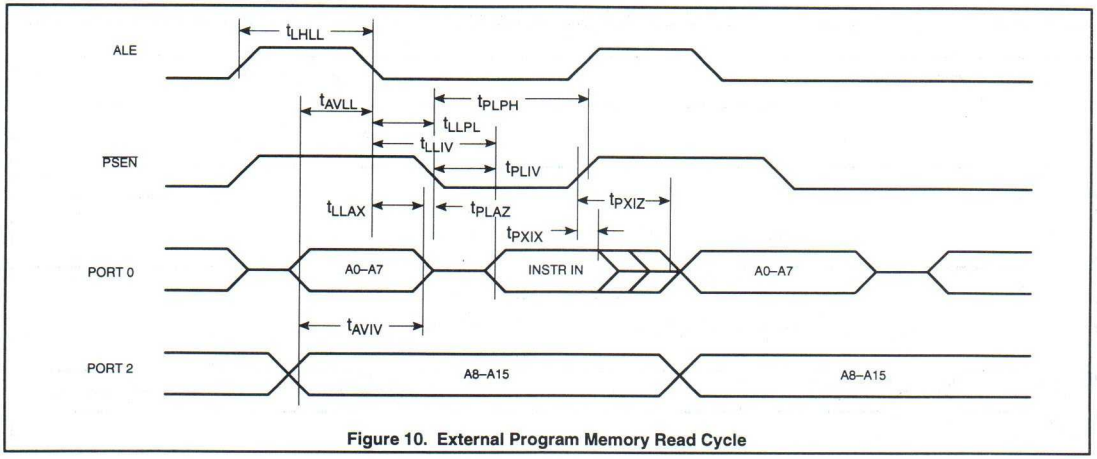



**Figure 10. External Program Memory Read Cycle**

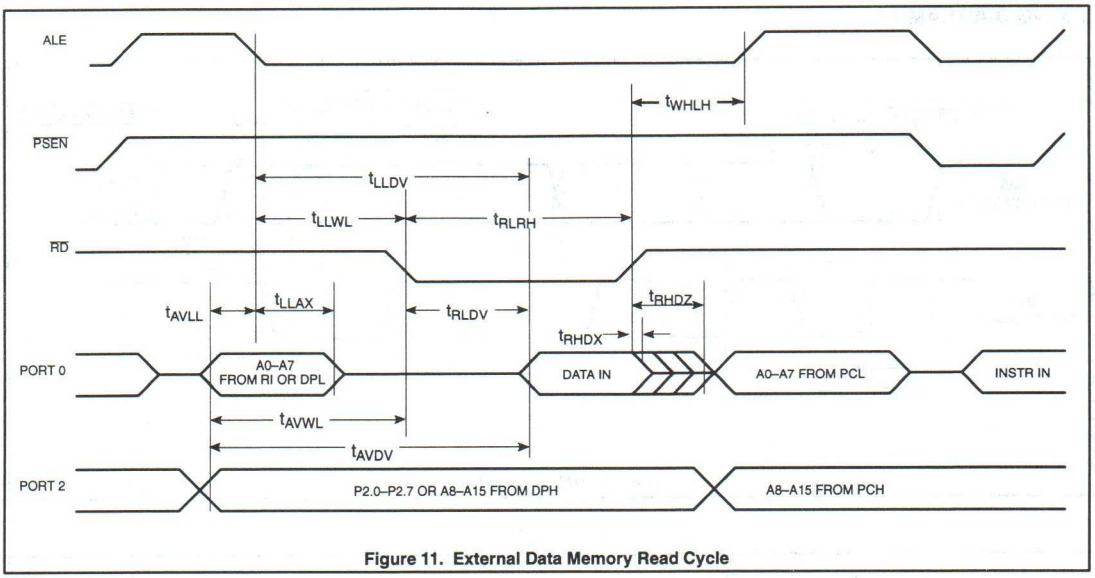



**Figure 11. External Data Memory Read Cycle**

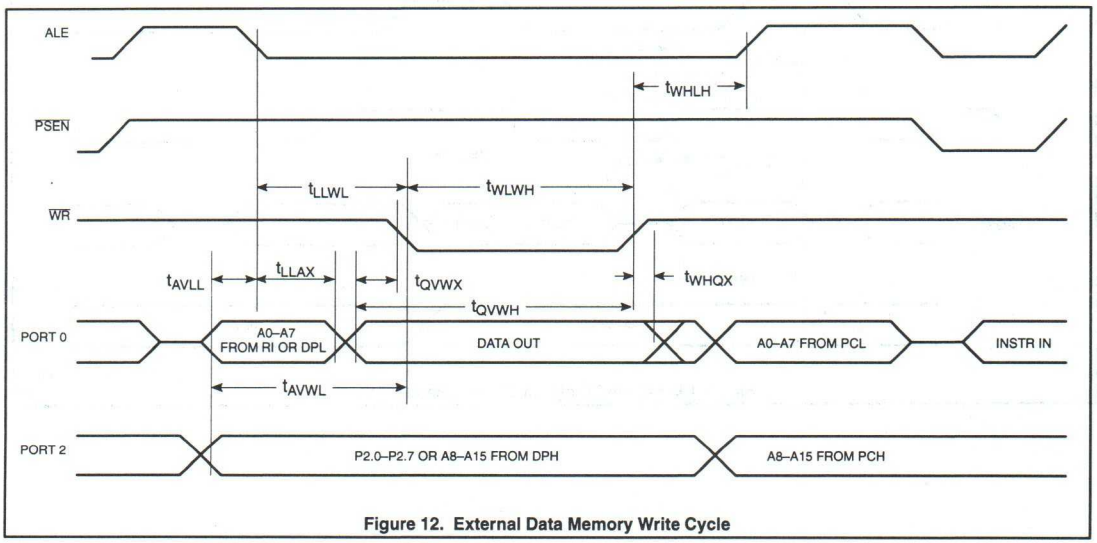
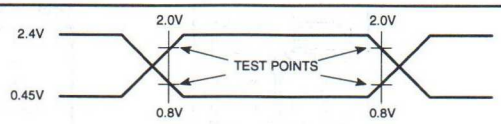
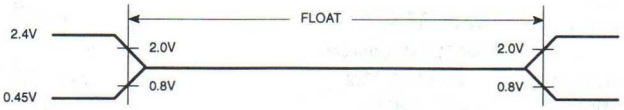



**Figure 12. External Data Memory Write Cycle**




(a)




(b)

AC testing inputs are driven at 2.4V for a logic 1 and 0.45V for a logic 0. Timing measurements are taken at 2.0V for a logic 1 and 0.8V for logic 0, see (a). The float state is defined as the point at which a Port 0 pin sinks 3.2mA or sources 400µA at the voltage test levels, see (b).

**Figure 13. AC Testing Input, Output Waveform (a) and Float Waveform (b)**

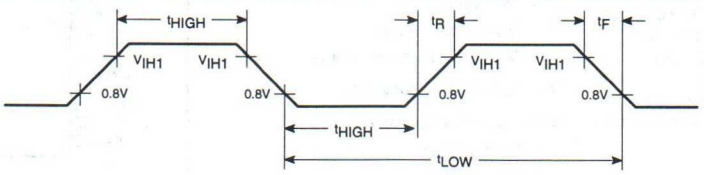



**Figure 14. External Clock Drive XTAL1**

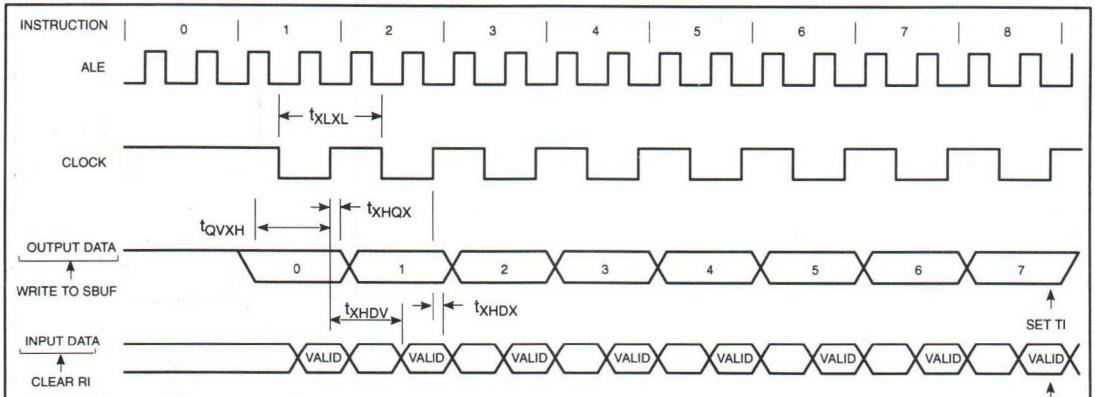



**Figure 15. Shift Register Mode Timing**

# CMOS single-chip 8-bit microcontroller 87C524

## DESCRIPTION

The 87C524 single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 87C524 has the same instruction set as the 80C51.

This device provides architectural enhancements that make it applicable in a variety of applications in consumer, telecom and general control systems, especially in those systems which need large ROM and RAM capacity on-chip.

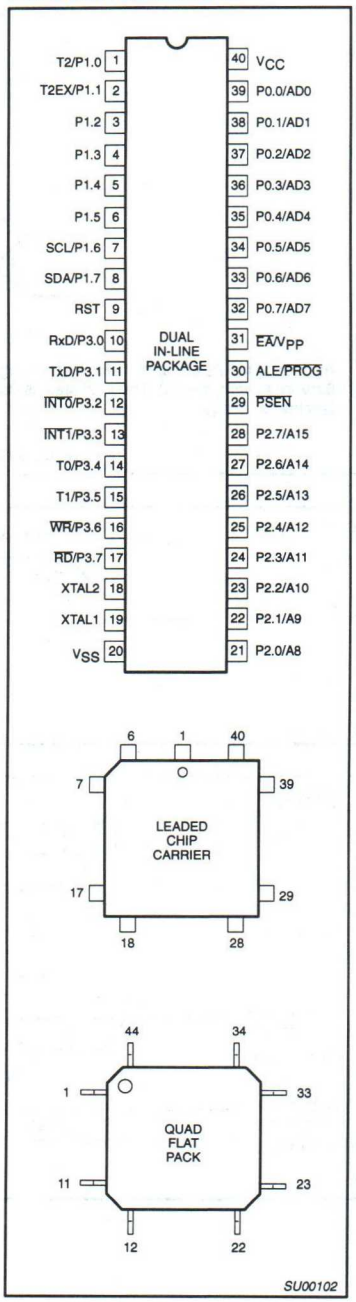
The 87C524 contains a 16k × 8 EPROM, a 512 × 8 RAM, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a 16-bit timer (identical to the timer 2 of the 80C52), a watchdog timer with a separate oscillator, a multi-source, two-priority-level, nested interrupt structure, two serial interfaces (UART and I²C-bus), and on-chip oscillator and timing circuits.

In addition, the 87C524 has two software selectable modes of power reduction—idle mode and power-down mode. The idle mode freezes the CPU while allowing the RAM, timers, serial port, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

## FEATURES

- 80C51 instruction set
  - 16k × 8 EPROM
  - 512 × 8 RAM
  - Memory addressing capability 64k ROM and 64k RAM
  - Three 16-bit counter/timers
  - On-chip watchdog timer with oscillator
  - Full duplex UART
  - I²C serial interface
- Power control modes:
  - Idle mode
  - Power-down mode
  - Warm start from power-down
- CMOS and TTL compatible
- Two speed ranges at $V_{CC}$ = 5V ±10%
  - 3.5 to 16MHz
  - 3.5 to 20MHz
- Extended temperature ranges
- OTP package available
- EPROM code protection

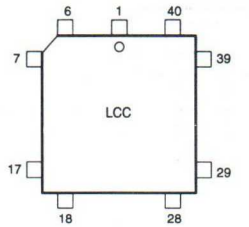
## PIN CONFIGURATIONS



## ORDERING INFORMATION

| EPROM | TEMPERATURE RANGE °C AND PACKAGE | FREQUENCY | DRAWING NUMBER |
|---|---|---|---|
| P87C524EBP N | 0 to +70, Plastic Dual In-line Package | 16MHz | SOT129-1 |
| P87C524EBF FA | 0 to +70, Ceramic Dual In-line Package w/Window | 16MHz | 0590B |
| P87C524EBA A | 0 to +70, Plastic Leaded Chip Carrier | 16MHz | SOT187-2 |
| P87C524EBL KA | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 16MHz | 1472A |
| P87C524EBB B | 0 to +70, Plastic Quad Flat Pack | 16MHz | SOT307-2 |
| P87C524GFP N | –40 to +85, Plastic Dual In-line Package | 20MHz | SOT129-1 |
| P87C524GFF FA | –40 to +85, Ceramic Dual In-line Package w/Window | 20MHz | 0590B |
| P87C524GFA A | –40 to +85, Plastic Leaded Chip Carrier | 20MHz | SOT187-2 |
| P87C524GFL KA | –40 to +85, Ceramic Leaded Chip Carrier w/Window | 20MHz | 1472A |

## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

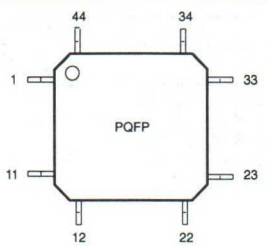



| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC* | 23 | NC* |
| 2 | P1.0/T2 | 24 | P2.0/A8 |
| 3 | P1.1/T2EX | 25 | P2.1/A9 |
| 4 | P1.2 | 26 | P2.2/A10 |
| 5 | P1.3 | 27 | P2.3/A11 |
| 6 | P1.4 | 28 | P2.4/A12 |
| 7 | P1.5 | 29 | P2.5/A13 |
| 8 | P1.6/SCL | 30 | P2.6/A14 |
| 9 | P1.7/SDA | 31 | P2.7/A15 |
| 10 | RST | 32 | $\overline{PSEN}$ |
| 11 | P3.0/RxD | 33 | ALE/$\overline{PROG}$ |
| 12 | NC* | 34 | NC* |
| 13 | P3.1/TxD | 35 | $\overline{EA}$/$V_{PP}$ |
| 14 | P3.2/$\overline{INT0}$ | 36 | P0.7/AD7 |
| 15 | P3.3/$\overline{INT1}$ | 37 | P0.6/AD6 |
| 16 | P3.4/T0 | 38 | P0.5/AD5 |
| 17 | P3.5/T1 | 39 | P0.4/AD4 |
| 18 | P3.6/$\overline{WR}$ | 40 | P0.3/AD3 |
| 19 | P3.7/$\overline{RD}$ | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | $V_{SS}$ | 44 | $V_{CC}$ |

* DO NOT CONNECT

*SU00103A*

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS

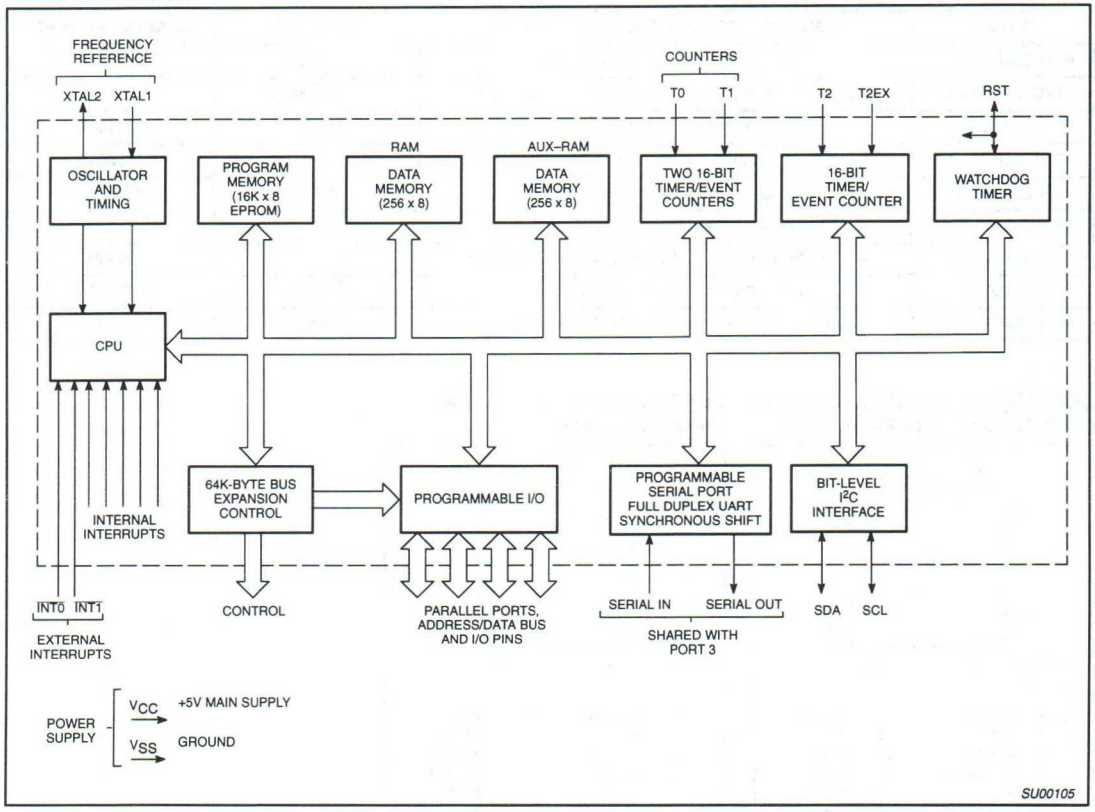



| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5 | 23 | P2.5/A13 |
| 2 | P1.6/SCL | 24 | P2.6/A14 |
| 3 | P1.7/SDA | 25 | P2.7/A15 |
| 4 | RST | 26 | $\overline{PSEN}$ |
| 5 | P3.0/RxD | 27 | ALE/$\overline{PROG}$ |
| 6 | NC* | 28 | NC* |
| 7 | P3.1/TxD | 29 | $\overline{EA}$/$V_{PP}$ |
| 8 | P3.2/$\overline{INT0}$ | 30 | P0.7/AD7 |
| 9 | P3.3/$\overline{INT1}$ | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/$\overline{WR}$ | 34 | P0.3/AD3 |
| 13 | P3.7/$\overline{RD}$ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | $V_{SS}$ | 38 | $V_{CC}$ |
| 17 | NC* | 39 | NC* |
| 18 | P2.0/A8 | 40 | P1.0/T2 |
| 19 | P2.1/A9 | 41 | P1.1/T2EX |
| 20 | P2.2/A10 | 42 | P1.2 |
| 21 | P2.3/A11 | 43 | P.13 |
| 22 | P2.4/A12 | 44 | P1.4 |

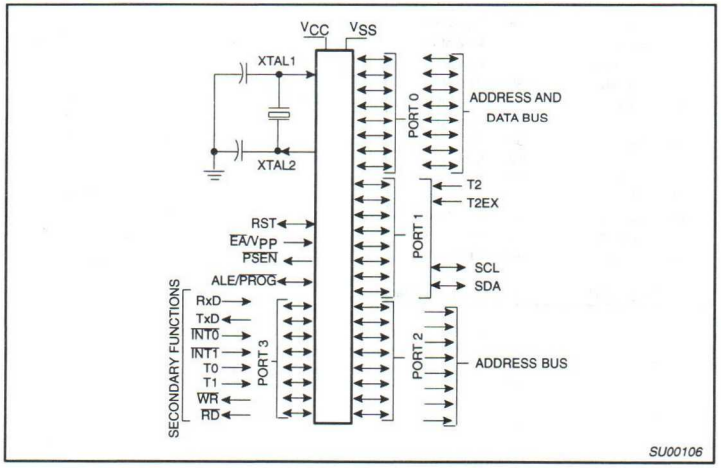
* DO NOT CONNECT

*SU00104A*

## BLOCK DIAGRAM




## LOGIC SYMBOL



## PIN DESCRIPTION

| MNEMONIC | PIN NO. DIP | PIN NO. LCC | PIN NO. QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification in the P87C524. External pull-ups are required during program verification. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups, except P1.6 and P1.7 which are open drain. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 can sink/source one TTL (4LSTTL) inputs. Port 1 receives the low-order address byte during program memory verification. Port 1 also serves alternate functions for timer 2: |
| | 1 | 2 | 40 | I | **T2 (P1.0):** Timer/counter 2 external count input. |
| | 2 | 3 | 41 | I | **T2EX (P1.1):** Timer/counter 2 trigger input. |
| | 7 | 8 | 2 | I/O | **SCL (P1.6):** $I^2C$ serial port clock line. |
| | 8 | 9 | 3 | I/O | **SDA (P1.7):** $I^2C$ serial port data line. |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the SC80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I/O | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{CC}$. After a watchdog timer overflow, this pin is pulled high while the internal reset signal is active. |
| ALE/$\overline{PROG}$ | 30 | 33 | 27 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the device is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}$/$V_{PP}$ | 31 | 35 | 29 | I | **External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H to 7FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 7FFFH. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**Table 1. 8XC524/8XC528 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes): | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | ES1 | ET2 | ES0 | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP*# | Interrupt priority | B8H | – | PS1 | PT2 | PS0 | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | SDA | SEL | – | – | – | – | T2EX | T2 | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | RD | WR | T1 | T0 | INT1 | INT0 | TxD | RxD | FFH |
| PCON | Power control | 87H | SMOD | – | – | – | GF1 | GF0 | PD | IDL | 0xxx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | F1 | P | 00H |
| RCAP2H# | Capture high | CBH | | | | | | | | | 00H |
| RCAP2L# | Capture low | CAH | | | | | | | | | 00H |
| SBUF | Serial data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial controller | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| S1BIT# | Serial I²C data | D9H/RD | SDI | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x0000000B |
| | | WR | SD0 | X | X | X | X | X | X | X | 0xxxxxxxB |
| S1INT# | Serial I²C interrupt | DAH | INT | X | X | X | X | X | X | X | 0xxxxxxxB |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| S1SCS*# | Serial I²C control | D8H/RD | SDI | SCI | CLH | BB | RBF | WBF | STR | ENS | xxxx0000B |
| | | WR | SD0 | SC0 | CLH | X | X | X | STR | ENS | 00xxxx00B |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| T2CON*# | Timer 2 control | C8H | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/T2 | CP/RL2 | 00H |
| TH0 | Timer high 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer high 1 | 8DH | | | | | | | | | 00H |
| TH2# | Timer high 2 | CDH | | | | | | | | | 00H |
| TL0 | Timer low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer low 1 | 8BH | | | | | | | | | 00H |
| TL2# | Timer low 2 | CCH | | | | | | | | | 00H |
| T3# | Watchdog timer | FFH | | | | | | | | | 00H |
| TMOD | Timer mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |
| WDCON# | Watchdog control | A5H | | | | | | | | | A5H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

**Table 2. Internal and External Program Memory Access with Security Bit Set**

| INSTRUCTION | ACCESS TO INTERNAL PROGRAM MEMORY | ACCESS TO EXTERNAL PROGRAM MEMORY |
|---|---|---|
| MOVC in internal program memory | YES | YES |
| MOVC in external program memory | NO | YES |

## INTERNAL DATA MEMORY

The internal data memory is divided into three physically separated segments: 256 bytes of RAM, 256 bytes of AUX-RAM, and a 128 bytes special function area. These can be addressed each in a different way.

- RAM 0 to 127 can be addressed directly and indirectly as in the 80C51. Address pointers are R0 and R1 of the selected register bank.
- RAM 128 to 255 can only be addressed indirectly as in the 80C51. Address pointers are R0 and R1 of the selected register bank.
- AUX-RAM 0 to 255 is indirectly addressed in the same way as external data memory with the MOVX instructions. Address pointers are R0, R1 of the selected register bank and DPTR. An access to AUX-RAM 0 to 255 will not affect ports P0, P2, P3.6 and P3.7.

An access to external data memory locations higher than 255 will be performed with the MOVX DPTR instructions in the same way as in the 8051 structure, so with P0 and P2 as data/address bus and P3.6 and P3.7 as write and read timing signals. Note that these external data memory cannot be accessed with R0 and R1 as address pointer.

## TIMER 2

Timer 2 is functionally equal to the Timer 2 of the 8052AH. Timer 2 is a 16-bit timer/counter. These 16 bits are formed by two special function registers TL2 and TH2. Another pair of special function register RCAP2L and RCAP2H form a 16-bit capture register or a 16-bit reload register. Like Timer 0 and 1, it can operate either as a timer or as an event counter. This is selected by bit C/T2N in the special function register T2CON. It has three operating modes: capture, autoload, and baud rate generator mode which are selected by bits in T2CON.

## WATCHDOG TIMER T3

The watchdog timer consists of an 11-bit prescaler and an 8-bit timer formed by special function register T3. The prescaler is incremented by an on-chip oscillator with a fixed frequency of 1MHz. The maximum tolerance on this frequency is –50% and +100%. The 8-bit timer increments every 2048 cycles of the on-chip oscillator. When a timer overflow occurs, the microcontroller is reset and a reset output pulse of 16 × 2048 cycles of the on-chip oscillator is generated at pin RST. The internal RESET signal is not inhibited when the external RST pin is kept low by, for example, an external reset circuit. The RESET signal drives port 1, 2, 3 into the high state and port 0 into the high impedance state.

The watchdog timer is controlled by one special function register WDCON with the direct address location A5H. WDCON can be read and written by software. A value of A5H in WDCON halts the on-chip oscillator and clears both the prescaler and timer T3. After the RESET signal, WDCON contains A5H. Every value other than A5H in WDCON enables the watchdog timer. When the watchdog timer is enabled, it runs independently of the XTAL-clock.

Timer T3 can be read on the fly. Timer T3 can only be written if WDCON contains the value 5AH. A successful write operation to T3 will clear the prescaler and WDCON, leaving the watchdog enabled and preventing inadvertent changes of T3. To prevent an overflow of the watchdog timer, the user program has to reload the watchdog timer within periods that are shorter than the programmed watchdog timer internal. This time interval is determined by an 8-bit value that has to be loaded in register T3 while at the same time the prescaler is cleared by hardware.

$$\text{Watchdog timer interval} = \frac{[256 - (T3)] \times 2048}{\text{on - chip oscillator frequency}}$$

## BIT-LEVEL I²C INTERFACE

This bit-level serial I/O interface supports the I²C-bus. P1.6/SCL and P1.7/SDA are the serial I/O pins. These two pins meet the I²C specification concerning the input levels and output drive capability. Consequently, these pins have an open drain output configuration. All the four modes of the I²C-bus are supported:

- master transmitter
- master receiver
- slave transmitter
- slave receiver

The advantages of the bit-level I²C hardware compared with a full software I²C implementation are:

- the hardware can generate the SCL pulse
- Testing a single bit (RBF respectively, WBF) is sufficient as a check for error free transmission.

The bit-level I²C hardware operates on serial bit level and performs the following functions:

- filtering the incoming serial data and clock signals
- recognizing the START condition
- generating a serial interrupt request SI after reception of a START condition and the first falling edge of the serial clock
- recognizing the STOP condition
- recognizing a serial clock pulse on the SCL line
- latching a serial bit on the SDA line (SDI)
- stretching the SCL LOW period of the serial clock to suspend the transfer of the next serial data bit
- setting Read Bit Finished (RBF) when the SCL clock pulse has finished and Write Bit Finished (WBF) if there is no arbitration loss detected (i.e., SDA = 0 while SDO = 1)
- setting a serial clock Low-to-High detected (CLH) flag
- setting a Bus Busy (BB) flag on a START condition and clearing this flag on a STOP condition
- releasing the SCL line and clearing the CLH, RBF and WBF flags to resume transfer of the next serial data bit
- generating an automatic clock if the single bit data register S1BIT is used in master mode.

The following functions must be done in software:

- handling the I²C START interrupts
- converting serial to parallel data when receiving
- converting parallel to serial data when transmitting
- comparing the received slave address with its own
- interpreting the acknowledge information
- guarding the I²C status if RBF or WBF = 0.

Additionally, if acting as master:

- generating START and STOP conditions
- handling bus arbitration
- generating serial clock pulses if S1BIT is not used.

Three SFRs control the bit-level I²C interface: S1INT, S1BIT and S1SCS.

## INTERRUPT SYSTEM

The interrupt structure of the 8XC524 is the same as that used in the 80C51, but includes two additional interrupt sources: one for the third timer/counter, T2, and one for the $I^2C$ interface. The interrupt enable and interrupt priority registers are IE and IP.

### IE: Interrupt Enable Register

This register is located at address A8H. Refer to Table 3.

**IE SFR (A8H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| EA | ES1 | ET2 | ES | ET1 | EX1 | ET0 | EX0 |

### IP: Interrupt Priority Register

This register is located at address B8H. Refer to Table 4.

**IP SFR (B8H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | PS1 | PT2 | PS | PT1 | PX1 | PT0 | PX0 |

**Table 3. Description of IE Bits**

| MNEMONIC | BIT | FUNCTION |
|---|---|---|
| EA | IE.7 | General enable/disable control:<br>0 = **NO** interrupt is enabled.<br>1 = **ANY** individually enabled interrupt will be accepted. |
| ES1 | IE.6 | Enable **bit-level $I_2C$ I/O** interrupt |
| ET2 | IE.5 | Enable **Timer 2** interrupt |
| ES | IE.4 | Enable **Serial Port** interrupt |
| ET1 | IE.3 | Enable **Timer 1** interrupt |
| EX1 | IE.2 | Enable **External** interrupt 1 |
| ET0 | IE.1 | Enable **Timer 0** interrupt |
| EX0 | IE.0 | Enable **External** interrupt 0 |

**Table 4. Description of IP Bits**

| MNEMONIC | BIT | FUNCTION |
|---|---|---|
| – | IP.7 | **Reserved.** |
| PS1 | IP.6 | **Bit-level $I^2C$** interrupt priority level |
| PT2 | IP.5 | **Timer 2** interrupt priority level |
| PS | IP.4 | **Serial Port** interrupt priority level |
| PT1 | IP.3 | **Timer 1** interrupt priority level |
| PX1 | IP.2 | **External Interrupt 1** priority level |
| PT0 | IP.1 | **Timer 0** interrupt priority level |
| PX0 | IP.0 | **External Interrupt 0** priority level |

The interrupt vector locations and the interrupt priorities are:

| **Source** | | **Priority within Level** |
|---|---|---|
| Vector | Address | |
| 0003H | IE0 | Highest |
| 002BH | TF2+EXF2 | |
| 0053H | SI ($I^2C$) | |
| 000BH | TF0 | |
| 0013H | IE1 | |
| 001BH | TF1 | |
| 0023H | R1+T1 | Lowest |

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the logic symbol, page 3-374.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-up reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-up, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up.

## IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. The power-down mode can be terminated by a RESET in the same way as in the 80C51 or in addition by one of two external interrupts, INT0 or INT1. A termination with an external interrupt does not affect the internal data memory and does not affect the special function registers. This makes it possible to exit power-down without changing the port output levels. To terminate the power-down mode with an external interrupt $\overline{INT0}$ or $\overline{INT1}$ must be switched to level-sensitive and must be enabled. The external interrupt input signal $\overline{INT0}$ and $\overline{INT1}$ must be kept low until the oscillator has restarted and stabilized. An instruction following the instruction that puts the device in the power-down mode will be executed. A reset generated by the watchdog timer terminates the power-down mode in the same way as an external RESET, and only the contents of the on-chip RAM are preserved. The control bits for the reduced power modes are in the special function register PCON.

## DESIGN CONSIDERATIONS

At power-on, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up.

When the idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when idle is terminated by reset, the instruction following the one that invokes idle should not be one that writes to a port pin or to external memory.

Table 5 shows the state of I/O ports during low current operating modes.

**Table 5. External Pin Status During Idle and Power-Down Modes**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | 0 to +70 or –40 to +85 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to $V_{CC}$ +0.5 | V |
| Input, output current on any two pins | ±10 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.0 | W |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | PART TYPE | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input low voltage,<br>except $\overline{EA}$, P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C | | –0.5<br>–0.5 | $0.2V_{CC}$–0.1<br>$0.2V_{CC}$–0.15 | V<br>V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | 0 to +70°C<br>–40 to +85°C | | 0<br>0 | $0.2V_{CC}$–0.3<br>$0.2V_{CC}$–0.35 | V<br>V |
| $V_{IL2}$ | Input low voltage to P1.6/SCL, P1.7SDA[5] | | | –0.5 | 1.5 | V |
| $V_{IH}$ | Input high voltage,<br>except XTAL1, RST, P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C | | $0.2V_{CC}$+0.9<br>$0.2V_{CC}$+1 | $V_{CC}$+0.5<br>$V_{CC}$+0.5 | V<br>V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | 0 to +70°C<br>–40 to +85°C | | $0.7V_{CC}$<br>$0.7V_{CC}$+0.1 | $V_{CC}$+0.5<br>$V_{CC}$+0.5 | V<br>V |
| $V_{IH2}$ | Input high voltage, P1.6/SCL, P1.7/SDA[5] | | | 3.0 | 6.0 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3,<br>except P1.6/SCL, P1.7/SDA | | $I_{OL}$ = 1.6mA[1] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$ | | $I_{OL}$ = 3.2mA[1] | | 0.45 | V |
| $V_{OL2}$ | Output low voltage, P1.6/SCL, P1.7/SDA | | $I_{OL}$ = 3.0mA[1] | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3[2] | | $I_{OH}$ = –60µA<br>$I_{OH}$ = –25µA<br>$I_{OH}$ = –10µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$ | | V<br>V<br>V |
| $V_{OH1}$ | Output high voltage, Port 0 in external bus mode,<br>ALE, $\overline{PSEN}$, RST | | $I_{OH}$ = –800µA<br>$I_{OH}$ = –300µA<br>$I_{OH}$ = –80µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$ | | V<br>V<br>V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3,<br>except P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C | $V_{IN}$ = 0.45V | | –50<br>–75 | µA<br>µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3,<br>except P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C | See Note 3 | | –650<br>–750 | µA<br>µA |
| $I_{L1}$ | Input leakage current, port 0 | | $V_{IN} = V_{IL}$ or $V_{IH}$ | | ±10 | µA |
| $I_{L2}$ | Input leakage current, P1.6/SCL, P1.7/SDA | | 0V < $V_i$ < 6V<br>0V < $V_{CC}$ < 5.5V | | ±10 | µA |
| $I_{CC}$ | Power supply current:<br>Active mode @ 16MHz<br><br>Idle mode @ 16MHz<br><br>Power down mode | <br>0 to +70°C<br>–40 to +85°C<br>0 to +70°C<br>–40 to +85°C | See Note 4 | | <br>25<br>35<br>5<br>6<br>50 | <br>mA<br>mA<br>mA<br>mA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | | 50 | 300 | kΩ |
| $C_{IO}$ | Pin capacitance | | | | 10 | pF |

**NOTES:**

1. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows: 10mA per port pin, port 0 total (all bits) 26mA, ports 1, 2, and total each (all bits) 15mA.
2. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
3. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
4. See Figures 10 through 13 for $I_{CC}$ test conditions.
5. The input threshold voltage of P1.6 and P1.7 (SI01) meets the $I^2C$ specification, so an input voltage below 1.5V will be recognized as a logic 0 while an input voltage above 3.0V will be recognized as a logic 1.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 1 | Oscillator frequency: **Speed Versions**<br>87C524 E<br>87C524 G | | | <br>3.5<br>3.5 | <br>16<br>20 | <br>MHz<br>MHz |
| $t_{LHLL}$ | 1 | ALE pulse width | 85 | | $2t_{CLCL}$–40 | | ns |
| $t_{AVLL}$ | 1 | Address valid to ALE low | 8 | | $t_{CLCL}$–55 | | ns |
| $t_{LLAX}$ | 1 | Address hold after ALE low | 28 | | $t_{CLCL}$–35 | | ns |
| $t_{LLIV}$ | 1 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}$–100 | ns |
| $t_{LLPL}$ | 1 | ALE low to $\overline{PSEN}$ low | 23 | | $t_{CLCL}$–40 | | ns |
| $t_{PLPH}$ | 1 | $\overline{PSEN}$ pulse width | 143 | | $3t_{CLCL}$–45 | | ns |
| $t_{PLIV}$ | 1 | $\overline{PSEN}$ low to valid instruction in | | 83 | | $3t_{CLCL}$–105 | ns |
| $t_{PXIX}$ | 1 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 1 | Input instruction float after $\overline{PSEN}$ | | 38 | | $t_{CLCL}$–25 | ns |
| $t_{AVIV}$ | 1 | Address to valid instruction in | | 208 | | $5t_{CLCL}$–105 | ns |
| $t_{PLAZ}$ | 1 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 2, 3 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}$–100 | | ns |
| $t_{WLWH}$ | 2, 3 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}$–100 | | ns |
| $t_{RLDV}$ | 2, 3 | $\overline{RD}$ low to valid data in | | 148 | | $5t_{CLCL}$–165 | ns |
| $t_{RHDX}$ | 2, 3 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 2, 3 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}$–70 | ns |
| $t_{LLDV}$ | 2, 3 | ALE low to valid data in | | 350 | | $8t_{CLCL}$–150 | ns |
| $t_{AVDV}$ | 2, 3 | Address to valid data in | | 398 | | $9t_{CLCL}$–165 | ns |
| $t_{LLWL}$ | 2, 3 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 138 | 238 | $3t_{CLCL}$–50 | $3t_{CLCL}$+50 | ns |
| $t_{AVWL}$ | 2, 3 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 120 | | $4t_{CLCL}$–130 | | ns |
| $t_{QVWX}$ | 2, 3 | Data valid to $\overline{WR}$ transition | 3 | | $t_{CLCL}$–60 | | ns |
| $t_{WHQX}$ | 2, 3 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}$–50 | | ns |
| $t_{RLAZ}$ | 2, 3 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 2, 3 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}$–40 | $t_{CLCL}$+40 | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 6 | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | 6 | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | 6 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 6 | Fall time | | 20 | | 20 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 4 | Serial port clock cycle time | 750 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 4 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}$–133 | | ns |
| $t_{XHQX}$ | 4 | Output data hold after clock rising edge | 8 | | $2t_{CLCL}$–117 | | ns |
| $t_{XHDX}$ | 4 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 4 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}$–133 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.

## AC ELECTRICAL CHARACTERISTICS – I²C INTERFACE

| SYMBOL | PARAMETER | INPUT | OUTPUT | I²C SPECIFICATION |
|---|---|---|---|---|
| **SCL Timing Characteristics** | | | | |
| $t_{HD;STA}$ | START condition hold time | ≥ 14 $t_{CLCL}$[1] | Note 2 | ≥ 4.0µs |
| $t_{LOW}$ | SCL LOW time | ≥ 16 $t_{CLCL}$ | Note 2 | ≥ 4.7µs |
| $t_{HIGH}$ | SCL HIGH time | ≥ 14 $t_{CLCL}$[1] | ≥ 80 $t_{CLCL}$[3] | ≥ 4.0µs |
| $t_{RC}$ | SCL rise time | ≤ 1µs[4] | Note 5 | ≤ 1.0µs |
| $t_{FC}$ | SCL fall time | ≤ 0.3µs[4] | ≤ 0.3µs [6] | ≤ 0.3µs |
| **SDA Timing Characteristics** | | | | |
| $t_{SU;DAT}$ | Data set-up time | ≥ 250ns | Note 2 | ≥ 250ns |
| $t_{HD;DAT}$ | Data hold time | ≥ 0ns | Note 2 | ≥ 0ns |
| $t_{SU;STA}$ | Repeated START set-up time | ≥ 14 $t_{CLCL}$[1] | Note 2 | ≥ 4.7µs |
| $t_{SU;STO}$ | STOP condition set-up time | ≥ 14 $t_{CLCL}$[1] | Note 2 | ≥ 4.0µs |
| $t_{BUF}$ | Bus free time | ≥ 14 $t_{CLCL}$[1] | Note 2 | ≥ 4.7µs |
| $t_{RD}$ | SDA rise time | ≤ 1µs[4] | Note 5 | ≤ 1.0µs |
| $t_{FD}$ | SDA fall time | ≤ 0.3µs[4] | ≤ 0.3µs [6] | ≤ 0.3µs |

**NOTES:**
1. At $f_{CLK}$ = 3.5MHz, this evaluates to 14 × 286ns = 4µs, i.e., the bit-level I²C interface can respond to the I²C protocol for $f_{CLK}$ ≥ 3.5MHz.
2. This parameter is determined by the user software, it has to comply with the I²C specification.
3. This value gives the autoclock pulse length which meets the I²C specification for the specified XTAL clock frequency range. Alternatively, the SCL pulse may be timed by software.
4. Spikes on SDA and SCL lines with a duration of less than 4 × $f_{CLK}$ will be filtered out.
5. The rise time is determined by the external bus line capacitance and pull-up resistor, it must be ≤ 1µs.
6. The maximum capacitance on bus lines SDA and SCL is 400pF.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{PSEN}$ low.

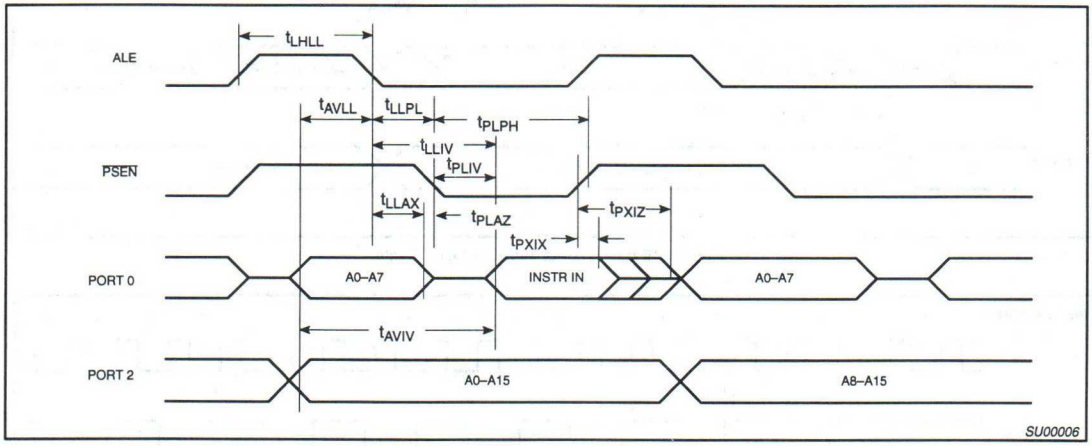



**Figure 1. External Program Memory Read Cycle**

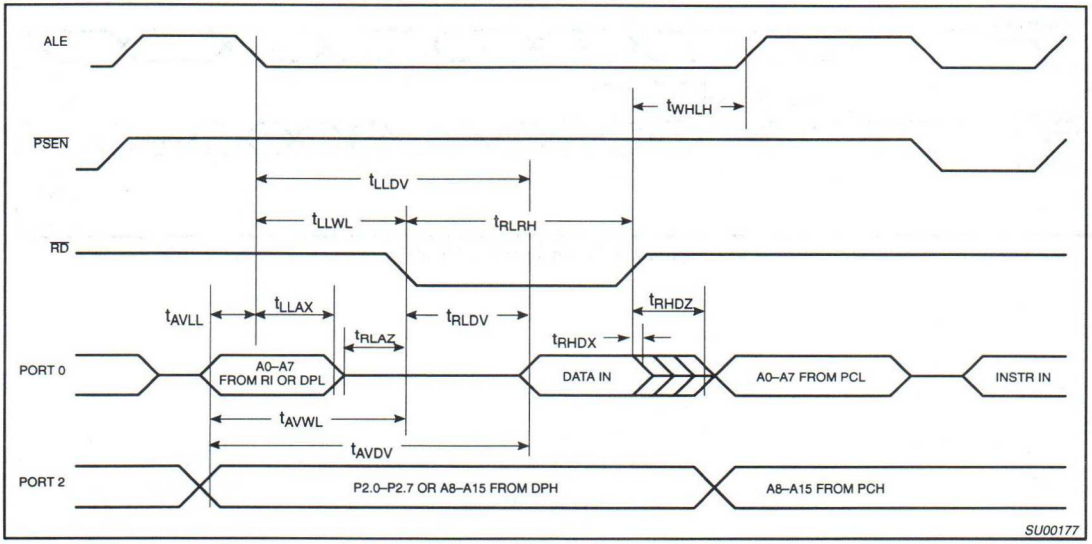



**Figure 2. External Data Memory Read Cycle**

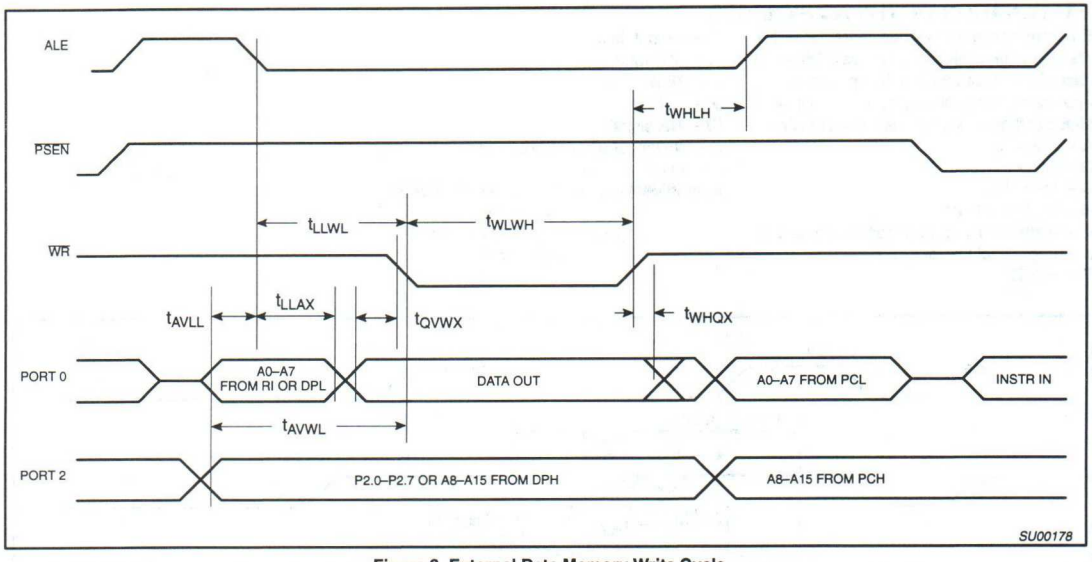



**Figure 3. External Data Memory Write Cycle**

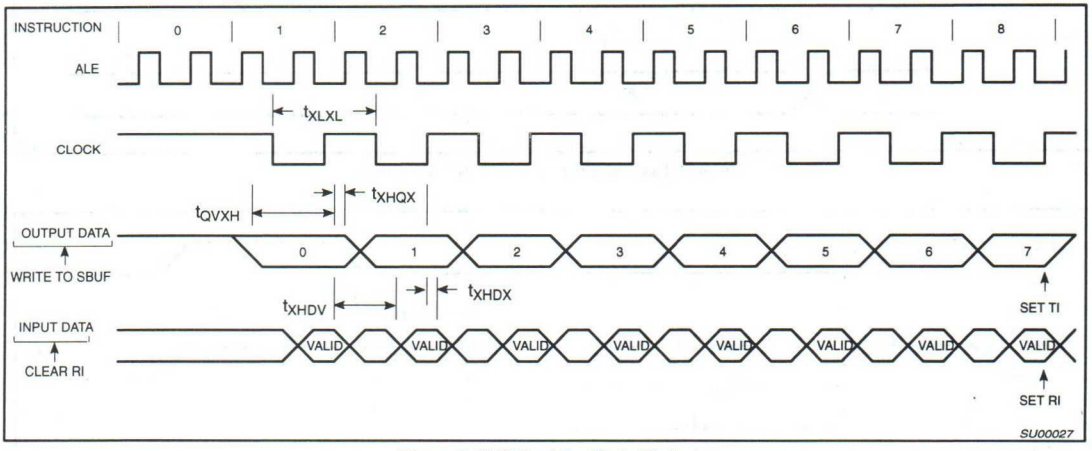



**Figure 4. Shift Register Mode Timing**

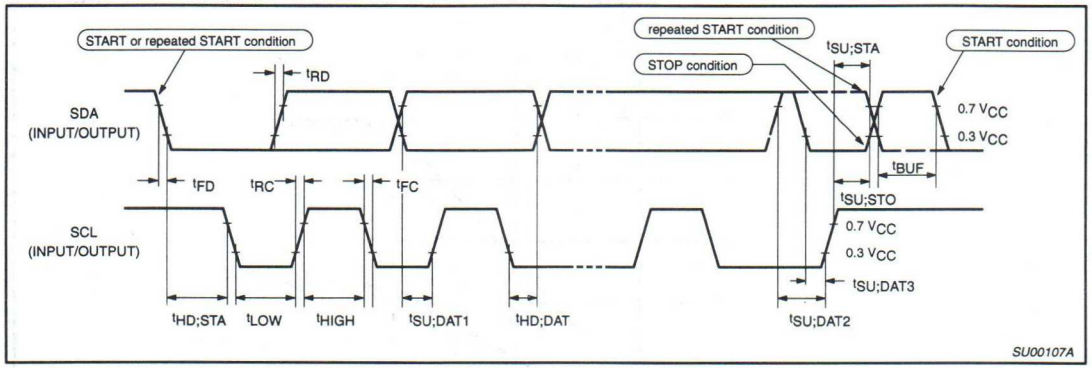



**Figure 5. Timing SIO1 ($I^2C$) Interface**

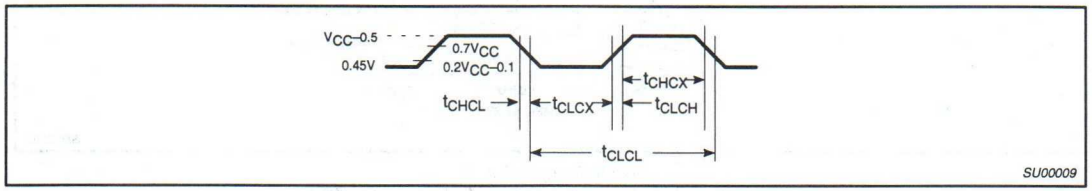



**Figure 6. External Clock Drive**

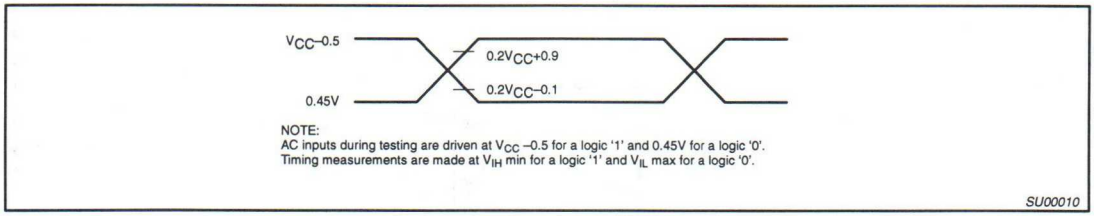



**Figure 7. AC Testing Input/Output**

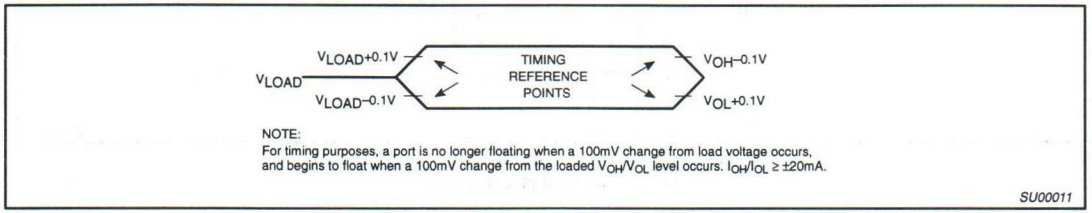



**Figure 8. Float Waveform**

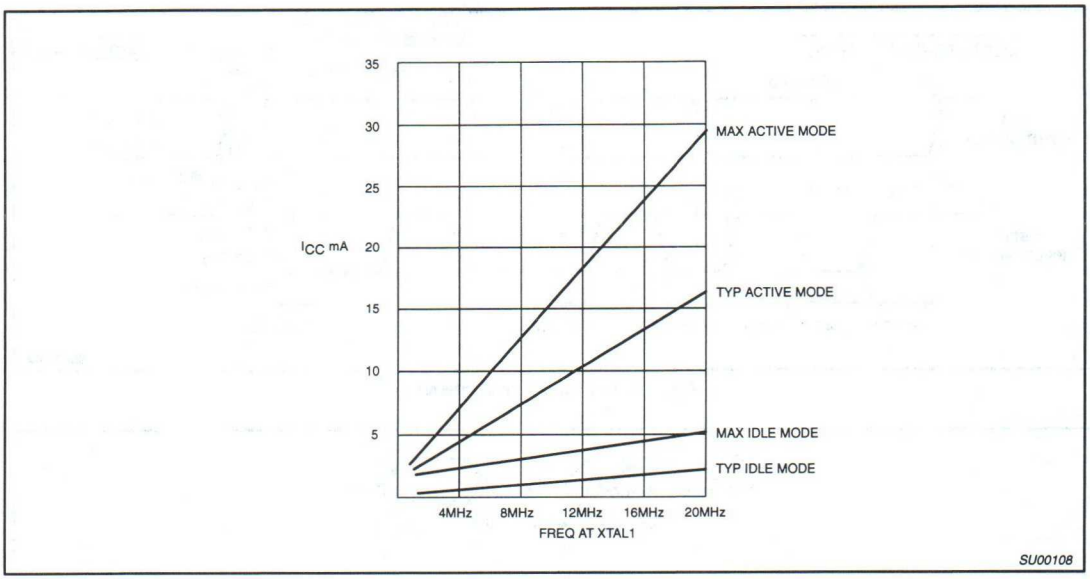



**Figure 9. $I_{CC}$ vs. FREQ**
**Valid only within frequency specifications of the device under test**

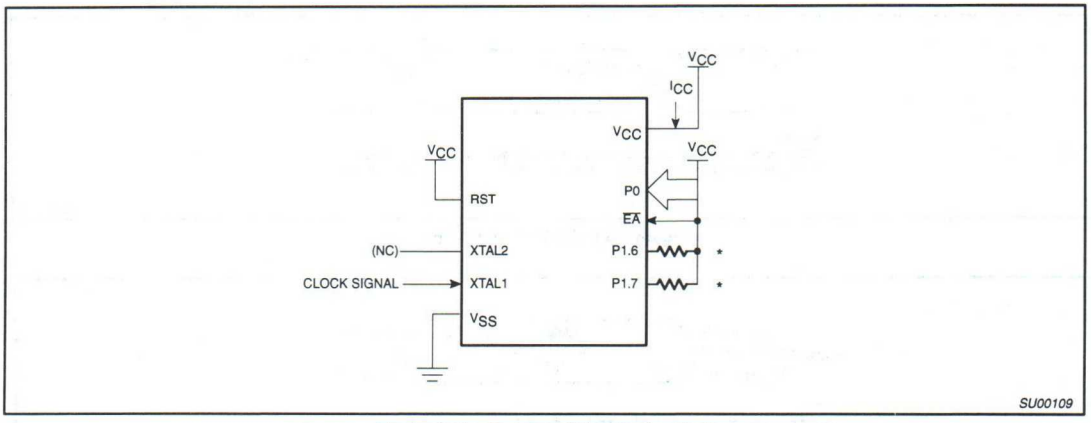



**Figure 10. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

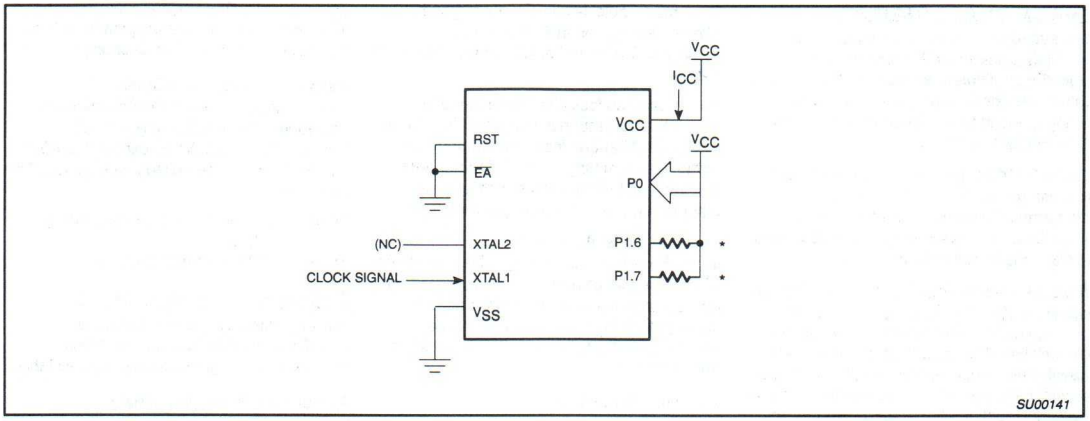



**Figure 11. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

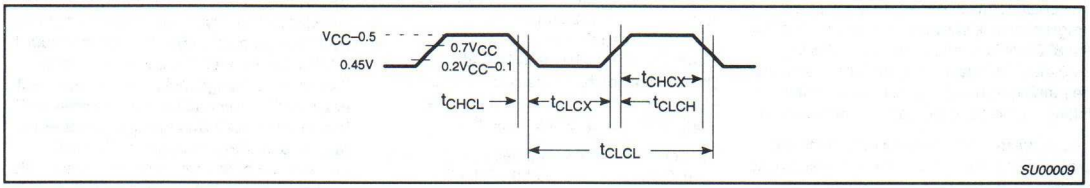



**Figure 12. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

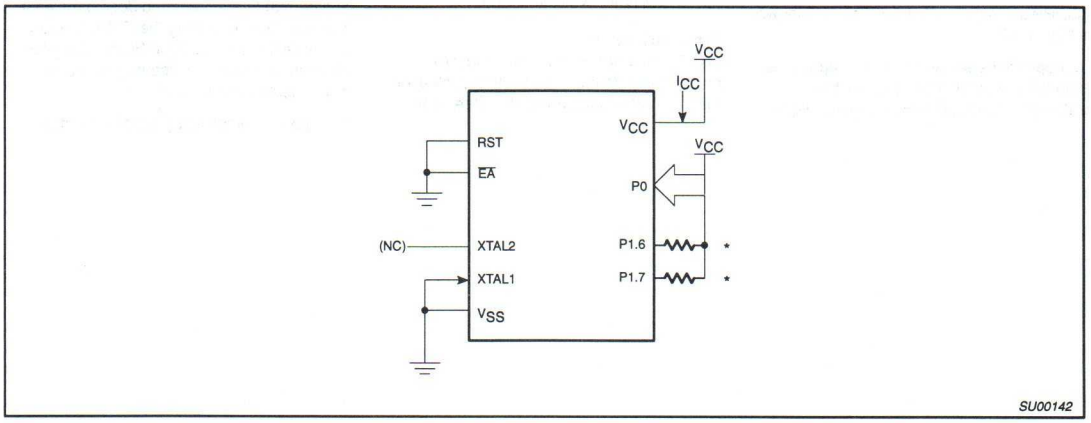



**Figure 13. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2V to 5.5V**

**NOTE:**
* Ports 1.6 and 1.7 should be connected to $V_{CC}$ through resistors of sufficiently high value such that the sink current into these pins does not exceed the $I_{OL1}$ specification.

## EPROM CHARACTERISTICS

The 87C524 is programmed by using a modified Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C524 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C524 manufactured by Philips.

Table 6 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the lock bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 14 and 15. Figure 16 shows the circuit configuration for normal program memory verification.

## Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 14. Note that the 87C524 is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1, 2 and 3, as shown in Figure 14. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 3 specified in Table 6 are held at the 'Program Code Data' levels indicated in Table 6. The ALE/$\overline{PROG}$ is pulsed low 25 times as shown in Figure 15.

To program the encryption table, repeat the 25 pulse programming sequence for addresses 0 through 3FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the lock bits, repeat the 25 pulse programming sequence using the 'Pgm Lock Bit' levels. After one lock bit is programmed, further programming of the code memory and encryption table is disabled. However, the other lock bit can still be programmed.

Note that the $\overline{EA}$/$V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

### Program Verification

If lock bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 1, 2 and 3 as shown in Figure 16. The other pins are held at the 'Verify Code Data' levels indicated in Table 6. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

### Program Lock Bits

The 87C524 has 3 programmable lock bits that will provide different levels of protection for the on-chip code and data. (See Table 7.)

Erasing the EPROM also erases the encryption array and the program lock bits, returning the part to full functionality.

### Reading the Signature Bytes

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:

(030H) = 15H indicates manufactured by Philips
(031H) = 9DH indicates 87C524

## Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 6, and which satisfies the timing specifications, is suitable.

## Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-sec/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000uW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

™Trademark phrase of Intel Corporation.

### Table 6. EPROM Programming Modes

| MODE | RST | PSEN | ALE/PROG | EA/$V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm lock bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm lock bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |
| Pgm lock bit 3 | 1 | 0 | 0* | $V_{PP}$ | 0 | 1 | 0 | 1 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V±10% during programming and verification.
* ALE/PROG receives 25 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

### Table 7. Program Lock Bits

| PROGRAM LOCK BITS[1, 2] | | | | |
|---|---|---|---|---|
| | LB1 | LB2 | LB3 | PROTECTION DESCRIPTION |
| 1 | U | U | U | No Program Lock features enabled. (Code verify will still be encrypted by the Encryption Array if programmed.) |
| 2 | P | U | U | MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory, EA is sampled and latched on Reset, and further programming of the EPROM is disabled. |
| 3 | P | P | U | Same as 2, also verify is disabled. |
| 4 | P | P | P | Same as 3, external execution is disabled. Internal data RAM is not accessible. |

**NOTES:**
1. P – programmed. U – unprogrammed.
2. Any other combination of the lock bits is not defined.

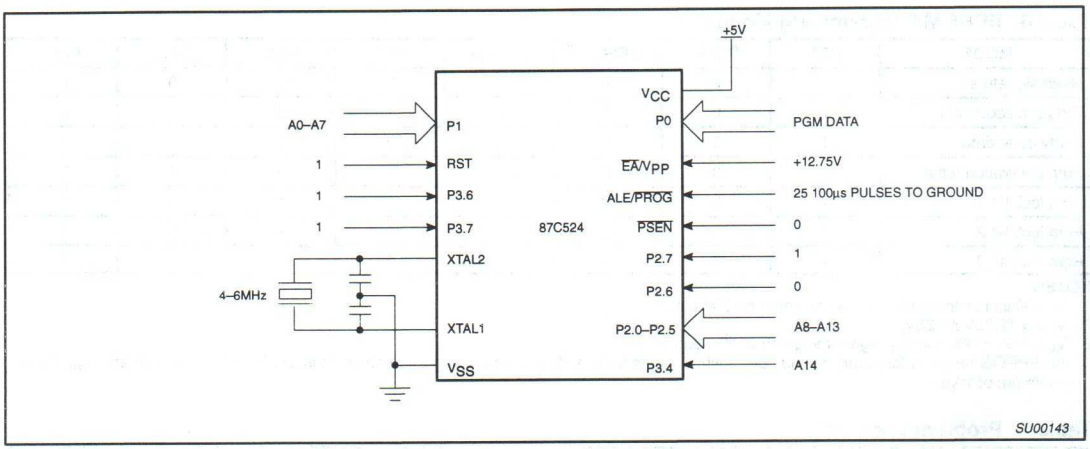



**Figure 14. Programming Configuration**

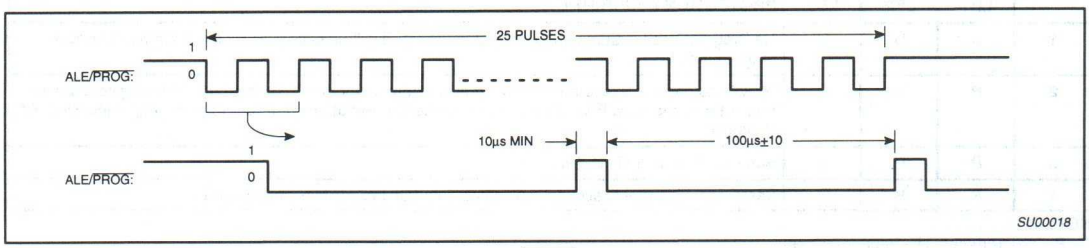



**Figure 15. PROG Waveform**

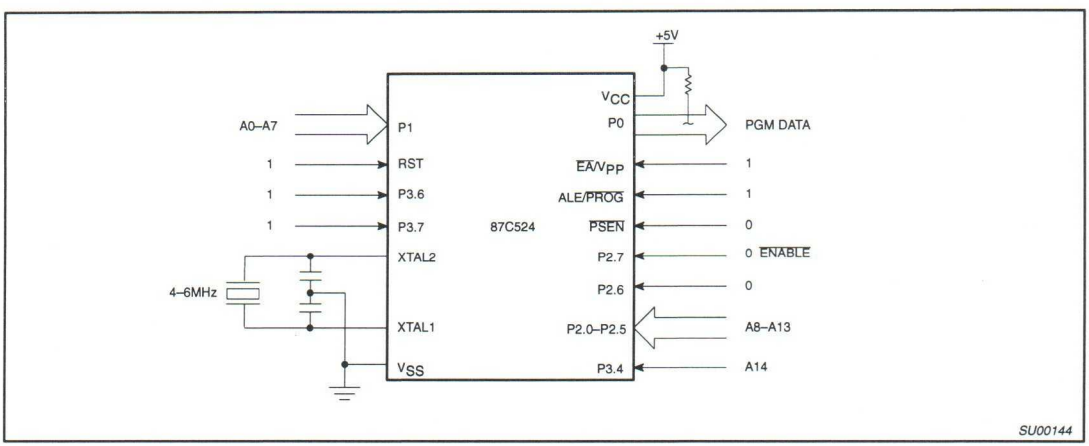
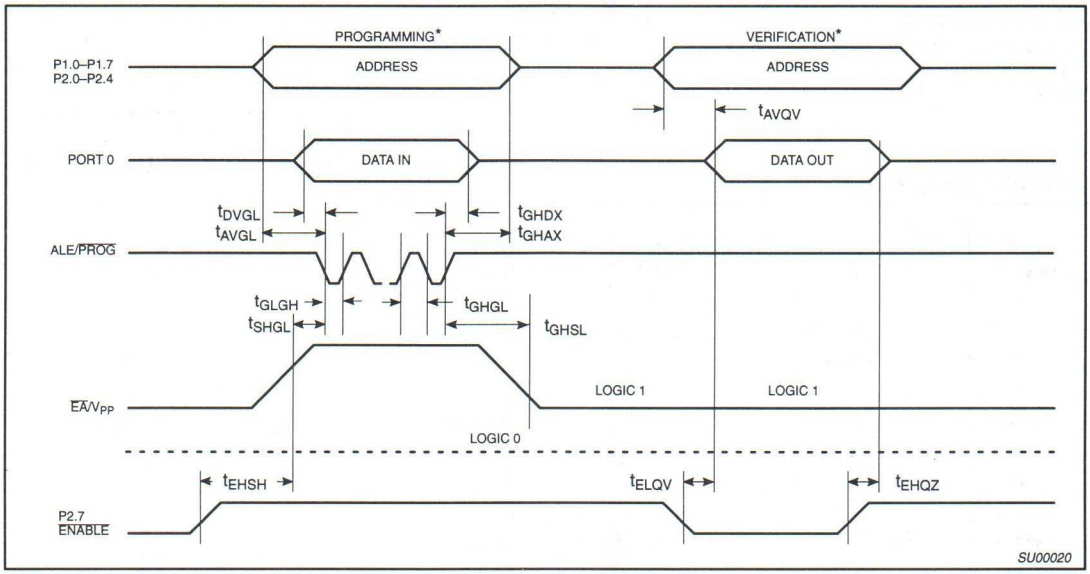



**Figure 16. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (See Figure 17)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




**NOTE:**

* FOR PROGRAMMING VERIFICATION SEE FIGURE 14.
FOR VERIFICATION CONDITIONS SEE FIGURE 16.

**Figure 17. EPROM Programming and Verification**

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# CMOS single-chip 8-bit microcontrollers

# 80C528/83C528

## DESCRIPTION

The 8XC528 single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 8XC528 has the same instruction set as the 80C51. Three versions of the derivative exist:

- 83C528 — 32k bytes mask programmable ROM
- 80C528 — ROMless version of the 83C528
- 87C528 — 32k bytes EPROM (described in a separate data sheet)

This device provides architectural enhancements that make it applicable in a variety of applications in consumer, telecom and general control systems, especially in those systems which need large ROM and RAM capacity on-chip.

The 8XC528 contains a 32k × 8 ROM (83C528), a 512 × 8 RAM, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a 16-bit timer (identical to the timer 2 of the 80C52), a watchdog timer with a separate oscillator, a multi-source, two-priority-level, nested interrupt structure, two serial interfaces (UART and I²C-bus), and on-chip oscillator and timing circuits.

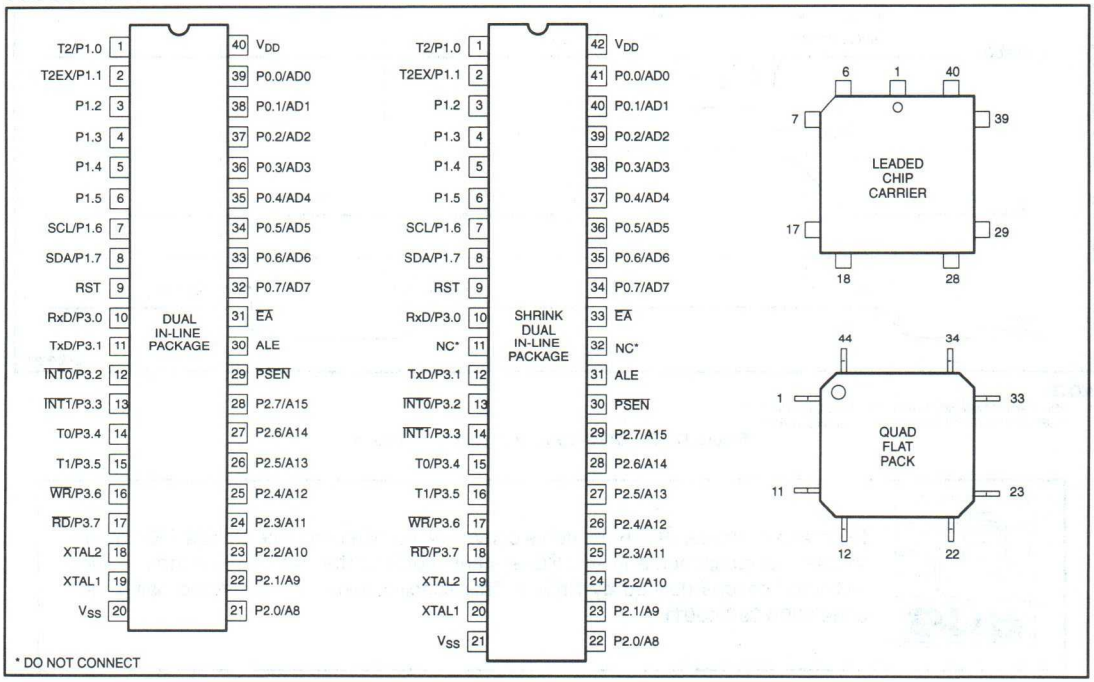
In addition, the 8XC528 has two software selectable modes of power reduction — idle mode and power-down mode. The idle mode freezes the CPU while allowing the RAM, timers, serial port, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.



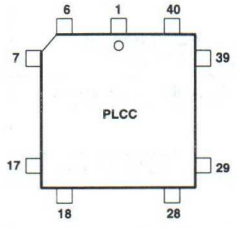

## FEATURES

- 80C51 instruction set
  - 32k × 8 ROM (83C528)
  - ROMless (80C528)
  - 512 × 8 RAM
  - Memory addressing capability 64k ROM and 64k RAM
  - Three 16-bit counter/timers
  - On-chip watchdog timer with oscillator
  - Full duplex UART
  - I²C serial interface
  - Four 8-bit I/O ports
- Power control modes:
  - Idle mode
  - Power-down mode
  - Warm start from power-down
- CMOS and TTL compatible
- Extended temperature ranges
- ROM code protection
- 7-source and 7-vector interrupt structure with 2 priority levels
- Up to 3 external interrupt request inputs
- Two programmable power reduction modes (Idle and Power-down)
- Termination of Idle mode by any interrupt, external or WDT (watchdog) reset
- XTAL frequency range: 1.2 MHz to 16 MHz

## PIN CONFIGURATIONS



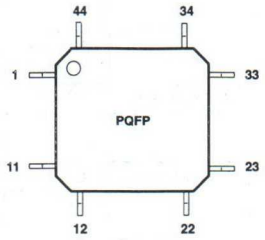
## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC* | 23 | NC* |
| 2 | P1.0/T2 | 24 | P2.0/A8 |
| 3 | P1.1/T2EX | 25 | P2.1/A9 |
| 4 | P1.2 | 26 | P2.2/A10 |
| 5 | P1.3 | 27 | P2.3/A11 |
| 6 | P1.4 | 28 | P2.4/A12 |
| 7 | P1.5 | 29 | P2.5/A13 |
| 8 | P1.6/SCL | 30 | P2.6/A14 |
| 9 | P1.7/SDA | 31 | P2.7/A15 |
| 10 | RST | 32 | $\overline{PSEN}$ |
| 11 | P3.0/RxD | 33 | ALE |
| 12 | NC* | 34 | NC* |
| 13 | P3.1/TxD | 35 | $\overline{EA}$ |
| 14 | P3.2/$\overline{INT0}$ | 36 | P0.7/AD7 |
| 15 | P3.3/$\overline{INT1}$ | 37 | P0.6/AD6 |
| 16 | P3.4/T0 | 38 | P0.5/AD5 |
| 17 | P3.5/T1 | 39 | P0.4/AD4 |
| 18 | P3.6/$\overline{WR}$ | 40 | P0.3/AD3 |
| 19 | P3.7/$\overline{RD}$ | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | $V_{SS}$ | 44 | $V_{DD}$ |

* DO NOT CONNECT

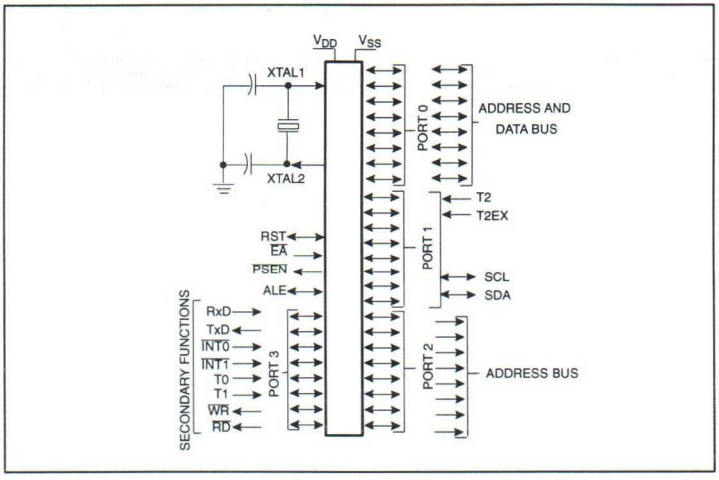
## PLASTIC QUAD FLAT PACK PIN FUNCTIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5 | 23 | P2.5/A13 |
| 2 | P1.6/SCL | 24 | P2.6/A14 |
| 3 | P1.7/SDA | 25 | P2.7/A15 |
| 4 | RST | 26 | $\overline{PSEN}$ |
| 5 | P3.0/RxD | 27 | ALE |
| 6 | NC* | 28 | NC* |
| 7 | P3.1/TxD | 29 | $\overline{EA}$ |
| 8 | P3.2/$\overline{INT0}$ | 30 | P0.7/AD7 |
| 9 | P3.3/$\overline{INT1}$ | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/$\overline{WR}$ | 34 | P0.3/AD3 |
| 13 | P3.7/$\overline{RD}$ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | $V_{SS}$ | 38 | $V_{DD}$ |
| 17 | NC* | 39 | NC* |
| 18 | P2.0/A8 | 40 | P1.0/T2 |
| 19 | P2.1/A9 | 41 | P1.1/T2EX |
| 20 | P2.2/A10 | 42 | P1.2 |
| 21 | P2.3/A11 | 43 | P1.3 |
| 22 | P2.4/A12 | 44 | P1.4 |

* DO NOT CONNECT

## LOGIC SYMBOL



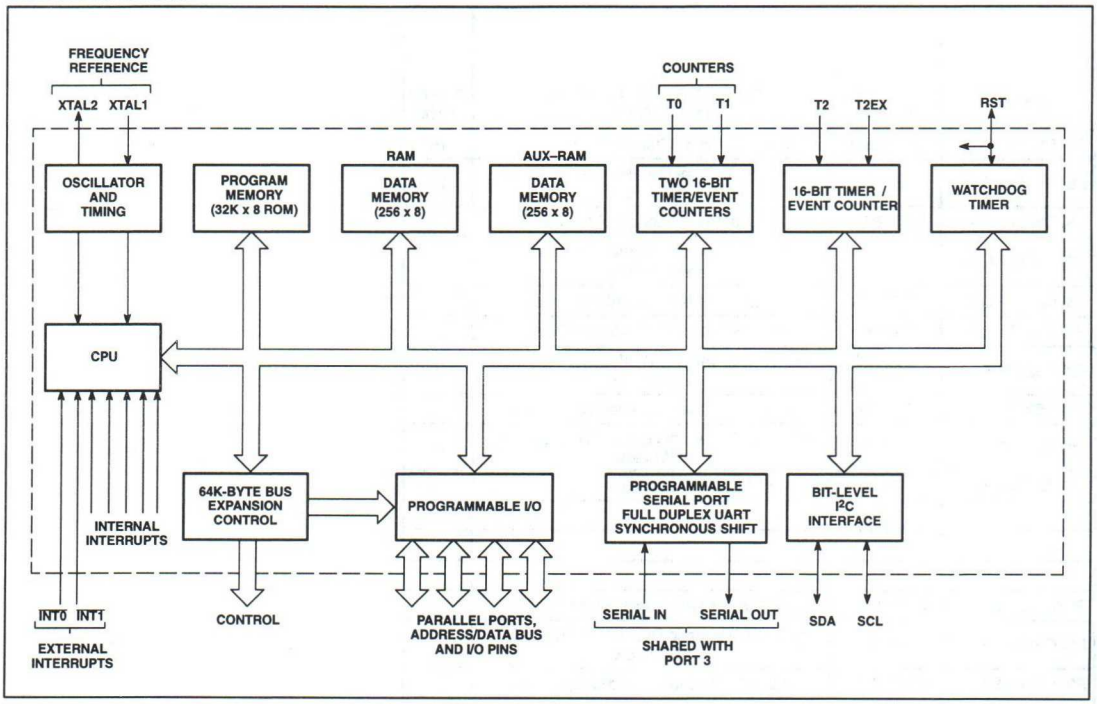
## ORDERING INFORMATION

| PHILIPS PART ORDER NUMBER PART MARKING | | PHILIPS NORTH AMERICA PART ORDER NUMBER | | | | |
|---|---|---|---|---|---|---|
| ROMless | ROM | ROMless | ROM | Drawing Number | TEMPERATURE °C RANGE AND PACKAGE | FREQ MHz |
| P80C528FBP | P83C528FBP/xxx | P80C528FBP N | P83C528FBP N | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 16 |
| P80C528FBA | P83C528FBA/xxx | P80C528FBA A | P83C528FBA A | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| P80C528FBB | P83C528FBB/xxx | P80C528FBB B | P83C528FBB B | SOT307-2 | 0 to +70, Plastic Quad Flat Pack | 16 |
| P80C528FFP | P83C528FFP/xxx | P80C528FFP N | P83C528FFP N | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 16 |
| P80C528FFA | P83C528FFA/xxx | P80C528FFA A | P83C528FFA A | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| P80C528FFB | P83C528FFB/xxx | P80C528FFB B | P83C528FFB B | SOT307-2 | –40 to +85, Plastic Quad Flat Pack | 16 |
| P80C528FHP | P83C528FHP/xxx | P80C528FHP N | P83C528FHP N | SOT129-1 | –40 to +125, Plastic Dual In-line Package | 16 |
| P80C528FHA | P83C528FHA/xxx | P80C528FHA A | P83C528FHA A | SOT187-2 | –40 to +125, Plastic Leaded Chip Carrier | 16 |
| P80C528FHB | P83C528FHB/xxx | P80C528FHB B | P83C528FHB B | SOT307-2 | –40 to +125, Plastic Quad Flat Pack | 16 |
| | P83C528FBR/xxx | | | SOT270-1 | 0 to +70, Plastic Shrink Dual In-Linr Package | 16 |

**NOTE:**
1. xxx denotes the ROM code number.

| EPROM | Drawing Number | TEMPERATURE °C RANGE AND PACKAGE | FREQ MHz |
|---|---|---|---|
| P87C528EBP N | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 16 |
| P87C528EBF FA | 0590B | 0 to +70, Ceramic Dual In-line Package w/Window | 16 |
| P87C528EBA AA | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| P87C528EBL KA | 1472A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 16 |
| P87C528EBB B | SOT307-2 | 0 to +70, Plastic Quad Flat Pack | 16 |
| P87C528EFP N | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 16 |
| P87C528EFF FA | 0590B | –40 to +85, Ceramic Dual In-line Package w/Window | 16 |
| P87C528EFF FA | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| P87C528EFL KA | 1472A | –40 to +85, Ceramic Leaded Chip Carrier w/Window | 16 |
| P87C528EFB B | SOT307-2 | –40 to +85, Plastic Quad Flat Pack | 16 |
| | | | |
| | | | |
| | | | |
| P87C528GBP N | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 20 |
| P87C528GBF FA | 0590B | 0 to +70, Ceramic Dual In-line Package w/Window | 20 |
| P87C528GBA A | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 20 |
| P87C528GBL KA | 1472A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 20 |
| P87C528GFP N | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 20 |
| P87C528GFF FA | 0590B | –40 to +85, Ceramic Dual In-line Package w/Window | 20 |
| P87C528GFA A | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 20 |
| P87C528GFL KA | 1472A | –40 to +85, Ceramic Leaded Chip Carrier w/Window | 20 |
| | | | |

## BLOCK DIAGRAM



## PIN DESCRIPTION

| MNEMONIC | PIN NO. DIP | SDIL | LCC | QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 21 | 22 | 16 | I | **Ground:** circuit ground potential. |
| $V_{DD}$ | 40 | 42 | 44 | 38 | I | **Power Supply:** +5V power supply pin during normal operation, Idle mode and Power-down mode. |
| P0.0–0.7 | 39–32 | 41–34 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. |
| P1.0–P1.7 | 1–8 | 1–8 | 2–9 | 40–44 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups, except P1.6 and P1.7 which have open drain. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 can sink/source one TTL (4 LSTTL) inputs. |
| | 1 | 1 | 2 | 40 | I | **T2 (P1.0):** Timer/counter 2 external count input (following edge triggered). |
| | 2 | 2 | 3 | 41 | I | **T2EX (P1.1):** Timer/counter 2 trigger input. |
| | 7 | 7 | 8 | 2 | I/O | **SCL (P1.6):** $I^2C$ serial port clock line. |
| | 8 | 8 | 9 | 3 | I/O | **SDA (P1.7):** $I^2C$ serial port data line. |
| P2.0–P2.7 | 21–28 | 22–29 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |
| P3.0–P3.7 | 10–17 | 10–18 (11=NC) | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the SC80C51 family, as listed below: |
| | 10 | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 12 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 13 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 14 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 15 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 16 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 17 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 18 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 9 | 10 | 4 | I/O | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{DD}$. After a watchdog timer overflow, this pin is pulled high while the internal reset signal is active. |
| ALE | 30 | 31 | 33 | 27 | I/O | **Address Latch Enable:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. |
| $\overline{PSEN}$ | 29 | 30 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the device is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}$ | 31 | 33 | 35 | 29 | I | **External Access Enable:** $\overline{EA}$ must be externally held low during RESET to enable the device to fetch code from external program memory locations 0000H to 7FFFH. If $\overline{EA}$ is held high during RESET, the device executes from internal program memory unless the program counter contains an address greater than 7FFFH. $\overline{EA}$ is don't care after RESET. |
| XTAL1 | 19 | 20 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 19 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**Table 1. 8XC524/8XC528 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes): | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | ES1 | ET2 | ES0 | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP*# | Interrupt priority | B8H | – | PS1 | PT2 | PS0 | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | SDA | SEL | – | – | – | – | T2EX | T2 | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | RD | WR | T1 | T0 | INT1 | INT0 | TxD | RxD | FFH |
| PCON | Power control | 87H | SMOD | – | – | – | GF1 | GF0 | PD | IDL | 0xxx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | F1 | P | 00H |
| RCAP2H# | Capture high | CBH | | | | | | | | | 00H |
| RCAP2L# | Capture low | CAH | | | | | | | | | 00H |
| SBUF | Serial data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial controller | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| S1BIT# | Serial I²C data | D9H/RD | SDI | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x0000000B |
| | | WR | SD0 | X | X | X | X | X | X | X | 0xxxxxxxB |
| S1INT# | Serial I²C interrupt | DAH | INT | X | X | X | X | X | X | X | 0xxxxxxxB |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| S1SCS*# | Serial I²C control | D8H/RD | SDI | SCI | CLH | BB | RBF | WBF | STR | ENS | xxxx0000B |
| | | WR | SD0 | SC0 | CLH | X | X | X | STR | ENS | 00xxxx00B |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| T2CON*# | Timer 2 control | C8H | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/T2 | CP/RL2 | 00H |
| TH0 | Timer high 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer high 1 | 8DH | | | | | | | | | 00H |
| TH2# | Timer high 2 | CDH | | | | | | | | | 00H |
| TL0 | Timer low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer low 1 | 8BH | | | | | | | | | 00H |
| TL2# | Timer low 2 | CCH | | | | | | | | | 00H |
| T3# | Watchdog timer | FFH | | | | | | | | | 00H |
| TMOD | Timer mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |
| WDCON# | Watchdog control | A5H | | | | | | | | | A5H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

**Table 2. Internal and External Program Memory Access with Security Bit Set**

| INSTRUCTION | ACCESS TO INTERNAL PROGRAM MEMORY | ACCESS TO EXTERNAL PROGRAM MEMORY |
|---|---|---|
| MOVC in internal program memory | YES | YES |
| MOVC in external program memory | NO | YES |

## ROM CODE PROTECTION

By setting a mask programmable security bit, the ROM content in the 83C528 is protected, i.e., it cannot be read out by any test mode or by any instruction in the external program memory space. The MOVC instructions are the only ones which have access to program code in the internal or external program memory. The $\overline{EA}$ input is latched during RESET and is 'don't care' after RESET (also if security bit is not set). This implementation prevents reading from internal program code by switching from external program memory to internal program memory during MOVC instruction or an instruction that handles immediate data. Table 2 lists the access to the internal and external program memory by the MOVC instructions when the security bit has been set to logical one. If the security bit has been set to a logical 0 there are no restrictions for the MOVC instructions.

## INTERNAL DATA MEMORY

The internal data memory is divided into three physically separated segments: 256 bytes of RAM, 256 bytes of AUX-RAM, and a 128 bytes special function area. These can be addressed each in a different way.

- RAM 0 to 127 can be addressed directly and indirectly as in the 80C51. Address pointers are R0 and R1 of the selected register bank.
- RAM 128 to 255 can only be addressed indirectly as in the 80C51. Address pointers are R0 and R1 of the selected register bank.
- AUX-RAM 0 to 255 is indirectly addressed in the same way as external data memory with the MOVX instructions. Address pointers are R0, R1 of the selected register bank and DPTR. An access to AUX-RAM 0 to 255 will not affect ports P0, P2, P3.6 and P3.7.

An access to external data memory locations higher than 255 will be performed with the MOVX DPTR instructions in the same way as in the 8051 structure, so with P0 and P2 as data/address bus and P3.6 and P3.7 as write and read timing signals. Note that these external data memory cannot be accessed with R0 and R1 as address pointer.

## TIMER 2

Timer 2 is functionally equal to the Timer 2 of the 8052AH. Timer 2 is a 16-bit timer/counter. These 16 bits are formed by two special function registers TL2 and TH2. Another pair of special function register RCAP2L and RCAP2H form a 16-bit capture register or a 16-bit reload register. Like Timer 0 and 1, it can operate either as a timer or as an event counter. This is selected by bit C/T2N in the special function register T2CON. It has three operating modes: capture, autoload, and baud rate generator mode which are selected by bits in T2CON.

## WATCHDOG TIMER T3

The watchdog timer consists of an 11-bit prescaler and an 8-bit timer formed by special function register T3. The prescaler is incremented by an on-chip oscillator with a fixed frequency of 1MHz. The maximum tolerance on this frequency is –50% and +100%. The 8-bit timer increments every 2048 cycles of the on-chip oscillator. When a timer overflow occurs, the microcontroller is reset and a reset output pulse of 16 × 2048 cycles of the on-chip oscillator is generated at pin RST. The internal RESET signal is not inhibited when the external RST pin is kept low by, for example, an external reset circuit. The RESET signal drives port 1, 2, 3 into the high state and port 0 into the high impedance state.

The watchdog timer is controlled by one special function register WDCON with the direct address location A5H. WDCON can be read and written by software. A value of A5H in WDCON halts the on-chip oscillator and clears both the prescaler and timer T3. After the RESET signal, WDCON contains A5H. Every value other than A5H in WDCON enables the watchdog timer. When the watchdog timer is enabled, it runs independently of the XTAL-clock.

Timer T3 can be read on the fly. Timer T3 can only be written if WDCON contains the value 5AH. A successful write operation to T3 will clear the prescaler and WDCON, leaving the watchdog enabled and preventing inadvertent changes of T3. To prevent an overflow of the watchdog timer, the user program has to reload the watchdog timer within periods that are shorter than the programmed watchdog timer internal. This time interval is determined by an 8-bit value that has to be loaded in register T3 while at the same time the prescaler is cleared by hardware.

Watchdog timer interval =

$$\frac{[256 - (T3)] \times 2048}{\text{on - chip oscillator frequency}}$$

## BIT-LEVEL I²C INTERFACE

This bit-level serial I/O interface supports the I²C-bus. P1.6/SCL and P1.7/SDA are the serial I/O pins. These two pins meet the I²C specification concerning the input levels and output drive capability. Consequently, these pins have an open drain output configuration. All the four modes of the I²C-bus are supported:

- master transmitter
- master receiver
- slave transmitter
- slave receiver

The advantages of the bit-level I²C hardware compared with a full software I²C implementation are:

- the hardware can generate the SCL pulse
- Testing a single bit (RBF respectively, WBF) is sufficient as a check for error free transmission.

The bit-level I²C hardware operates on serial bit level and performs the following functions:

- filtering the incoming serial data and clock signals
- recognizing the START condition
- generating a serial interrupt request SI after reception of a START condition and the first falling edge of the serial clock
- recognizing the STOP condition
- recognizing a serial clock pulse on the SCL line
- latching a serial bit on the SDA line (SDI)
- stretching the SCL LOW period of the serial clock to suspend the transfer of the next serial data bit

- setting Read Bit Finished (RBF) when the SCL clock pulse has finished and Write Bit Finished (WBF) if there is no arbitration loss detected (i.e., SDA = 0 while SDO = 1)
- setting a serial clock Low-to-High detected (CLH) flag
- setting a Bus Busy (BB) flag on a START condition and clearing this flag on a STOP condition
- releasing the SCL line and clearing the CLH, RBF and WBF flags to resume transfer of the next serial data bit
- generating an automatic clock if the single bit data register S1BIT is used in master mode.

The following functions must be done in software:
- handling the $I^2C$ START interrupts
- converting serial to parallel data when receiving
- converting parallel to serial data when transmitting
- comparing the received slave address with its own
- interpreting the acknowledge information
- guarding the $I^2C$ status if RBF or WBF = 0.

Additionally, if acting as master:
- generating START and STOP conditions
- handling bus arbitration
- generating serial clock pulses if S1BIT is not used.

Three SFRs control the bit-level $I^2C$ interface: S1INT, S1BIT and S1SCS.

## INTERRUPT SYSTEM

The interrupt structure of the 8XC528 is the same as that used in the 80C51, but includes two additional interrupt sources: one for the third timer/counter, T2, and one for the $I^2C$ interface. The interrupt enable and interrupt priority registers are IE and IP.

### IE: Interrupt Enable Register

This register is located at address A8H. Refer to Table 3.

**IE SFR (A8H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| EA | ES1 | ET2 | ES | ET1 | EX1 | ET0 | EX0 |

### IP: Interrupt Priority Register

This register is located at address B8H. Refer to Table 4.

**IP SFR (B8H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | PS1 | PT2 | PS | PT1 | PX1 | PT0 | PX0 |

The interrupt vector locations and the interrupt priorities are:

| Source Vector | Address | Priority within Level |
|---|---|---|
| 0003H | IE0 | Highest |
| 002BH | TF2+EXF2 | |
| 0053H | SI ($I^2C$) | |
| 000BH | TF0 | |
| 0013H | IE1 | |
| 001BH | TF1 | |
| 0023H | R1+T1 | Lowest |

**Table 3. Description of IE Bits**

| MNEMONIC | BIT | FUNCTION |
|---|---|---|
| EA | IE.7 | General enable/disable control:<br>0 = **NO** interrupt is enabled.<br>1 = **ANY** individually enabled interrupt will be accepted. |
| ES1 | IE.6 | Enable **bit-level $I_2C$ I/O** interrupt |
| ET2 | IE.5 | Enable **Timer 2** interrupt |
| ES | IE.4 | Enable **Serial Port** interrupt |
| ET1 | IE.3 | Enable **Timer 1** interrupt |
| EX1 | IE.2 | Enable **External** interrupt 1 |
| ET0 | IE.1 | Enable **Timer 0** interrupt |
| EX0 | IE.0 | Enable **External** interrupt 0 |

**Table 4. Description of IP Bits**

| MNEMONIC | BIT | FUNCTION |
|---|---|---|
| – | IP.7 | **Reserved.** |
| PS1 | IP.6 | **Bit-level $I^2C$** interrupt priority level |
| PT2 | IP.5 | **Timer 2** interrupt priority level |
| PS | IP.4 | **Serial Port** interrupt priority level |
| PT1 | IP.3 | **Timer 1** interrupt priority level |
| PX1 | IP.2 | **External Interrupt 1** priority level |
| PT0 | IP.1 | **Timer 0** interrupt priority level |
| PX0 | IP.0 | **External Interrupt 0** priority level |

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the Logic Symbol.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-up reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-up, the voltage on $V_{DD}$ and RST must come up at the same time for a proper start-up.

## IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. The power-down mode can be terminated by a RESET in the same way as in the 80C51 or in addition by one of two external interrupts, INT0 or INT1. A termination with an external interrupt does not affect the internal data memory and does not affect the special function registers. This makes it possible to exit power-down without changing the port output levels. To terminate the power-down mode with an external interrupt $\overline{INT0}$ or $\overline{INT1}$ must be switched to level-sensitive and must be enabled. The external interrupt input signal $\overline{INT0}$ and $\overline{INT1}$ must be kept low until the oscillator has restarted and stabilized. An instruction following the instruction that puts the device in the power-down mode will be executed. A reset generated by the watchdog timer terminates the power-down mode in the same way as an external RESET, and only the contents of the on-chip RAM are preserved. The control bits for the reduced power modes are in the special function register PCON.

## DESIGN CONSIDERATIONS

At power-on, the voltage on $V_{DD}$ and RST must come up at the same time for a proper start-up.

When the idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when idle is terminated by reset, the instruction following the one that invokes idle should not be one that writes to a port pin or to external memory.

Table 5 shows the state of I/O ports during low current operating modes.

**Table 5. External Pin Status During Idle and Power-Down Modes**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | 0 to +70, or –40 to +85, or –40 to +125 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on any other pin to $V_{SS}$ | –0.5 to $V_{DD}$ +0.5 | V |
| Input, output current on any two pins | ±10 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.0 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C ($V_{DD}$ = 5V ±20%), –40°C to +85°C ($V_{DD}$ = 5V ±20%), or –40°C to +125°C ($V_{DD}$ = 5V ±10%), $V_{SS}$=0V

| SYMBOL | PARAMETER | PART TYPE | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input low voltage, except $\overline{EA}$, P1.6/SCL, P1.7/SDA | 0°C to 70°C<br>–40°C to +85°C<br>–40°C to +125°C | | –0.5<br>–0.5<br>–0.5 | $0.2V_{DD}$–0.1<br>$0.2V_{DD}$–0.15<br>$0.2V_{DD}$–0.25 | V<br>V<br>V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | 0°C to 70°C<br>–40°C to +85°C<br>–40°C to +125°C | | –0.5<br>–0.5<br>–0.5 | $0.2V_{DD}$–0.3<br>$0.2V_{DD}$–0.35<br>$0.2V_{DD}$–0.45 | V<br>V<br>V |
| $V_{IL2}$ | Input low voltage to P1.6/SCL, P1.7/SDA[3] | | | –0.5 | $0.3V_{DD}$ | V |
| $V_{IH}$ | Input high voltage, except XTAL1, $\overline{RST}$, P1.6/SCL, P1.7/SDA | 0°C to 70°C<br>–40°C to +85°C<br>–40°C to +125°C | | $0.2V_{DD}$+0.9<br>$0.2V_{DD}$+1.0<br>$0.2V_{DD}$+1.0 | $V_{DD}$+0.5<br>$V_{DD}$+0.5<br>$V_{DD}$+0.5 | V<br>V<br>V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | 0°C to 70°C<br>–40°C to +85°C<br>–40°C to +125°C | | $0.7V_{DD}$<br>$0.7V_{DD}$+0.1<br>$0.7V_{DD}$+0.1 | $V_{DD}$+0.5<br>$V_{DD}$+0.5<br>$V_{DD}$+0.5 | V<br>V<br>V |
| $V_{IH2}$ | Input high voltage, P1.6/SCL, P1.7/SDA[3] | | | $0.7V_{DD}$ | 6.0 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3, except P1.6/SCL, P1.7/SDA[1] | | $I_{OL}$ = 1.6mA[4] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$[1] | | $I_{OL}$ = 3.2mA[4] | | 0.45 | V |
| $V_{OL2}$ | Output low voltage, P1.6/SCL, P1.7/SDA | | $I_{OL}$ = 3.0mA[4] | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3 | | $V_{DD}$ = 5V ±10%,<br>$I_{OH}$ = –60µA<br>$I_{OH}$ = –25µA<br>$I_{OH}$ = –10µA | <br>2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | <br>V<br>V<br>V |
| $V_{OH1}$ | Output high voltage, Port 0 in external bus mode, ALE, $\overline{PSEN}$, RST[2] | | $V_{DD}$ = 5V ±10%,<br>$I_{OH}$ = –800µA<br>$I_{OH}$ = –300µA<br>$I_{OH}$ = –80µA | <br>2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | <br>V<br>V<br>V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3, except P1.6/SCL, P1.7/SDA | 0°C to 70°C<br>–40°C to +85°C<br>–40°C to +125°C | $V_{IN}$ = 0.45V | | –50<br>–75<br>–75 | µA<br>µA<br>µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3, except P1.6/SCL, P1.7/SDA | 0°C to 70°C<br>–40°C to +85°C<br>–40°C to +125°C | See note 5 | | –650<br>–750<br>–750 | µA<br>µA<br>µA |

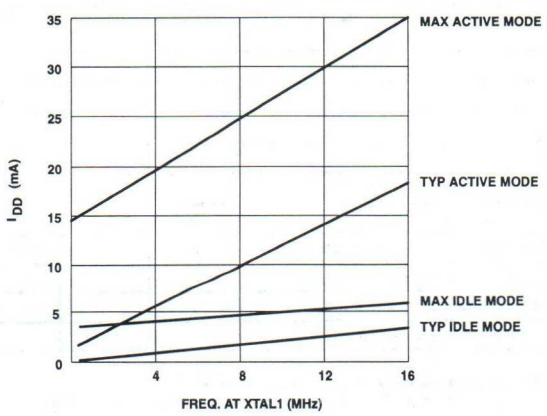
## DC ELECTRICAL CHARACTERISTICS (Continued)

$T_{amb}$ = 0°C to +70°C ($V_{DD}$ = 5V ±20%), –40°C to +85°C ($V_{DD}$ = 5V ±20%), or –40°C to +125°C ($V_{DD}$ = 5V ±10%), $V_{SS}$=0V

| SYMBOL | PARAMETER | PART TYPE | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|---|
| $I_{IL1}$ | Input leakage current, port 0, $\overline{EA}$ | | 0.45<Vi<$V_{DD}$ | | ±10 | µA |
| $I_{IL2}$ | Input leakage current, P1.6/SCL, P1.7/SDA | | 0V<Vi<6.0V<br>0V<$V_{DD}$<6.0V | | ±10 | µA<br>µA |
| $I_{DD}$ | Power supply current:<br>Active mode<br>Idle mode<br>Power down mode<br>Power down mode | <br><br><br><br>–40°C to +125°C | See notes 6, 7 | <br><br><br><br> | <br>35<br>6<br>100<br>150 | <br>mA<br>mA<br>µA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | | 50 | 150 | kΩ |
| $C_{IO}$ | Capacitance of I/O buffer | | Freq.=1MHz<br>$T_{amb}$ = 25°C | | 10 | pF |

**NOTES:**

1. Capacitive loading on Port 0 and Port 2 may cause spurious noise pulses to be superimposed on the LOW level ouput voltage of ALE, Port 1 and Port 3. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make a 1-to-0 transition during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE line may exceed 0.8V. In such cases it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.
2. Capacitive loading on Port 0 and Port 2 may cause the HIGH level output voltage on ALE and $\overline{PSEN}$ to momentarily fall below the 0.9$V_{DD}$ specification when the address bits are stabilizing.
3. The input threshold voltage of P1.6 and P1.7 (SIO1) meets the $I^2C$ specification, so a voltage below 0.3$V_{DD}$ will be recognized as a logic 0 while an input above 0.7$V_{DD}$ will be recognized as a logic 1.
4. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 10mA
   Maximum $I_{OL}$ per 8–bit port: –
   Port 0: 26mA
   Ports 1, 2, & 3: 15mA
   Maximum total $I_{OL}$ for all output pins: 71mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
5. Pins of ports 1, 2, and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
6. See Figures 9 through 12 for $I_{DD}$ test conditions.
7. $I_{DDMAX}$ at other frequencies can be derived from the figure below, where FREQ is the external oscillator frequency in MHz. $I_{DDMAX}$ is given in mA.




VALID ONLY WITHIN FREQUENCY SPECIFICATIONS OF DEVICE UNDER TEST.

**$I_{DD}$ vs. FREQUENCY**

## AC ELECTRICAL CHARACTERISTICS[1, 2]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 1 | Oscillator frequency | | | 1.2 | 16 | MHz |
| $t_{LHLL}$ | 1 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 1 | Address valid to ALE low | 8 | | $t_{CLCL}-55$ | | ns |
| $t_{LLAX}$ | 1 | Address hold after ALE low | 28 | | $t_{CLCL}-35$ | | ns |
| $t_{LLIV}$ | 1 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 1 | ALE low to $\overline{PSEN}$ low | 23 | | $t_{CLCL}-40$ | | ns |
| $t_{PLPH}$ | 1 | $\overline{PSEN}$ pulse width | 143 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 1 | $\overline{PSEN}$ low to valid instruction in | | 83 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 1 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 1 | Input instruction float after $\overline{PSEN}$ | | 38 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 1 | Address to valid instruction in | | 208 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 1 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 2, 3 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 2, 3 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 2, 3 | $\overline{RD}$ low to valid data in | | 148 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 2, 3 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 2, 3 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-70$ | ns |
| $t_{LLDV}$ | 2, 3 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 2, 3 | Address to valid data in | | 398 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 2, 3 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 138 | 238 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 2, 3 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 120 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 2, 3 | Data valid to $\overline{WR}$ transition | 3 | | $t_{CLCL}-60$ | | ns |
| $t_{WHQX}$ | 2, 3 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{RLAZ}$ | 2, 3 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 2, 3 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 6 | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | 6 | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | 6 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 6 | Fall time | | 20 | | 20 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 4 | Serial port clock cycle time | 750 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 4 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 4 | Output data hold after clock rising edge | 8 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 4 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 4 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}-133$ | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.

## AC ELECTRICAL CHARACTERISTICS – I²C INTERFACE

| SYMBOL | PARAMETER | INPUT | OUTPUT | I²C SPECIFICATION |
|---|---|---|---|---|
| **SCL TIMING CHARACTERISTICS** | | | | |
| $t_{HD;STA}$ | START condition hold time | ≥ 14 $t_{CLCL}$[1] | Note 2 | ≥ 4.0µs |
| $t_{LOW}$ | SCL LOW time | ≥ 16 $t_{CLCL}$ | Note 2 | ≥ 4.7µs |
| $t_{HIGH}$ | SCL HIGH time | ≥ 14 $t_{CLCL}$[1] | ≥ 80 $t_{CLCL}$[3] | ≥ 4.0µs |
| $t_{RC}$ | SCL rise time | ≤ 1µs[4] | Note 5 | ≤ 1.0µs |
| $t_{FC}$ | SCL fall time | ≤ 0.3µs[4] | ≤ 0.3µs [6] | ≤ 0.3µs |
| **SDA TIMING CHARACTERISTICS** | | | | |
| $t_{SU;DAT1}$ | Data set-up time | ≥ 250ns | Note 2 | ≥ 250ns |
| $t_{HD;DAT}$ | Data hold time | ≥ 0ns | Note 2 | ≥ 0ns |
| $t_{SU;STA}$ | Repeated START set-up time | ≥ 14 $t_{CLCL}$[1] | Note 2 | ≥ 4.7µs |
| $t_{SU;STO}$ | STOP condition set-up time | ≥ 14 $t_{CLCL}$[1] | Note 2 | ≥ 4.0µs |
| $t_{BUF}$ | Bus free time | ≥ 14 $t_{CLCL}$[1] | Note 2 | ≥ 4.7µs |
| $t_{RD}$ | SDA rise time | ≤ 1µs[4] | Note 5 | ≤ 1.0µs |
| $t_{FD}$ | SDA fall time | ≤ 0.3µs[4] | ≤ 0.3µs [6] | ≤ 0.3µs |

**NOTES:**
1. At $f_{CLK}$ = 3.5MHz, this evaluates to 14 × 286ns = 4µs, i.e., the bit-level I²C interface can respond to the I²C protocol for $f_{CLK}$ ≥ 3.5MHz.
2. This parameter is determined by the user software, it has to comply with the I²C.
3. This value gives the autoclock pulse length which meets the I²C specification for the specified XTAL clock frequency range. Alternatively, the SCL pulse may be timed by software.
4. Spikes on SDA and SCL lines with a duration of less than 4 × $f_{CLK}$ will be filtered out.
5. The rise time is determined by the external bus line capacitance and pull-up resistor, it must be ≤ 1µs.
6. The maximum capacitance on bus lines SDA and SCL is 400pF.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{\text{PSEN}}$
Q – Output data
R – $\overline{\text{RD}}$ signal
t – Time
V – Valid
W – $\overline{\text{WR}}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{\text{PSEN}}$ low.

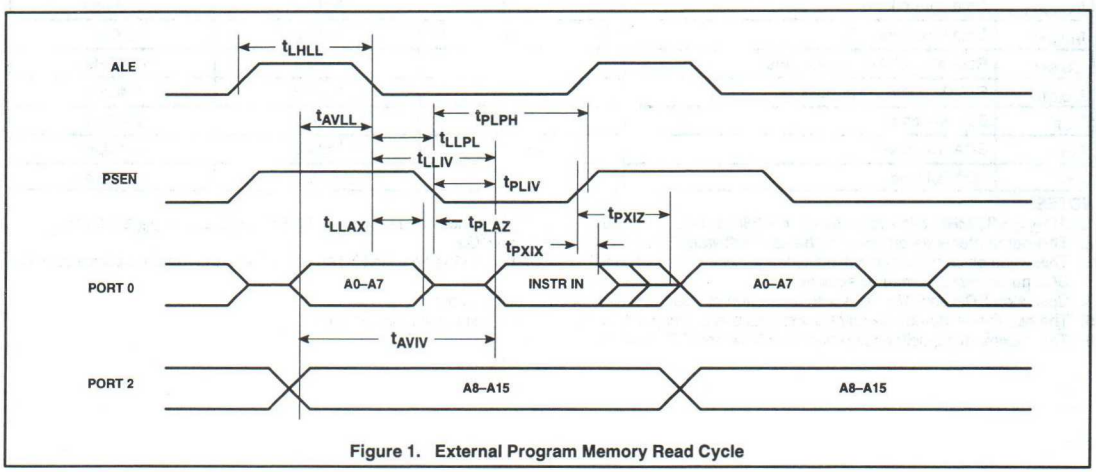



**Figure 1. External Program Memory Read Cycle**

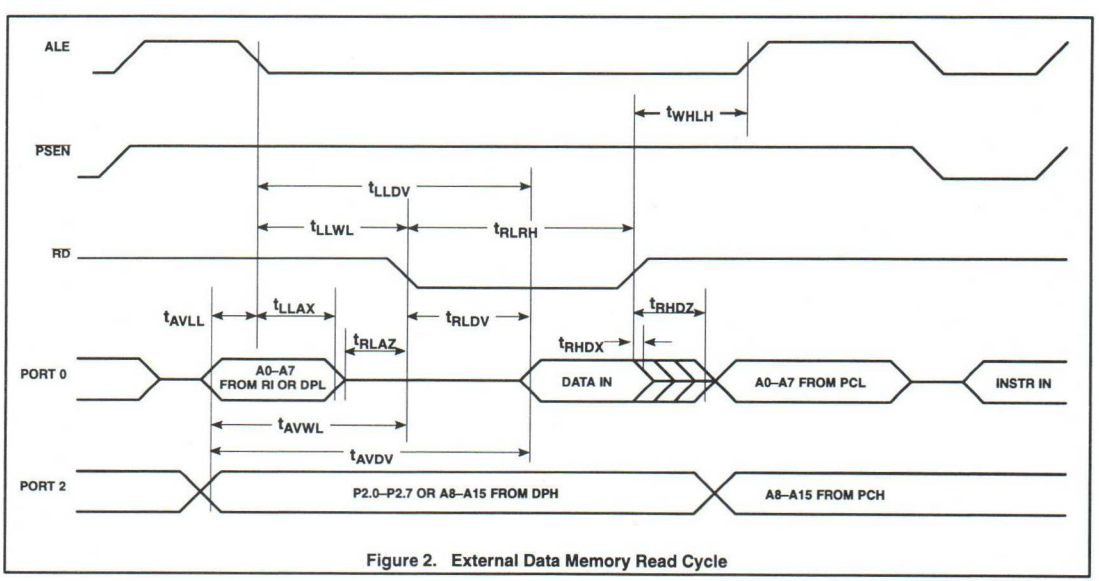



**Figure 2. External Data Memory Read Cycle**

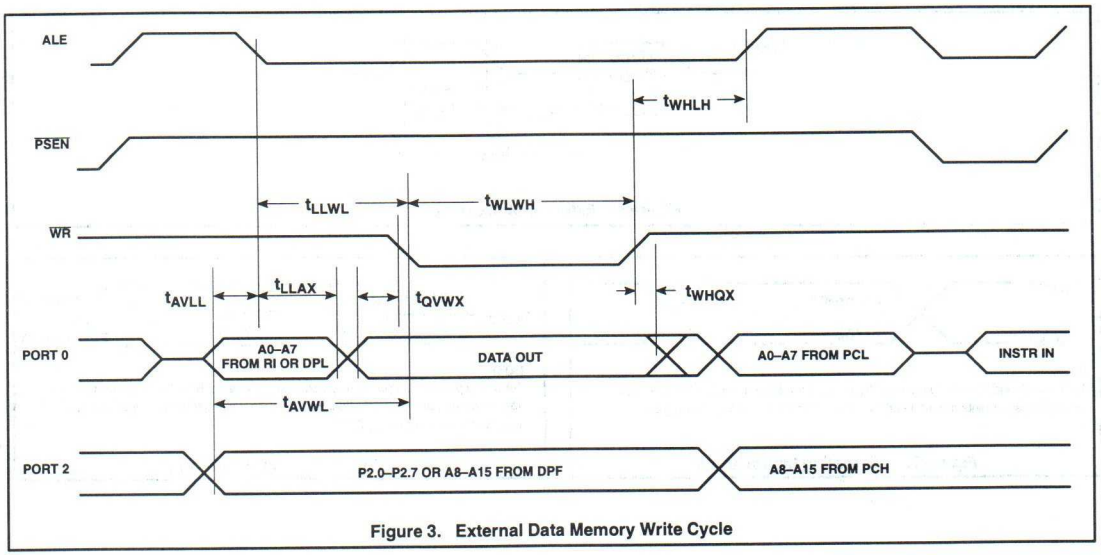



**Figure 3. External Data Memory Write Cycle**

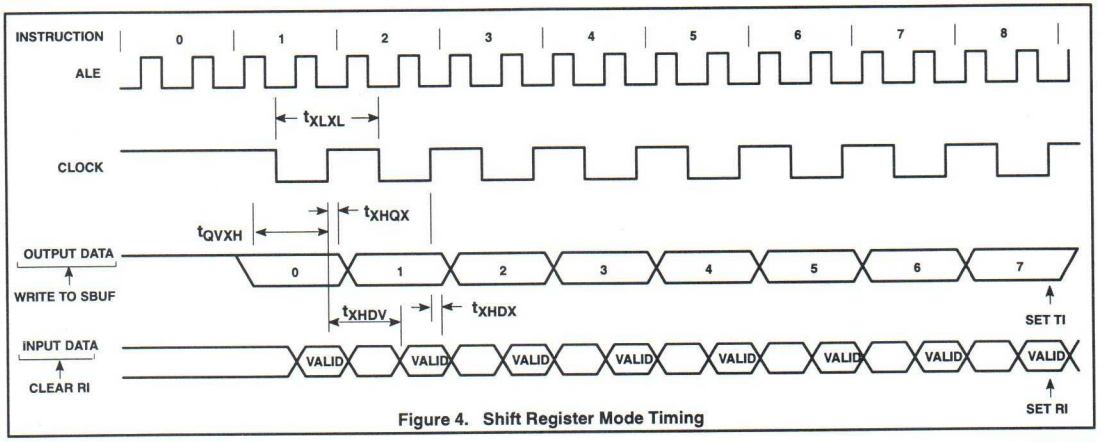



**Figure 4. Shift Register Mode Timing**

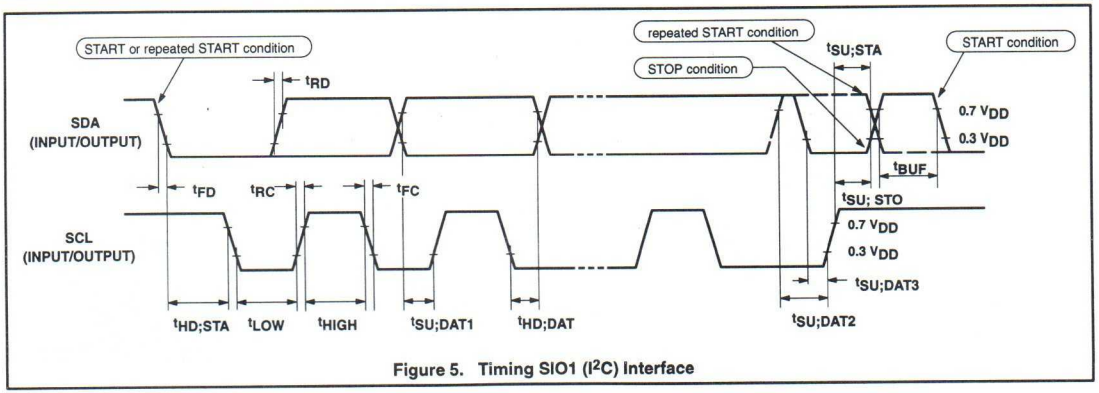



**Figure 5. Timing SIO1 ($I^2C$) Interface**

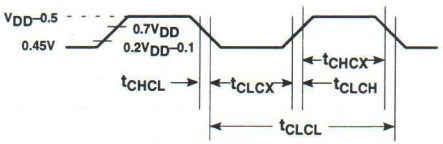
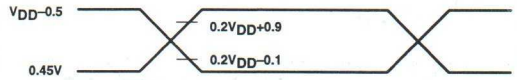
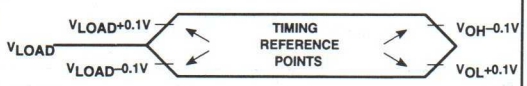



**Figure 6. External Clock Drive**




**NOTE:**
AC inputs during testing are driven at $V_{DD}$ –0.5 for a logic '1' and 0.45V for a logic '0'. Timing measurements are made at $V_{IH}$ min for a logic '1' and $V_{IL}$ for a logic '0'.

**Figure 7. AC Testing Input/Output**




**NOTE:**
For timing purposes, a port is no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded $V_{OH}$/$V_{OL}$ level occurs. $I_{OH}/I_{OL} \geq \pm 20mA$.

**Figure 8. Float Waveform**

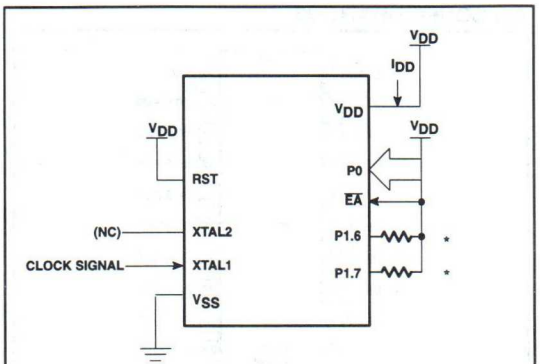



**Figure 9. $I_{DD}$ Test Condition, Active Mode**
**All other pins are disconnected**

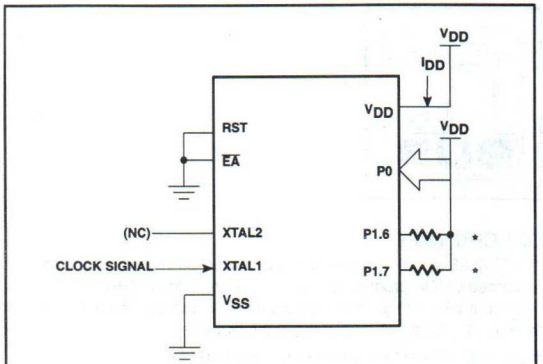



**Figure 10. $I_{DD}$ Test Condition, Idle Mode**
**All other pins are disconnected**

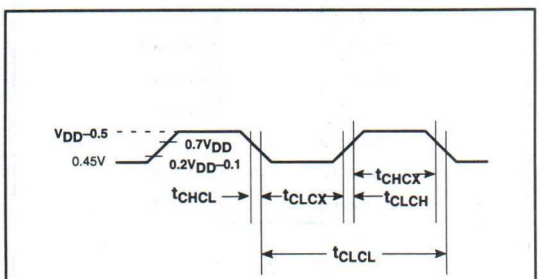



**Figure 11. Clock Signal Waveform for $I_{DD}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

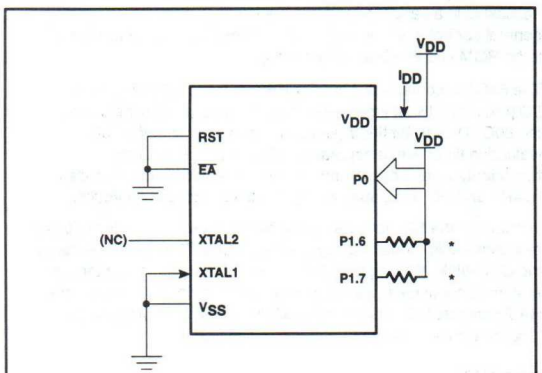



**Figure 12. $I_{DD}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{DD}$ = 2V to 5.5V**

**NOTE:**
* Ports 1.6 and 1.6 should be connected to $V_{DD}$ through resistors of sufficiently high value such that the sink current into these pins does not exceed the $I_{OL1}$ specifications.

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# CMOS single-chip 8-bit microcontroller 87C528

## DESCRIPTION

The 87C528 single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 87C528 has the same instruction set as the 80C51. Three versions of the derivative exist:

- 83C528—32k bytes mask programmable ROM
- 80C528—ROMless version of the 83C528
- 87C528—32k bytes EPROM

This device provides architectural enhancements that make it applicable in a variety of applications in consumer, telecom and general control systems, especially in those systems which need large ROM and RAM capacity on-chip.

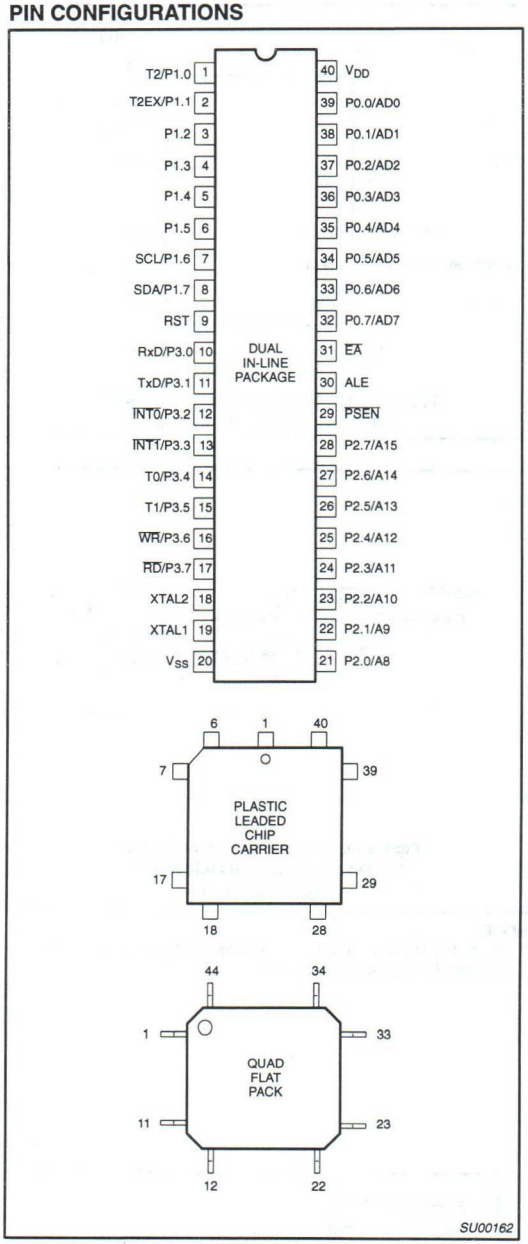
The 87C528 contains a 32k × 8 EPROM, a 512 × 8 RAM, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a 16-bit timer (identical to the timer 2 of the 80C52), a watchdog timer with a separate oscillator, a multi-source, two-priority-level, nested interrupt structure, two serial interfaces (UART and I²C-bus), and on-chip oscillator and timing circuits.

In addition, the 87C528 has two software selectable modes of power reduction—idle mode and power-down mode. The idle mode freezes the CPU while allowing the RAM, timers, serial port, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

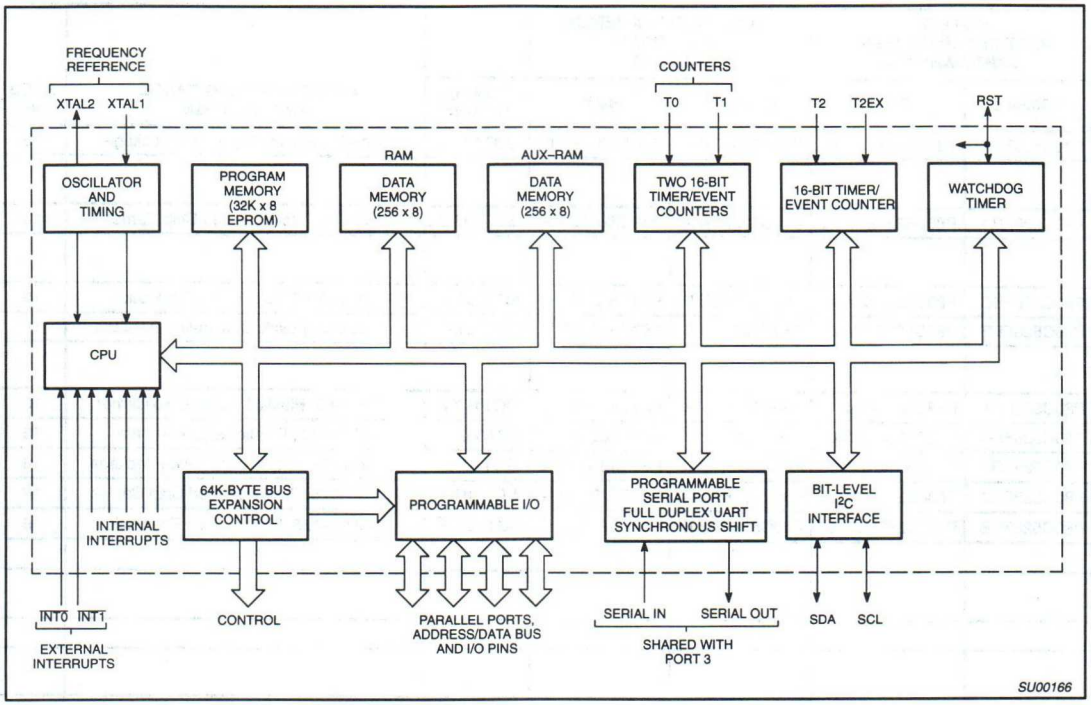
## FEATURES

- 80C51 instruction set
  - 32k × 8 EPROM
  - 512 × 8 RAM
  - Memory addressing capability
    64k ROM and 64k RAM
  - Three 16-bit counter/timers
  - On-chip watchdog timer with oscillator
  - Full duplex UART
  - I²C serial interface
- Power control modes:
  - Idle mode
  - Power-down mode
  - Warm start from power-down
- CMOS and TTL compatible
- Extended temperature ranges
- EPROM code protection
- OTP package available
- Two speed ranges at $V_{CC}$ = 5V
  - 16MHz
  - 20MHz

## PIN CONFIGURATIONS



## BLOCK DIAGRAM



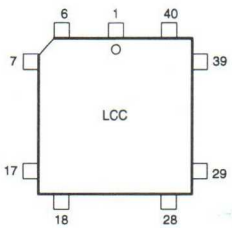
## ORDERING INFORMATION

| PHILIPS PART ORDER NUMBER PART MARKING | | PHILIPS NORTH AMERICA PART ORDER NUMBER | | | | |
|---|---|---|---|---|---|---|
| ROMless | ROM | ROMless | ROM | Drawing Number | TEMPERATURE °C RANGE AND PACKAGE | FREQ MHz |
| P80C528FBP | P83C528FBP/xxx | P80C528FBP N | P83C528FBP N | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 16 |
| | | | | | | |
| P80C528FBA | P83C528FBA/xxx | P80C528FBA A | P83C528FBA A | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| | | | | | | |
| P80C528FBB | P83C528FBB/xxx | P80C528FBB B | P83C528FBB B | SOT307-2[2] | 0 to +70, Plastic Quad Flat Pack | 16 |
| P80C528FFP | P83C528FFP/xxx | P80C528FFP N | P83C528FFP N | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 16 |
| | | | | | | |
| P80C528FFA | P83C528FFA/xxx | P80C528FFA A | P83C528FFA A | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| P80C528FFB | P83C528FFB/xxx | P80C528FFB B | P83C528FFB B | SOT307-2[2] | –40 to +85, Plastic Quad Flat Pack | 16 |
| P80C528FHP | P83C528FHP/xxx | P80C528FHP N | P83C528FHP N | SOT129-1 | –40 to +125, Plastic Dual In-line Package | 16 |
| P80C528FHA | P83C528FHA/xxx | P80C528FHA A | P83C528FHA A | SOT187-2 | –40 to +125, Plastic Leaded Chip Carrier | 16 |
| P80C528FHB | P83C528FHB/xxx | P80C528FHB B | P83C528FHB B | SOT307-2[2] | –40 to +125, Plastic Quad Flat Pack | 16 |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

**NOTE:**
1. xxx denotes the ROM code number.
2. SOT311 replaced by SOT307-2.

| EPROM | Drawing Number | TEMPERATURE °C RANGE AND PACKAGE | FREQ MHz |
|---|---|---|---|
| P87C528EBP N | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 16 |
| P87C528EBF FA | 0590B | 0 to +70, Ceramic Dual In-line Package w/Window | 16 |
| P87C528EBA A | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| P87C528EBL KA | 1472A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 16 |
| P87C528EBB B | SOT307-2[2] | 0 to +70, Plastic Quad Flat Pack | 16 |
| P87C528EFP N | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 16 |
| P87C528EFF FA | 0590B | –40 to +85, Ceramic Dual In-line Package w/Window | 16 |
| P87C528EFA A | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| P87C528EFB B | SOT307-2[2] | –40 to +85, Plastic Quad Flat Pack | 16 |
| | | | |
| | | | |
| | | | |
| P87C528GBP N | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 20 |
| P87C528GBF FA | 0590B | 0 to +70, Ceramic Dual In-line Package w/Window | 20 |
| P87C528GBA A | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 20 |
| P87C528GBL KA | 1472A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 20 |
| P87C528GFP N | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 20 |
| P87C528GFF FA | 0590B | –40 to +85, Ceramic Dual In-line Package w/Window | 20 |
| P87C528GFA A | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 20 |

## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

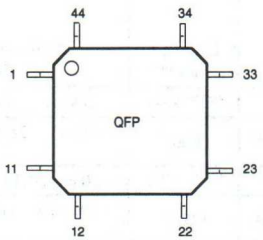



| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC* | 23 | NC* |
| 2 | P1.0/T2 | 24 | P2.0/A8 |
| 3 | P1.1/T2EX | 25 | P2.1/A9 |
| 4 | P1.2 | 26 | P2.2/A10 |
| 5 | P1.3 | 27 | P2.3/A11 |
| 6 | P1.4 | 28 | P2.4/A12 |
| 7 | P1.5 | 29 | P2.5/A13 |
| 8 | P1.6/SCL | 30 | P2.6/A14 |
| 9 | P1.7/SDA | 31 | P2.7/A15 |
| 10 | RST | 32 | $\overline{PSEN}$ |
| 11 | P3.0/RxD | 33 | ALE |
| 12 | NC* | 34 | NC* |
| 13 | P3.1/TxD | 35 | $\overline{EA}$ |
| 14 | P3.2/$\overline{INT0}$ | 36 | P0.7/AD7 |
| 15 | P3.3/$\overline{INT1}$ | 37 | P0.6/AD6 |
| 16 | P3.4/T0 | 38 | P0.5/AD5 |
| 17 | P3.5/T1 | 39 | P0.4/AD4 |
| 18 | P3.6/$\overline{WR}$ | 40 | P0.3/AD3 |
| 19 | P3.7/$\overline{RD}$ | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | $V_{SS}$ | 44 | $V_{DD}$ |

* DO NOT CONNECT

SU00163A

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS

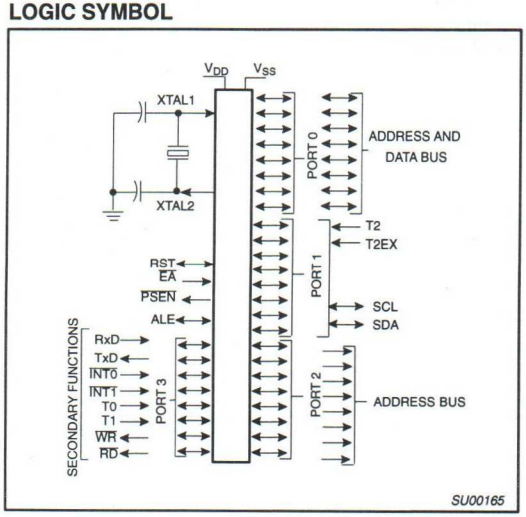



| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5 | 23 | P2.5/A13 |
| 2 | P1.6/SCL | 24 | P2.6/A14 |
| 3 | P1.7/SDA | 25 | P2.7/A15 |
| 4 | RST | 26 | $\overline{PSEN}$ |
| 5 | P3.0/RxD | 27 | ALE |
| 6 | NC* | 28 | NC* |
| 7 | P3.1/TxD | 29 | $\overline{EA}$ |
| 8 | P3.2/$\overline{INT0}$ | 30 | P0.7/AD7 |
| 9 | P3.3/$\overline{INT1}$ | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/$\overline{WR}$ | 34 | P0.3/AD3 |
| 13 | P3.7/$\overline{RD}$ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | $V_{SS}$ | 38 | $V_{DD}$ |
| 17 | NC* | 39 | NC* |
| 18 | P2.0/A8 | 40 | P1.0/T2 |
| 19 | P2.1/A9 | 41 | P1.1/T2EX |
| 20 | P2.2/A10 | 42 | P1.2 |
| 21 | P2.3/A11 | 43 | P1.3 |
| 22 | P2.4/A12 | 44 | P1.4 |

* DO NOT CONNECT

SU00164

## LOGIC SYMBOL




SU00165

## PIN DESCRIPTION

| MNEMONIC | PIN NO. DIP | PIN NO. LCC | PIN NO. QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** circuit ground potential. |
| $V_{DD}$ | 40 | 44 | 38 | I | **Power Supply:** +5V power supply pin during normal operation, Idle mode and Power-down mode. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups, except P1.6 and P1.7 which have open drain. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 can sink/source one TTL (4 LSTTL) inputs. Port 1 receives the low-order address byte during program memory verification. Port 1 also serves alternate functions for timer 2: |
| | 1 | 2 | 40 | I | **T2 (P1.0):** Timer/counter 2 external count input (following edge triggered). |
| | 2 | 3 | 41 | I | **T2EX (P1.1):** Timer/counter 2 trigger input. |
| | 7 | 8 | 2 | I/O | **SCL (P1.6):** $I^2C$ serial port clock line. |
| | 8 | 9 | 3 | I/O | **SDA (P1.7):** $I^2C$ serial port data line. |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the SC80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I/O | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{DD}$. After a watchdog timer overflow, this pin is pulled high while the internal reset signal is active. |
| ALE | 30 | 33 | 27 | I/O | **Address Latch Enable:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the device is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}$ | 31 | 35 | 29 | I | **External Access Enable:** $\overline{EA}$ must be externally held low during RESET to enable the device to fetch code from external program memory locations 0000H to 7FFFH. If $\overline{EA}$ is held high during RESET, the device executes from internal program memory unless the program counter contains an address greater than 7FFFH. $\overline{EA}$ is don't care after RESET. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**Table 1. 8XC524/8XC528 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes): | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | ES1 | ET2 | ES0 | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP*# | Interrupt priority | B8H | – | PS1 | PT2 | PS0 | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | SDA | SEL | – | – | – | – | T2EX | T2 | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | RD | WR | T1 | T0 | INT1 | INT0 | TxD | RxD | FFH |
| PCON | Power control | 87H | SMOD | – | – | – | GF1 | GF0 | PD | IDL | 0xxx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | F1 | P | 00H |
| RCAP2H# | Capture high | CBH | | | | | | | | | 00H |
| RCAP2L# | Capture low | CAH | | | | | | | | | 00H |
| SBUF | Serial data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial controller | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| S1BIT# | Serial I²C data | D9H/RD | SDI | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x0000000B |
| | | WR | SD0 | X | X | X | X | X | X | X | 0xxxxxxxB |
| S1INT# | Serial I²C interrupt | DAH | INT | X | X | X | X | X | X | X | 0xxxxxxxB |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| S1SCS*# | Serial I²C control | D8H/RD | SDI | SCI | CLH | BB | RBF | WBF | STR | ENS | xxxx0000B |
| | | WR | SD0 | SC0 | CLH | X | X | X | STR | ENS | 00xxxx00B |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| T2CON*# | Timer 2 control | C8H | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/T2 | CP/RL2 | 00H |
| TH0 | Timer high 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer high 1 | 8DH | | | | | | | | | 00H |
| TH2# | Timer high 2 | CDH | | | | | | | | | 00H |
| TL0 | Timer low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer low 1 | 8BH | | | | | | | | | 00H |
| TL2# | Timer low 2 | CCH | | | | | | | | | 00H |
| T3# | Watchdog timer | FFH | | | | | | | | | 00H |
| TMOD | Timer mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |
| WDCON# | Watchdog control | A5H | | | | | | | | | A5H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

**Table 2. Internal and External Program Memory Access with Security Bit Set**

| INSTRUCTION | ACCESS TO INTERNAL PROGRAM MEMORY | ACCESS TO EXTERNAL PROGRAM MEMORY |
|---|---|---|
| MOVC in internal program memory | YES | YES |
| MOVC in external program memory | NO | YES |

## INTERNAL DATA MEMORY

The internal data memory is divided into three physically separated segments: 256 bytes of RAM, 256 bytes of AUX-RAM, and a 128 bytes special function area. These can be addressed each in a different way.

- RAM 0 to 127 can be addressed directly and indirectly as in the 80C51. Address pointers are R0 and R1 of the selected register bank.
- RAM 128 to 255 can only be addressed indirectly as in the 80C51. Address pointers are R0 and R1 of the selected register bank.
- AUX-RAM 0 to 255 is indirectly addressed in the same way as external data memory with the MOVX instructions. Address pointers are R0, R1 of the selected register bank and DPTR. An access to AUX-RAM 0 to 255 will not affect ports P0, P2, P3.6 and P3.7.

An access to external data memory locations higher than 255 will be performed with the MOVX DPTR instructions in the same way as in the 8051 structure, so with P0 and P2 as data/address bus and P3.6 and P3.7 as write and read timing signals. Note that these external data memory cannot be accessed with R0 and R1 as address pointer.

## TIMER 2

Timer 2 is functionally equal to the Timer 2 of the 8052AH. Timer 2 is a 16-bit timer/counter. These 16 bits are formed by two special function registers TL2 and TH2. Another pair of special function register RCAP2L and RCAP2H form a 16-bit capture register or a 16-bit reload register. Like Timer 0 and 1, it can operate either as a timer or as an event counter. This is selected by bit C/T2N in the special function register T2CON. It has three operating modes: capture, autoload, and baud rate generator mode which are selected by bits in T2CON.

## WATCHDOG TIMER T3

The watchdog timer consists of an 11-bit prescaler and an 8-bit timer formed by special function register T3. The prescaler is incremented by an on-chip oscillator with a fixed frequency of 1MHz. The maximum tolerance on this frequency is –50% and +100%. The 8-bit timer increments every 2048 cycles of the on-chip oscillator. When a timer overflow occurs, the microcontroller is reset and a reset output pulse of 16 × 2048 cycles of the on-chip oscillator is generated at pin RST. The internal RESET signal is not inhibited when the external RST pin is kept low by, for example, an external reset circuit. The RESET signal drives port 1, 2, 3 into the high state and port 0 into the high impedance state.

The watchdog timer is controlled by one special function register WDCON with the direct address location A5H. WDCON can be read and written by software. A value of A5H in WDCON halts the on-chip oscillator and clears both the prescaler and timer T3. After the RESET signal, WDCON contains A5H. Every value other than A5H in WDCON enables the watchdog timer. When the watchdog timer is enabled, it runs independently of the XTAL-clock.

Timer T3 can be read on the fly. Timer T3 can only be written if WDCON contains the value 5AH. A successful write operation to T3 will clear the prescaler and WDCON, leaving the watchdog enabled and preventing inadvertent changes of T3. To prevent an overflow of the watchdog timer, the user program has to reload the watchdog timer within periods that are shorter than the programmed watchdog timer internal. This time interval is determined by an 8-bit value that has to be loaded in register T3 while at the same time the prescaler is cleared by hardware.

$$\text{Watchdog timer interval} = \frac{[256 - (T3)] \times 2048}{\text{on - chip oscillator frequency}}$$

## BIT-LEVEL I²C INTERFACE

This bit-level serial I/O interface supports the I²C-bus. P1.6/SCL and P1.7/SDA are the serial I/O pins. These two pins meet the I²C specification concerning the input levels and output drive capability. Consequently, these pins have an open drain output configuration. All the four modes of the I²C-bus are supported:

- master transmitter
- master receiver
- slave transmitter
- slave receiver

The advantages of the bit-level I²C hardware compared with a full software I²C implementation are:

- the hardware can generate the SCL pulse
- Testing a single bit (RBF respectively, WBF) is sufficient as a check for error free transmission.

The bit-level I²C hardware operates on serial bit level and performs the following functions:

- filtering the incoming serial data and clock signals
- recognizing the START condition
- generating a serial interrupt request SI after reception of a START condition and the first falling edge of the serial clock
- recognizing the STOP condition
- recognizing a serial clock pulse on the SCL line
- latching a serial bit on the SDA line (SDI)
- stretching the SCL LOW period of the serial clock to suspend the transfer of the next serial data bit
- setting Read Bit Finished (RBF) when the SCL clock pulse has finished and Write Bit Finished (WBF) if there is no arbitration loss detected (i.e., SDA = 0 while SDO = 1)
- setting a serial clock Low-to-High detected (CLH) flag
- setting a Bus Busy (BB) flag on a START condition and clearing this flag on a STOP condition
- releasing the SCL line and clearing the CLH, RBF and WBF flags to resume transfer of the next serial data bit
- generating an automatic clock if the single bit data register S1BIT is used in master mode.

The following functions must be done in software:

- handling the I²C START interrupts
- converting serial to parallel data when receiving
- converting parallel to serial data when transmitting
- comparing the received slave address with its own
- interpreting the acknowledge information
- guarding the I²C status if RBF or WBF = 0.

Additionally, if acting as master:
- generating START and STOP conditions
- handling bus arbitration
- generating serial clock pulses if S1BIT is not used.

Three SFRs control the bit-level I²C interface: S1INT, S1BIT and S1SCS.

## INTERRUPT SYSTEM

The interrupt structure of the 8XC528 is the same as that used in the 80C51, but includes two additional interrupt sources: one for the third timer/counter, T2, and one for the I²C interface. The interrupt enable and interrupt priority registers are IE and IP.

### IE: Interrupt Enable Register

This register is located at address A8H. Refer to Table 3.

**IE SFR (A8H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| EA | ES1 | ET2 | ES | ET1 | EX1 | ET0 | EX0 |

### IP: Interrupt Priority Register

This register is located at address B8H. Refer to Table 4.

**IP SFR (B8H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | PS1 | PT2 | PS | PT1 | PX1 | PT0 | PX0 |

**Table 3. Description of IE Bits**

| MNEMONIC | BIT | FUNCTION |
|---|---|---|
| EA | IE.7 | General enable/disable control:<br>0 = **NO** interrupt is enabled.<br>1 = **ANY** individually enabled interrupt will be accepted. |
| ES1 | IE.6 | Enable **bit-level $I_2C$ I/O** interrupt |
| ET2 | IE.5 | Enable **Timer 2** interrupt |
| ES | IE.4 | Enable **Serial Port** interrupt |
| ET1 | IE.3 | Enable **Timer 1** interrupt |
| EX1 | IE.2 | Enable **External** interrupt 1 |
| ET0 | IE.1 | Enable **Timer 0** interrupt |
| EX0 | IE.0 | Enable **External** interrupt 0 |

**Table 4. Description of IP Bits**

| MNEMONIC | BIT | FUNCTION |
|---|---|---|
| – | IP.7 | **Reserved.** |
| PS1 | IP.6 | **Bit-level I²C** interrupt priority level |
| PT2 | IP.5 | **Timer 2** interrupt priority level |
| PS | IP.4 | **Serial Port** interrupt priority level |
| PT1 | IP.3 | **Timer 1** interrupt priority level |
| PX1 | IP.2 | **External Interrupt 1** priority level |
| PT0 | IP.1 | **Timer 0** interrupt priority level |
| PX0 | IP.0 | **External Interrupt 0** priority level |

The interrupt vector locations and the interrupt priorities are:

| **Source** | | **Priority within Level** |
|---|---|---|
| Vector | Address | |
| 0003H | IE0 | Highest |
| 002BH | TF2+EXF2 | |
| 0053H | SI (I²C) | |
| 000BH | TF0 | |
| 0013H | IE1 | |
| 001BH | TF1 | |
| 0023H | R1+T1 | Lowest |

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the Logic Symbol.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-up reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-up, the voltage on $V_{DD}$ and RST must come up at the same time for a proper start-up.

## IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. The power-down mode can be terminated by a RESET in the same way as in the 80C51 or in addition by one of two external interrupts, INT0 or INT1. A termination with an external interrupt does not affect the internal data memory and does not affect the special function registers. This makes it possible to exit power-down without changing the port output levels. To terminate the power-down mode with an external interrupt $\overline{INT0}$ or $\overline{INT1}$ must be switched to level-sensitive and must be enabled. The external interrupt input signal $\overline{INT0}$ and $\overline{INT1}$ must be kept low until the oscillator has restarted and stabilized. An instruction following the instruction that puts the device in the power-down mode will be executed. A reset generated by the watchdog timer terminates the power-down mode in the same way as an external RESET, and only the contents of the on-chip RAM are preserved. The control bits for the reduced power modes are in the special function register PCON.

## DESIGN CONSIDERATIONS

At power-on, the voltage on $V_{DD}$ and RST must come up at the same time for a proper start-up.

When the idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when idle is terminated by reset, the instruction following the one that invokes idle should not be one that writes to a port pin or to external memory.

Table 5 shows the state of I/O ports during low current operating modes.

**Table 5. External Pin Status During Idle and Power-Down Modes**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | 0 to +70, or<br>–40 to +85, or<br>–40 to +125 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on any other pin to $V_{SS}$ | –0.5 to $V_{DD}$ +0.5 | V |
| Input, output current on any two pins | ±10 | mA |
| Power dissipation<br>(based on package heat transfer limitations, not device power consumption) | 1.0 | W |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C ($V_{DD}$ = 5V ±10%), –40°C to +85°C ($V_{DD}$ = 5V ±10%), or –40°C to +125°C ($V_{DD}$ = 5V ±10%), $V_{SS}$=0V

| SYMBOL | PARAMETER | PART TYPE | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input low voltage,<br>except $\overline{EA}$, P1.6/SCL, P1.7/SDA | 0°C to 70°C<br>–40°C to +85°C | | –0.5<br>–0.5 | $0.2V_{CC}$–0.1<br>$0.2V_{CC}$–0.15 | V<br>V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | 0°C to 70°C<br>–40°C to +85°C | | 0<br>0 | $0.2V_{CC}$–0.3<br>$0.2V_{CC}$–0.35 | V<br>V |
| $V_{IL2}$ | Input low voltage to P1.6/SCL, P1.7/SDA[5] | | | –0.5 | 0.3V | V |
| $V_{IH}$ | Input high voltage,<br>except XTAL1, RST, P1.6/SCL, P1.7/SDA | 0°C to 70°C<br>–40°C to +85°C | | $0.2V_{CC}$+0.9<br>$0.2V_{CC}$+1.0 | $V_{CC}$+0.5<br>$V_{CC}$+0.5 | V<br>V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | 0°C to 70°C<br>–40°C to +85°C | | $0.7V_{CC}$<br>$0.7V_{CC}$+0.1 | $V_{CC}$+0.5<br>$V_{CC}$+0.5 | V<br>V |
| $V_{IH2}$ | Input high voltage, P1.6/SCL, P1.7/SDA[5] | | | 3.0 | 6.0 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3, except P1.6/SCL, P1.7/SDA[1] | | $I_{OL}$ = 1.6mA[1] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$[1] | | $I_{OL}$ = 3.2mA[1] | | 0.45 | V |
| $V_{OL2}$ | Output low voltage, P1.6/SCL, P1.7/SDA | | $I_{OL}$ = 3.0mA[1] | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3 | | $I_{OH}$ = –60µA<br>$I_{OH}$ = –25µA | 2.4<br>$0.75V_{CC}$ | | V<br>V |
| $V_{OH1}$ | Output high voltage, Port 0 in external bus mode, ALE, $\overline{PSEN}$, RST | | $I_{OH}$ = –800µA<br>$I_{OH}$ = –300µA | 2.4<br>$0.75V_{CC}$ | | V<br>V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3,<br>except P1.6/SCL, P1.7/SDA | 0°C to 70°C<br>–40°C to +85°C | $V_{IN}$ = 0.45V | | –50<br>–75 | µA<br>µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3,<br>except P1.6/SCL, P1.7/SDA | 0°C to 70°C<br>–40°C to +85°C | See note 3 | | –650<br>–750 | µA<br>µA |
| $I_{IL1}$ | Input leakage current, port 0 | | $V_{IN}$ = $V_{IL}$ or $V_{IH}$ | | ±10 | µA |
| $I_{IL2}$ | Input leakage current, P1.6/SCL, P1.7/SDA | | 0V<Vi<6.0V<br>0V<$V_{CC}$<6.0V | | ±10 | µA<br>µA |
| $I_{CC}$ | Power supply current:<br><br>Active mode @ 16MHz<br><br>Idle mode @ 16MHz<br><br>Power down mode | <br><br>0°C to 70°C<br>–40°C to +85°C<br>0°C to 70°C<br>–40°C to +85°C | See note 4 | | <br><br>25<br>35<br>5<br>6<br>50 | <br><br>mA<br><br>mA<br><br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | | 50 | 300 | kΩ |
| $C_{IO}$ | Pin Capacitance | | | | 10 | pF |

**NOTES:**

1. Capacitive loading on Port 0 and Port 2 may cause spurious noise pulses to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port and port 2 pins when these pins make 1-to-0 transactions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows: 10mA per port pin, port 0 total (all bits) 26mA, ports 1, 2, and total each (all bits) 15mA.
2. Capacitive loading on Ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
3. Pins of ports 1, 2, and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
4. See Figures 10 through 13 for $I_{CC}$ test conditions.
5. The input threshold voltage of P1.6 and P1.7 (SIO1) meets the $I^2C$ specification, so an input voltage below 1.5V will be recognized as a logic 0 while an input voltage above 3.0V will be recognized as a logic 1.

## AC ELECTRICAL CHARACTERISTICS[1, 2]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 1 | Oscillator frequency: **Speed Versions**<br>87C528 P878C528EXX<br>P878C528GXX | | | 3.5<br>3.5 | 16<br>20 | MHz<br>MHz |
| $t_{LHLL}$ | 1 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 1 | Address valid to ALE low | 8 | | $t_{CLCL}-55$ | | ns |
| $t_{LLAX}$ | 1 | Address hold after ALE low | 28 | | $t_{CLCL}-35$ | | ns |
| $t_{LLIV}$ | 1 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 1 | ALE low to $\overline{PSEN}$ low | 23 | | $t_{CLCL}-40$ | | ns |
| $t_{PLPH}$ | 1 | $\overline{PSEN}$ pulse width | 143 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 1 | $\overline{PSEN}$ low to valid instruction in | | 83 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 1 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 1 | Input instruction float after $\overline{PSEN}$ | | 38 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 1 | Address to valid instruction in | | 208 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 1 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 2, 3 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 2, 3 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 2, 3 | $\overline{RD}$ low to valid data in | | 148 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 2, 3 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 2, 3 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-70$ | ns |
| $t_{LLDZ}$ | 2, 3 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 2, 3 | Address to valid data in | | 398 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 2, 3 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 138 | 238 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 2, 3 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 120 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 2, 3 | Data valid to $\overline{WR}$ transition | 3 | | $t_{CLCL}-60$ | | ns |
| $t_{WHQX}$ | 2, 3 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{RLAZ}$ | 2, 3 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 2, 3 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 6 | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | 6 | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | 6 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 6 | Fall time | | 20 | | 20 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 4 | Serial port clock cycle time | 750 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 4 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 4 | Output data hold after clock rising edge | 8 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 4 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 4 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}-133$ | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.

## AC ELECTRICAL CHARACTERISTICS – I²C INTERFACE

| SYMBOL | PARAMETER | INPUT | OUTPUT | I²C SPECIFICATION |
|---|---|---|---|---|
| **SCL TIMING CHARACTERISTICS** | | | | |
| $t_{HD;STA}$ | START condition hold time | ≥ 14 $t_{CLCL}$[1] | Note 2 | ≥ 4.0µs |
| $t_{LOW}$ | SCL LOW time | ≥ 16 $t_{CLCL}$ | Note 2 | ≥ 4.7µs |
| $t_{HIGH}$ | SCL HIGH time | ≥ 14 $t_{CLCL}$[1] | ≥ 80 $t_{CLCL}$[3] | ≥ 4.0µs |
| $t_{RC}$ | SCL rise time | ≤ 1µs[1] | Note 5 | ≤ 1.0µs |
| $t_{FC}$ | SCL fall time | ≤ 0.3µs[1] | ≤ 0.3µs[6] | ≤ 0.3µs |
| **SDA TIMING CHARACTERISTICS** | | | | |
| $t_{SU;DAT1}$ | Data set-up time | ≥ 250ns | Note 2 | ≥ 250ns |
| $t_{HD;DAT}$ | Data hold time | ≥ 0ns | Note 2 | ≥ 0ns |
| $t_{SU;STA}$ | Repeated START set-up time | ≥ 14 $t_{CLCL}$[1] | Note 2 | ≥ 4.7µs |
| $t_{SU;STO}$ | STOP condition set-up time | ≥ 14 $t_{CLCL}$[1] | Note 2 | ≥ 4.0µs |
| $t_{BUF}$ | Bus free time | ≥ 14 $t_{CLCL}$[1] | Note 2 | ≥ 4.7µs |
| $t_{RD}$ | SDA rise time | ≤ 1µs[4] | Note 5 | ≤ 1.0µs |
| $t_{FD}$ | SDA fall time | ≤ 0.3µs[4] | ≤ 0.3µs[6] | ≤ 0.3µs |

**NOTES:**
1. At $f_{CLK}$ = 3.5MHz, this evaluates to 14 × 286ns = 4µs, i.e., the bit-level I²C interface can respond to the I²C protocol for $f_{CLK}$ ≥ 3.5MHz.
2. This parameter is determined by the user software, it has to comply with the I²C.
3. This value gives the autoclock pulse length which meets the I²C specification for the specified XTAL clock frequency range. Alternatively, the SCL pulse may be timed by software.
4. Spikes on SDA and SCL lines with a duration of less than 4 × $f_{CLK}$ will be filtered out.
5. The rise time is determined by the external bus line capacitance and pull-up resistor, it must be ≤ 1µs.
6. The maximum capacitance on bus lines SDA and SCL is 400pF.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{\text{PSEN}}$
Q – Output data
R – $\overline{\text{RD}}$ signal
t – Time
V – Valid
W – $\overline{\text{WR}}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{\text{PSEN}}$ low.

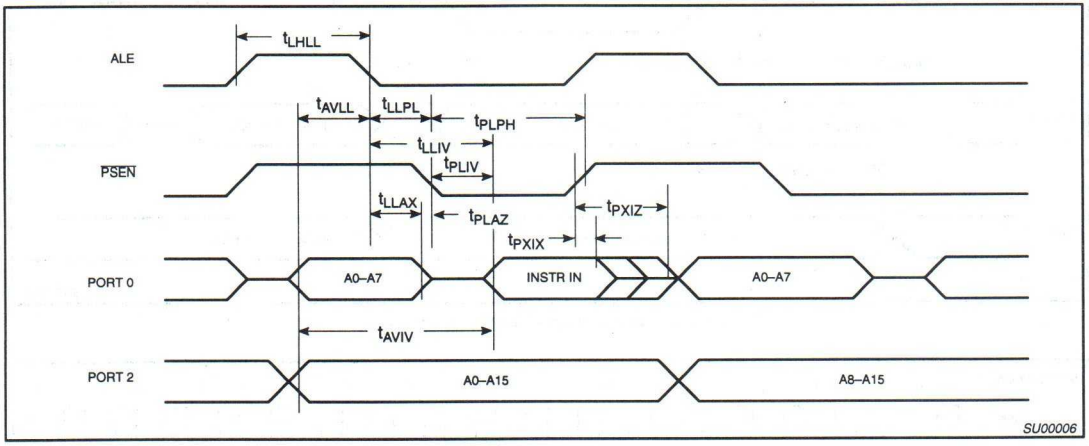



**Figure 1. External Program Memory Read Cycle**

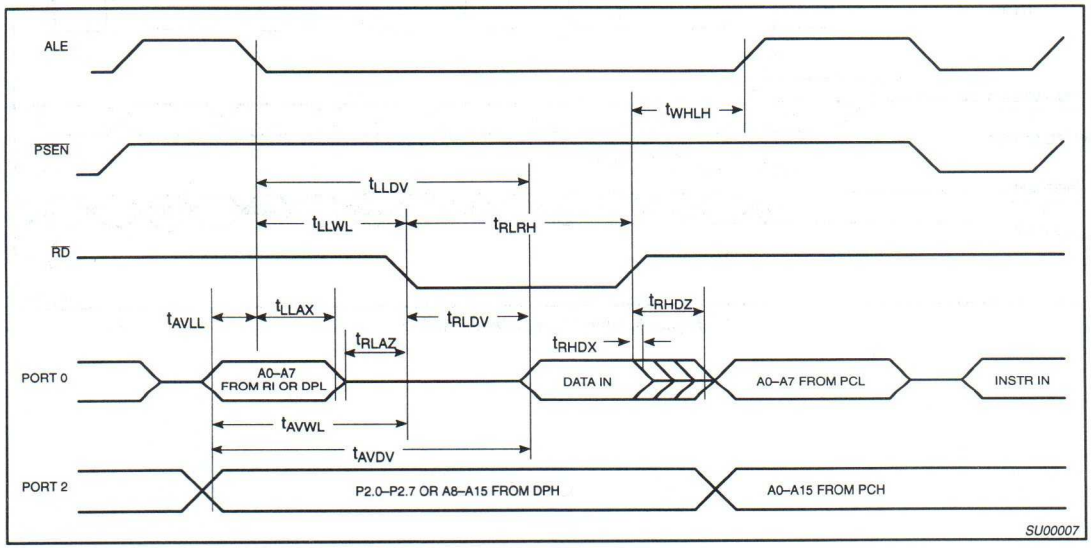



**Figure 2. External Data Memory Read Cycle**

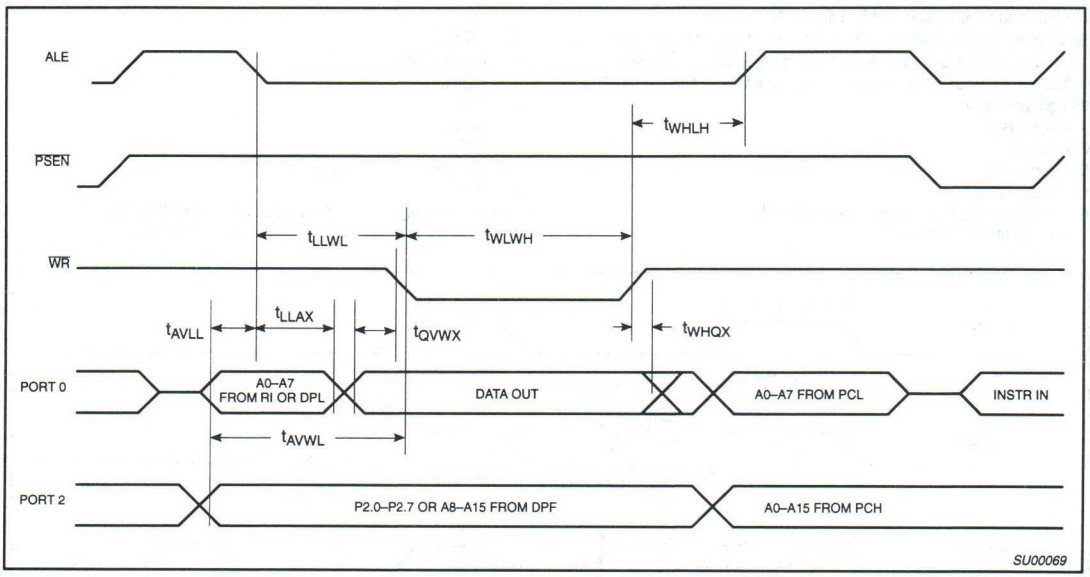



**Figure 3. External Data Memory Write Cycle**

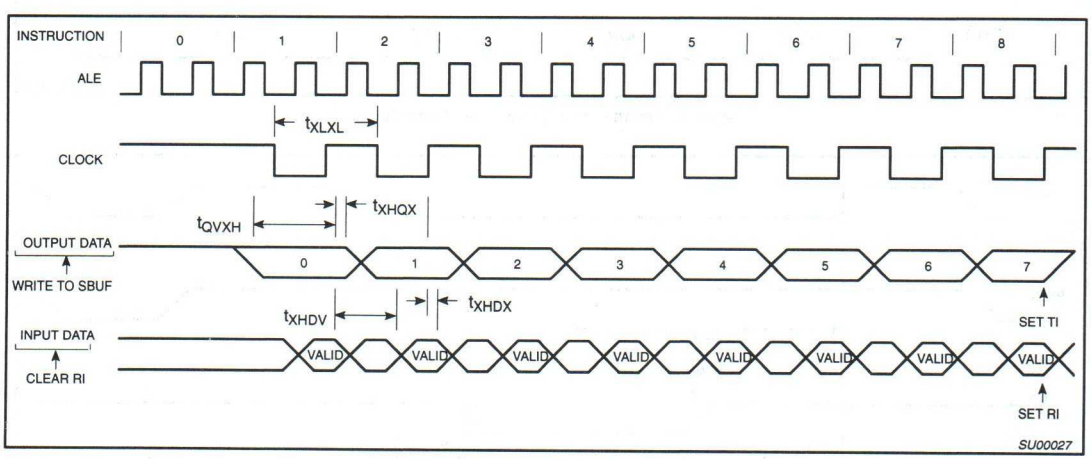



**Figure 4. Shift Register Mode Timing**

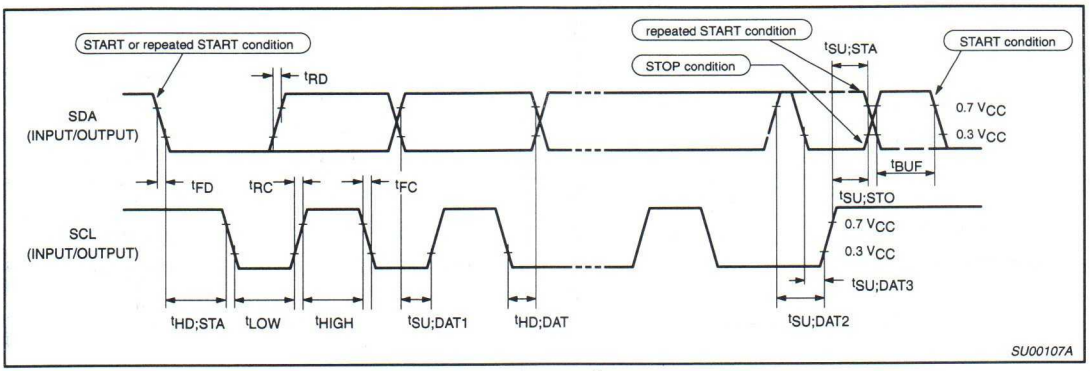



**Figure 5. Timing SIO1 ($I^2C$) Interface**

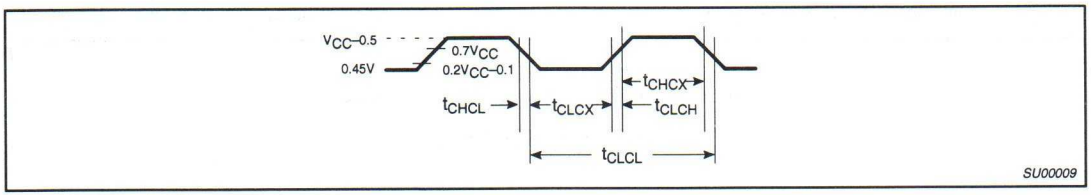



**Figure 6. External Clock Drive**

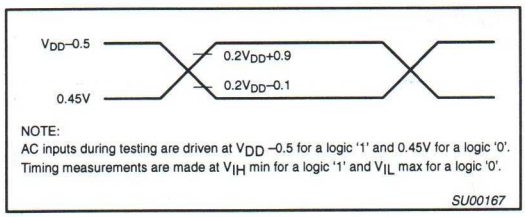



**Figure 7. AC Testing Input/Output**

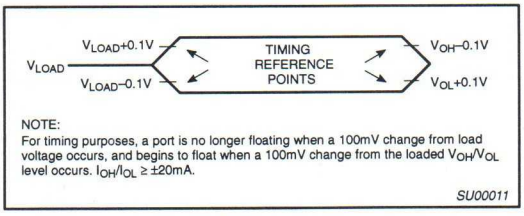



**Figure 8. Float Waveform**

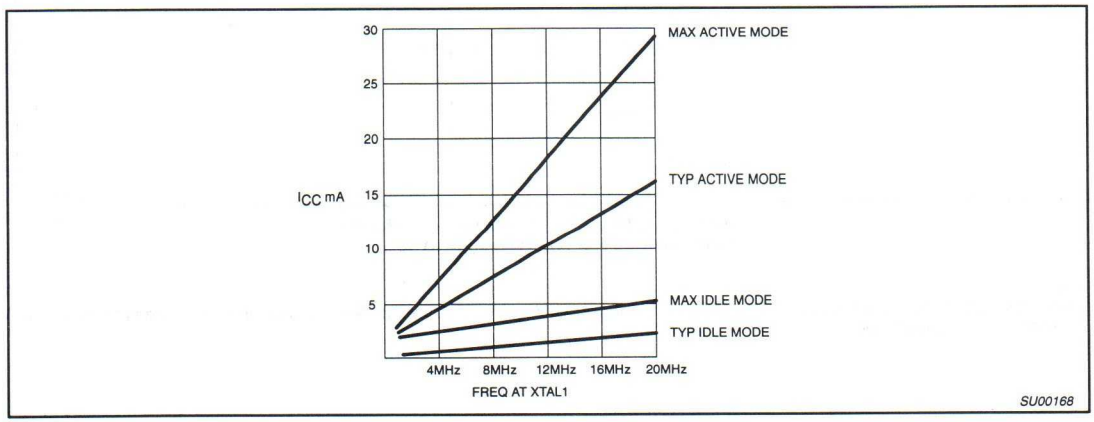



**Figure 9. $I_{CC}$ vs. FREQ.**
**Valid only within frequency specifications of the device under test**

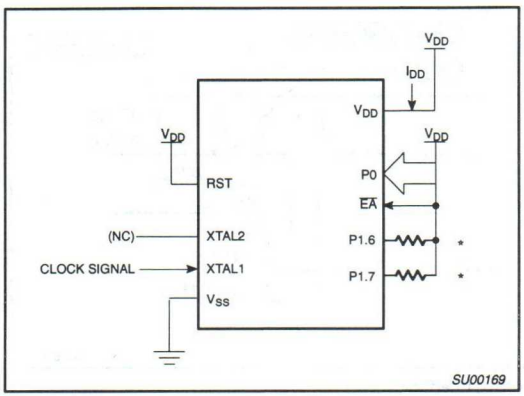



**Figure 10. $I_{DD}$ Test Condition, Active Mode**
**All other pins are disconnected**

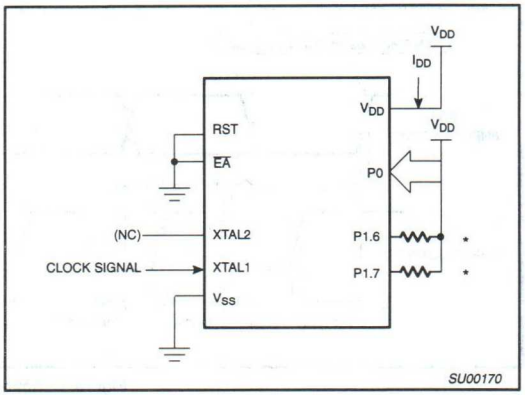



**Figure 11. $I_{DD}$ Test Condition, Idle Mode**
**All other pins are disconnected**

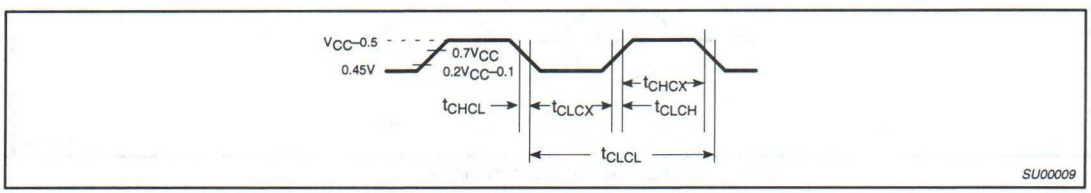



**Figure 12. Clock Signal Waveform for $I_{DD}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

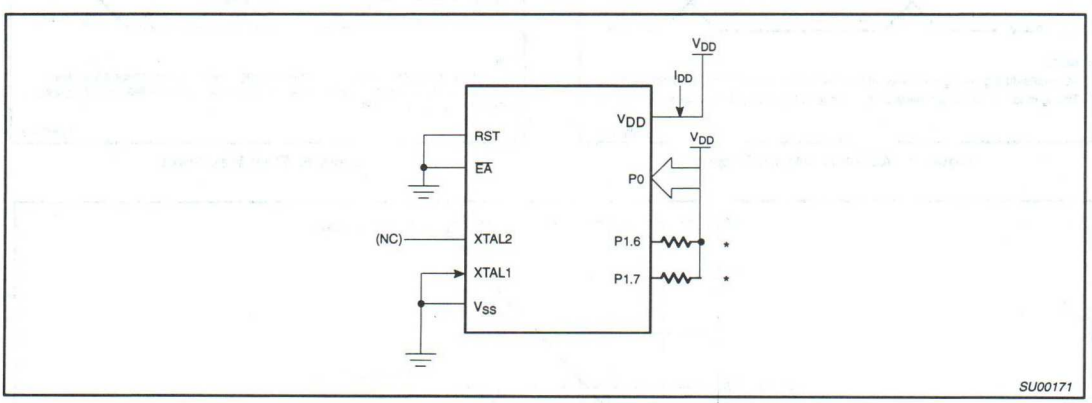



**Figure 13. $I_{DD}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{DD}$ = 2V to 5.5V**

**NOTE:**
* Ports 1.6 and 1.7 should be connected to $V_{DD}$ through resistors of sufficiently high value such that the sink current into these pins does not exceed the $I_{OL1}$ specifications.

## EPROM CHARACTERISTICS

The 87C528 is programmed by using a modified Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C528 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C528 manufactured by Philips.

Table 6 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the lock bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 14 and 15. Figure 16 shows the circuit configuration for normal program memory verification.

### Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 14. Note that the 87C528 is running with a 4 to 6MHz oscillator The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1, 2 and 3, as shown in Figure 14. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 3 specified in Table 6 are held at the 'Program Code Data' levels indicated in Table 6. The ALE/$\overline{PROG}$ is pulsed low 25 times as shown in Figure 15.

To program the encryption table, repeat the 25 pulse programming sequence for addresses 0 through 3FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the lock bits, repeat the 25 pulse programming sequence using the 'Pgm Lock Bit' levels. After one lock bit is programmed, further programming of the code memory and encryption table is disabled. However, the other lock bit can still be programmed.

Note that the $\overline{EA}$/$V_{PP}$ pin must not be allowed to 90 above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

#### Program Verification

If lock bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 1, 2 and 3 as shown in Figure 16. The other pins are held at the 'Verify Code Data' levels indicated in Table 6. The contents of the address location will be emitted on port 0. External pull ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

#### Reading the Signature Bytes

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031 H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:

(030H) = 15H indicates manufactured by Philips
(031H) = 9BH indicates 87C528

### Program Lock Bits

The 87C528 has 3 programmable lock bits that will provide different levels of protection for the on-chip code and data (see Table 7).

Erasing the EPROM also erases the encryption array and the program lock bits, returning the part to full functionality.

### Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 6, and which satisfies the timing specifications, is suitable.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345-5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-sec/cm2. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm2 rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient. Erasure leaves the array in an all 1 s state.

### Table 6. EPROM Programming Modes

| MODE | RST | PSEN | ALE/$\overline{PROG}$ | $\overline{EA}$/$V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm lock bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm lock bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |
| Pgm lock bit 3 | 1 | 0 | 0* | $V_{PP}$ | 0 | 1 | 0 | 1 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. Vcc = 5V ±10% during programming and verification.
* ALE/$\overline{PROG}$ receives 25 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

**Table 7. Program Lock Bits**

| PROGRAM LOCK BITS[1,2] | | | | |
|---|---|---|---|---|
| | LB1 | LB2 | LB3 | PROTECTION DESCRIPTION |
| 1 | U | U | U | No Program Lock features enabled. (Code verify will still be encrypted by the Encryption Array if programmed.) |
| 2 | P | U | U | MOVC instructions executed from external program memory are disabled from fetching code bytes from Internal memory, EA is jumped and latched on Reset, and further programming of the EPROM is disabled. |
| 3 | P | P | U | Same as 2, also verify is disabled. |
| 4 | P | P | P | Same as 3, external execution is disabled. Internal data RAM is not accessible. |

**NOTES:**
1. P - programmed. U - unprogrammed.
2. Any other combination of the lock bits is not defined.

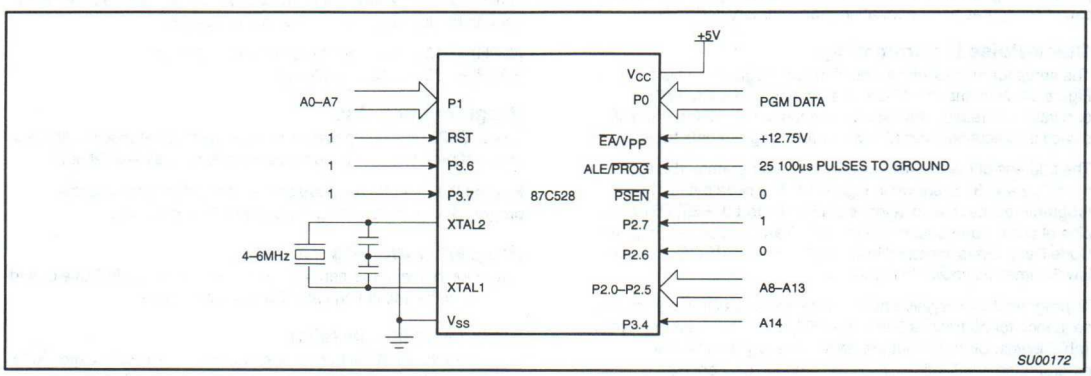



**Figure 14. Programming Configuration**

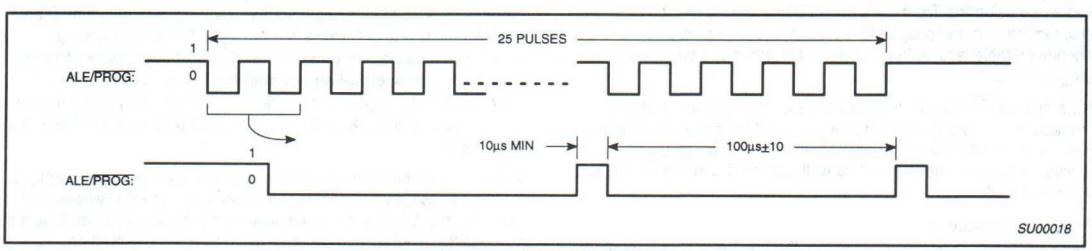



**Figure 15. PROG Waveform**

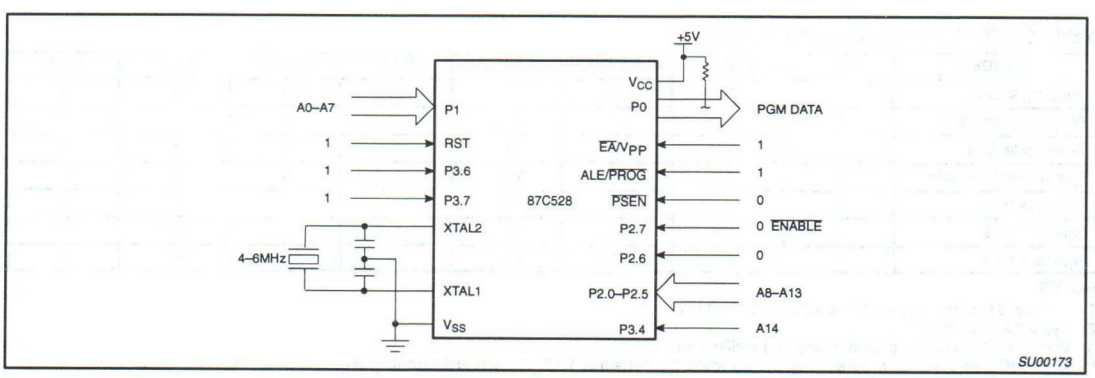
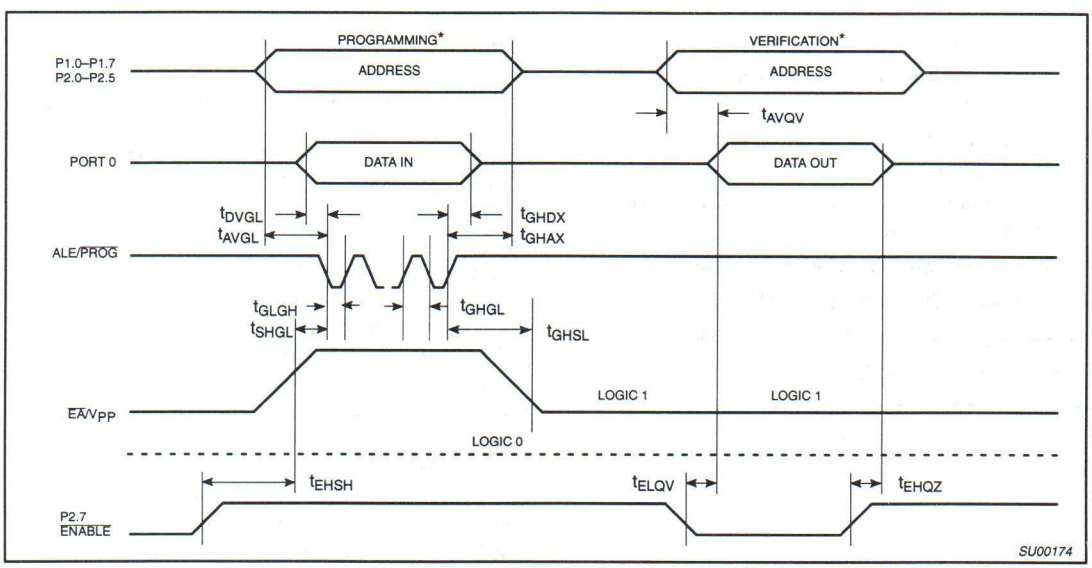



**Figure 16. Program Verfication**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, Vcc = 5V±10%, $V_{SS}$ = 0V (See Figure 17)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




**NOTE:**

* FOR PROGRAMMING VERIFICATION SEE FIGURE 14.
FOR VERIFICATION CONDITIONS SEE FIGURE 16.

**Figure 17. EPROM Programing and Verification**

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# 8-bit microcontroller with EMC P8xCE528

## 1 FEATURES

- 80C51 central processing unit
- 32K x 8 ROM, expandable externally to 64 kbytes
- ROM code protection
- 512 x 8 RAM, expandable externally to 63 kbytes
- Four 8-bit I/O ports
- Full-duplex UART compatible with the standard 80C51 and the 8052
- Two standard 16-bit timer/counters
- An additional 16-bit timer (functionally equivalent to the timer 2 of the 8052)
- On-chip Watchdog Timer (WDT) with an on-chip oscillator
- Bit-level $I^2C$-bus hardware serial I/O Port
- 7-source and 7-vector interrupt structure with 2 priority levels
- Up to 3 external interrupt request inputs
- Two programmable power reduction modes (Idle and Power-down)
- Termination of Idle mode by any interrupt, external or WDT (watchdog) reset
- Wake-up from Power-down by external interrupt, external or WDT reset
- Software enable/disable of ALE output pulse
- Electro-Magnetic Compatibility (EMC) improvements
- XTAL frequency range: 3.5 MHz to 16 MHz
- 4.5 to 5.5 V supply voltage range
- Extended Temperature range (–40 to +85°C)

## 2 GENERAL DESCRIPTION

The P83CE528, P80CE528 (hereafter generically referred to as P8xCE528) single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the PCB80C51 microcontroller family.

The P8xCE528 has the same instruction set as the 80C51. Two versions of the derivative exist:

- P83CE528: 32kbytes mask programmable ROM
- P80CE528: ROMless version of the P83CE528

This device provides architectural enhancements that make it applicable in a variety of applications in general control systems, especially in those systems which need a large ROM and RAM capacity on chip.

The P8xCE528 contains a non-volatile 32K x 8 read-only program memory (P83CE528), a volatile 512 x 8 read/write data memory, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a 16-bit timer (identical to the timer 2 of the 8052), a multi-source, two-priority-level, nested interrupt structure, two serial interfaces (UART and bit-level $I^2C$-bus), an on-chip oscillator and timing circuits, a watchdog timer (WDT) with a separate on-chip oscillator. For systems that require extra capability, the P8xCE528 can be expanded using standard TTL compatible memories and logic.

In addition, the P8xCE528 has two software selectable modes of power reduction – Idle mode and Power-down mode. The Idle mode freezes the CPU while allowing the RAM, timers, serial ports and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte, and 17 three-byte. With a 16 MHz crystal, 58% of the instructions are executed in 0.75µs and 40% in 1.5µs. Multiply and divide instruction require 3µs.

## 3 ELECTROMAGNETIC COMPATIBILITY (EMC)

Primary attention is paid to the reduction of electro-magnetic emission of the microcontroller P8xCE528. The following features reduce the electro-magnetic emission and additionally improve the electromagnetic susceptibility:

- Two supply voltage pins ($V_{DD1;2}$) and four ground pins ($V_{SS1;2;3;4}$) are provided on the package as follows:
  - one $V_{DD}$ and one $V_{SS}$ as a pair of pins placed mid-center on one side of the package
  - a second pair of $V_{DD}$ and $V_{SS}$ pins placed mid-center on the opposite side of the package
  - two more $V_{SS}$ pins, one placed on each of the other two sides of the package.
- Separated $V_{DD}$ pins for the internal logic and the port buffers.
- Internal decoupling capacitance improves the EMC radiation behavior and the EMC immunity.
- External capacitors are to be located as close as possible between pins $V_{DD1}$ and $V_{SS1}$ as well as $V_{DD2}$ and $V_{SS3}$; ceramic chip capacitors are recommended (100 nF).

### 3.1 Recommendation on ALE

For applications that require no external memory or temporarily no external memory: the ALE output signal (pulses at a frequency of $f_{OSC}/6$) can be disabled under software control (bit 5 in the PCON SFR: "RFI"); if disabled, no ALE pulse will occur. ALE pin will be pulled down internally, switching an external address latch to a quiet state. The MOVX instruction will still toggle ALE as a normal MOVX. ALE will retain its normal HIGH value during Idle mode and a LOW value during Power-down mode while in the "RFI" reduction mode.

Additionally during internal access ($\overline{EA}$ = 1) ALE will toggle normally when the address exceeds the internal program memory size. During external access ($\overline{EA}$ = 0) ALE will always toggle normally, whether the flag "RFI" is set or not.

## 4 ORDERING INFORMATION

| EXTENDED TYPE NUMBER | PACKAGE | | | | TEMPERATURE RANGE (°C) | FREQUENCY (MHz) |
|---|---|---|---|---|---|---|
| | PINS | PIN POSITION | MATERIAL | CODE | | |
| **ROMless** | | | | | | |
| P80CE528EBB | 44 | QFP | plastic | SOT307-2 | 0 to +70 | 3.5 to 16 |
| P80CE528EFB | 44 | QFP | plastic | SOT307-2 | –40 to +85 | 3.5 to 16 |
| P80CE528EBA | 44 | PLCC | plastic | SOT187-2 | 0 to +70 | 3.5 to 16 |
| P80CE528EFA | 44 | PLCC | plastic | SOT187-2 | –40 to +85 | 3.5 to 16 |
| **ROM** | | | | | | |
| P83CE528EBB | 44 | QFP | plastic | SOT307-2 | 0 to +70 | 3.5 to 16 |
| P83CE528EFB | 44 | QFP | plastic | SOT307-2 | –40 to +85 | 3.5 to 16 |
| P83CE528EBA | 44 | PLCC | plastic | SOT187-2 | 0 to +70 | 3.5 to 16 |
| P83CE528EFA | 44 | PLCC | plastic | SOT187-2 | –40 to +85 | 3.5 to 16 |

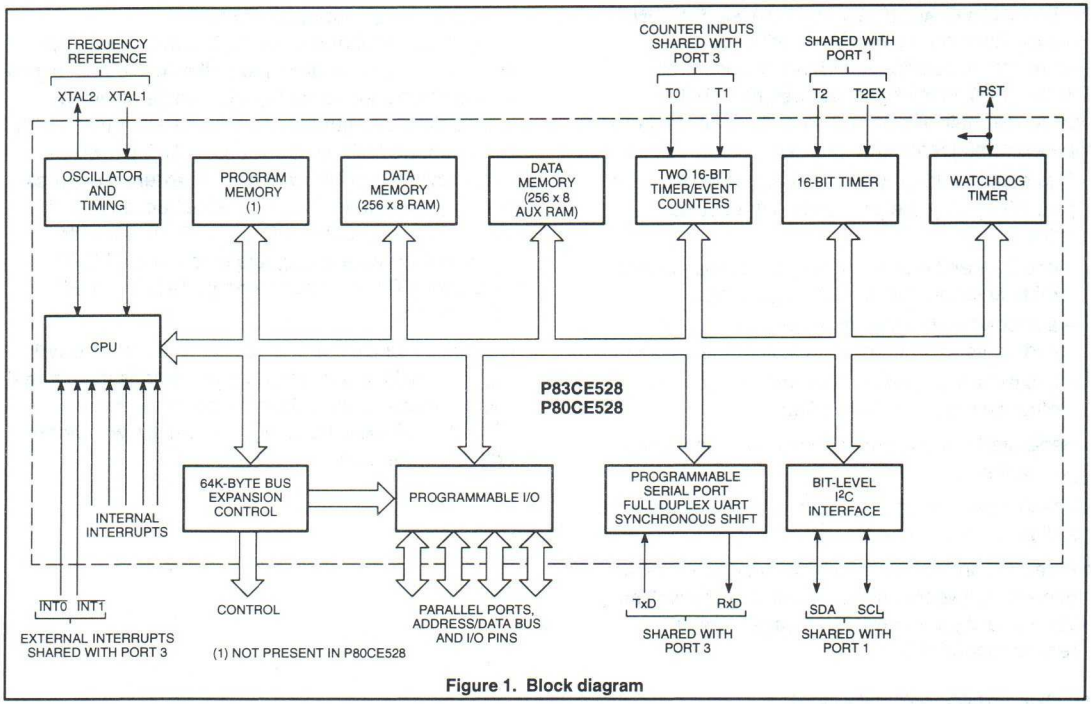



**Figure 1. Block diagram**

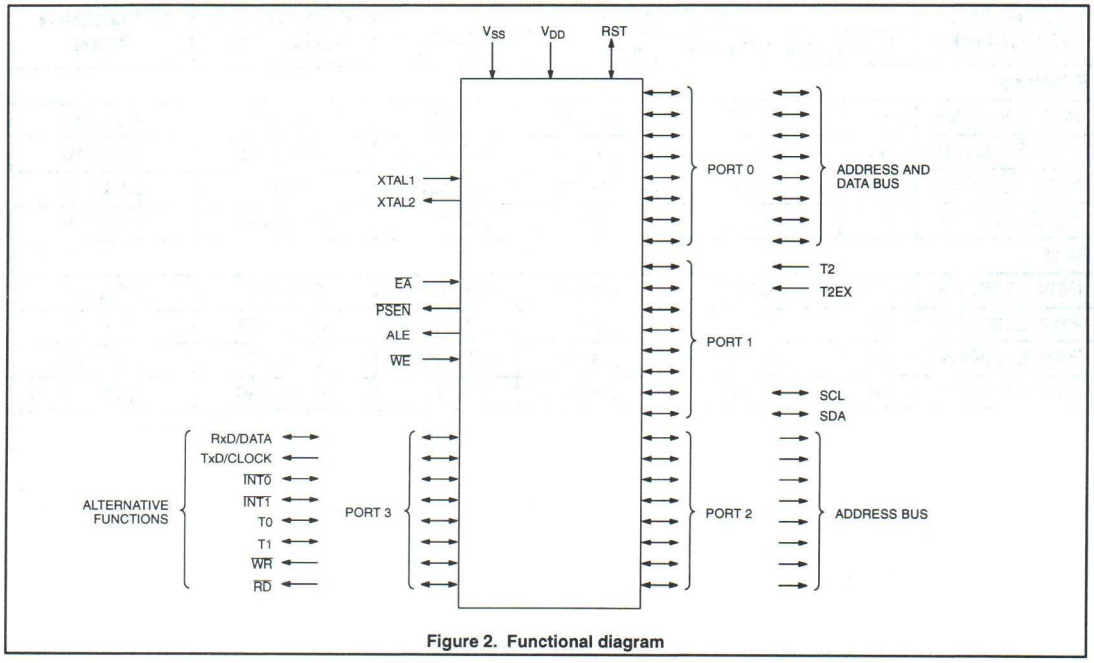



**Figure 2. Functional diagram**

## 5 PINNING INFORMATION

### 5.1 Pinning

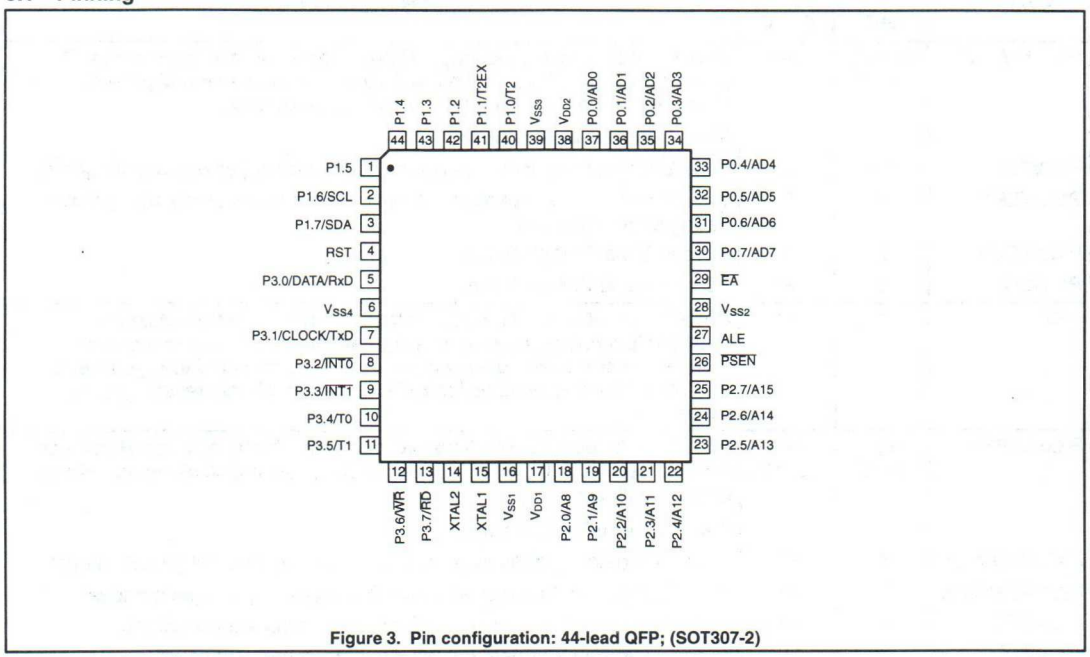



**Figure 3. Pin configuration: 44-lead QFP; (SOT307-2)**

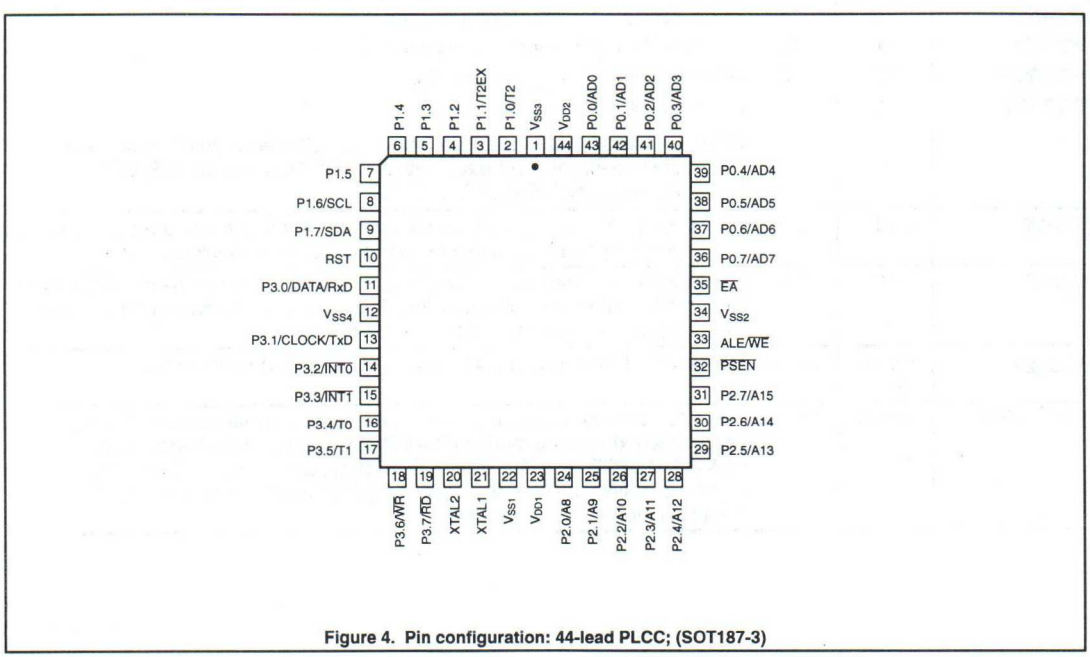



**Figure 4. Pin configuration: 44-lead PLCC; (SOT187-3)**

## PIN DESCRIPTION

| SYMBOL | PIN | | DESCRIPTION |
|---|---|---|---|
| | QFP | PLCC | |
| P1.0 to P1.7 | 40–44, 1–3 | 2–9 | **Port 1:** 8-it quasi-bidirectional I/O port. Port 1 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups, except P1.6 and P1.7, which have open drain outputs. |
| | | | **Port 1 alternative functions:** |
| P1.0/T2 | 40 | 2 | Timer/event counter 2 external event counter input (falling edge triggered). |
| P1.1/T2EX | 41 | 3 | Timer/event counter 2 capture/reload trigger or external interrupt 2 input (falling edge triggered). |
| P1.6/SCL | 2 | 8 | $I^2C$-bus Serial Port clock line. |
| P1.7/SDA | 3 | 9 | $I^2C$-bus Serial Port data line. |
| RST | 4 | 10 | **RESET:** A HIGH level on this pin for two machine cycles while the oscillator is running, resets the device. An internal pull-down resistor permits power-on reset using only a capacitor connected to $V_{DD}$. After a WDT overflow, this pin is pulled HIGH while the internal reset signal is active. |
| P3.0 to P3.7 | 5, 7–13 | 11, 13–19 | **Port 3:** 8-bit quasi-bidirectional I/O port with internal pull-ups. Port 3 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups. |
| | | | **Port 3 alternative functions:** |
| P3.0/RxD/data | 5 | 11 | Serial Port data input (asynchronous) or data input/output (synchronous). |
| P3.1/TxD/clock | 7 | 13 | Serial Port data output (asynchronous) or clock output (synchronous). |
| P3.2/$\overline{INT0}$ | 8 | 14 | External interrupt 0 or gate control input for timer/event counter 0. |
| P3.3/$\overline{INT1}$ | 9 | 15 | External interrupt 1 or gate control input for timer/event counter 1. |
| P3.4/T0 | 10 | 16 | External input for timer/event counter 0. |
| P3.5/T1 | 11 | 17 | External input for timer/event counter 1. |
| P3.6/$\overline{WR}$ | 12 | 18 | External data memory write strobe. |
| P3.7/$\overline{RD}$ | 13 | 19 | External data memory read strobe. |
| | | | The generation or use of a Port 3 pin as an alternative function is carried out automatically by the P8xCE528 provided the associated Special Function Register (SFR) bit is set HIGH. |
| XTAL2 | 14 | 20 | **Crystal pin 2:** Output of the inverting amplifier that forms the oscillator. This pin left open-circuit when an external oscillator clock is used. |
| XTAL1 | 15 | 21 | **Crystal pin 1:** Input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external oscillator clock signal when an external oscillator is used. |
| $V_{SS1;2;3;4}$ | 16, 28, 39, 6 | 22, 34, 1, 12 | **Ground:** Circuit ground potential. All pins must be connected. |
| P2.0 to P2.7 | 18–25 | 24–31 | **Port 2:** 8-bit quasi-bidirectional I/O Port with internal pull-ups. During access to external memories (RAM/ROM) that use 16-bit addresses (MOVX@DPTR) Port 2 emits the high-order address byte (A8 to A15). Port 2 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups. |

| SYMBOL | PIN | | DESCRIPTION |
|---|---|---|---|
| | QFP | PLCC | |
| $\overline{PSEN}$ | 26 | 32 | **Program Store Enable output:** Read strobe to the external program memory via Port 0 and Port 2. It is activated twice each machine cycle during fetches from external program memory. When executing out of external program memory, two activations of $\overline{PSEN}$ are skipped during each access to external data memory. $\overline{PSEN}$ is not activated (remains HIGH) during no fetches from external program memory. $\overline{PSEN}$ can sink/source 8 LSTTL inputs. It can drive CMOS inputs without external pull-ups. |
| ALE/$\overline{WE}$ | 27 | 33 | **Address Latch Enable output:** Latches the LOW byte of the address during access to external memory in normal operation. It is activated every six oscillator periods except during an external data memory access. ALE/$\overline{WE}$ can sink/source 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up (note 1). |
| $\overline{EA}$ | 29 | 35 | **External Access input:** When during RESET, $\overline{EA}$ is held at a TTL HIGH level, the CPU executes out of the internal program ROM, provided the program counter is less than 32768. When $\overline{EA}$ is held at a TTL LOW level during RESET, the CPU executes out of external program memory via Port 0 and Port 2. $\overline{EA}$ is not allowed to float. $\overline{EA}$ is latched during RESET and don't care after RESET. |
| P0.0 to P0.7 | 30–37 | 36–43 | **Port 0:** 8-bit open drain bidirectional I/O port. It is also the multiplexed low-order address and data bus during accesses to external memory (AD0 to AD7). During these accesses internal pull-ups are activated. Port 0 can sink/source 8 LSTTL inputs. |
| $V_{DD1;2}$ | | | **Power supply:** +5V power supply pin during normal operation and power reduction modes. Both pins must be connected. |
| $V_{DD1}$ | 17 | 23 | Power supply pin for ports, ALE, $\overline{PSEN}$, and on-chip oscillator. |
| $V_{DD2}$ | 38 | 44 | Power supply pin for internal logic |
| | | | To avoid a latch-up effect at power-on, the voltage on any pin (at any time) must not be higher than $V_{DD}$ + 0.5V or lower than $V_{SS}$ – 0.5V, respectively. |

**NOTES:**

1. To prohibit the toggling of ALE/$\overline{WE}$ pin (RFI noise reduction) the bit RFI in the PCON register (PCON.5) must be set by software. This bit is cleared on RESET and can be cleared by software. When set, ALE/$\overline{WE}$ pin will be pulled down internally, switching an external address latch to a quiet state. The MOVX instruction will still toggle ALE/$\overline{WE}$ as a normal MOVX. ALE/$\overline{WE}$ will retain its normal HIGH value during Idle mode and a LOW value during Power-down mode while in the "RFI" mode. Additionally, during internal access ($\overline{EA}$ = 1) ALE/$\overline{WE}$ will toggle normally when the address exceeds the internal program memory size. During external access ($\overline{EA}$ = 0) ALE/$\overline{WE}$ will always toggle normally, whether the flag "RFI" is set or not.

## 6 FUNCTIONAL DESCRIPTION

### 6.1 General

The P8xCE528 is a stand-alone high-performance microcontroller designed for use in real time applications such as instrumentation, industrial control, medium to high-end consumer applications and specific automotive control applications.

In addition to the 80C51 standard functions, the device provides a number of dedicated hardware functions for these applications. The P8xCE528 is a control-oriented CPU with on-chip program and data memory. It can be extended with external program memory up to 64 kbytes. It can also access up to 64 kbytes of external data memory. For systems requiring extra capability, the P8xCE528 can be expanded using standard memories and peripherals.

The P8xCE528 has two software selectable modes of reduced activity for further power reduction: Idle and Power-down. The Idle mode freezes the CPU while allowing the RAM, timers, serial ports and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative except the WDT if it is enabled. The Power-down mode can be terminated by an external reset, a WDT overflow, and in addition, by either of the two external interrupts.

### 6.2 Memory Organization

The central processing unit (CPU) manipulates operands in three memory spaces; these are the 64 kbyte external data memory (of which the lower 256 bytes reside in the internal AUX-RAM), 512 byte internal data memory (consisting of 256 bytes standard RAM and 256 bytes AUX-RAM) and the 64 kbyte internal and external program memory.

#### 6.2.1 Program Memory

The program memory of the P8xCE528 consists of 32 kbytes of ROM on-chip, externally expandable up to 64 kbytes. If the $\overline{EA}$ pin was HIGH during RESET, the P8xCE528 executes out of the internal program memory unless the address exceeds 7FFFH. Locations 8000H through 0FFFFH are then fetched from the external program memory. If the $\overline{EA}$ pin was LOW during RESET, the P8xCE528 fetches all instructions from the external program memory. The $\overline{EA}$ input level is latched during RESET and is "Don't Care" after RESET. Figure 5 illustrates the program memory address space.

By setting a mask programmable security bit (ROM) respectively software programmable security byte (FEEPROM) the internal memory content is protected, i.e., it cannot be read out by any test mode or by any instruction in the external program memory space. The MOVC instructions are the only ones which have access to program code in the internal or external program memory. The $\overline{EA}$ input is latched during RESET and is "Don't Care" after RESET. This implementation prevents reading from internal program code by switching from external program memory to internal program memory during MOVC instruction or an instruction that handles immediate data. Table 1 lists the access to the internal and external program memory by the MOVC instructions when the security feature has been activated. If the security feature is not activated, there are no restrictions for the MOVC instructions.

**Table 1. Internal and External Program Memory Access with Activated Security Feature**

| INSTRUCTION | ACCESS TO INTERNAL PROGRAM MEMORY | ACCESS TO EXTERNAL PROGRAM MEMORY |
|---|---|---|
| MOVC in internal program memory | YES | YES |
| MOVC in external program memory | NO | YES |

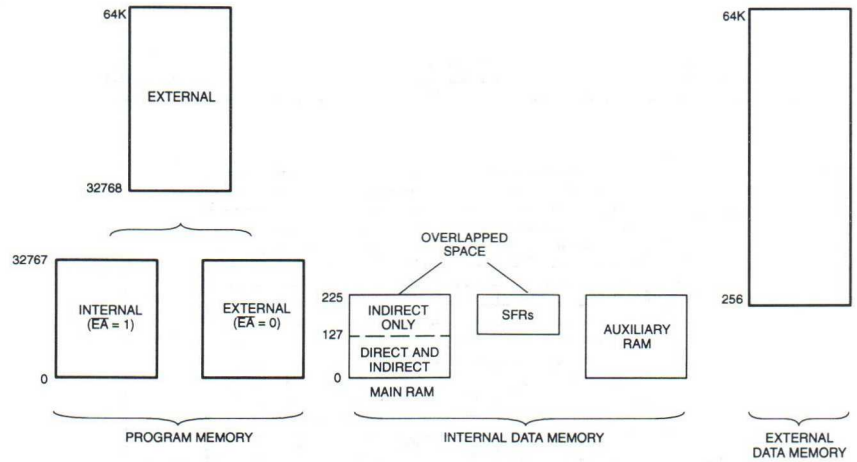



**Figure 5. Memory Map and Address Space**

6.2.2 INTERNAL DATA MEMORY

The internal data memory is divided into three physically separated parts: 256 byte of Main RAM, 256 byte of AUX-RAM, and a 128 byte special function area (SFR) see Table 2. These parts can be addressed as follows:

- Main RAM 0 to 127 can be addressed directly and indirectly as in the 80C51. Address pointers are R0 and R1 of the selected register bank.
- Main RAM 128 to 255 can only be addressed indirectly. Address pointers are R0 and R1 of the selected register bank.
- AUX-RAM 0 to 255 is indirectly addressable in the same way as the external Data Memory locations 0 to 255 with the MOVX instructions. Address pointers are R0 and R1 of the selected register bank and DPTR. When executing from internal program memory, an access to AUX-RAM 0 to 255 will not affect the Ports P0, P2, P3.6 and P3.7.
- An access to external Data Memory locations higher than 255 will be performed with the MOVX DPTR instructions in the same way as in the 80C51 structure, i.e., with P0 and P2 as data/address bus and P3.6 and P3.7 as write and read strobe signals. Note that it is impossible to access the external Data Memory with R0, R1 or DPTR < 256 as address pointer.
- The SFRs can only be addressed directly in the address range from 128 to 255 (Figure 6 illustrates the SFRs memory map).

Four 8-bit register banks occupy locations 0 through 31 in the lower RAM area. Only one of these banks may be enabled at a time. The next 16 bytes, locations 32 through 47, contain 128 directly addressable bit locations. The stack can be located anywhere in the internal 256 byte RAM. The stack depth is only limited by the available internal RAM space of 256 bytes. All registers except the Program Counter and the four 8-bit register banks reside in the SFR address space.

**Table 2. Internal Data Memory Access**

| LOCATION | ADDRESSED |
|---|---|
| Main RAM 0 to 127 | DIRECT and INDIRECT |
| AUX-RAM 0 to 255 | INDIRECT only with MOVX |
| Main RAM 128 to 255 | INDIRECT only |
| SFR 128 to 255 | DIRECT only |

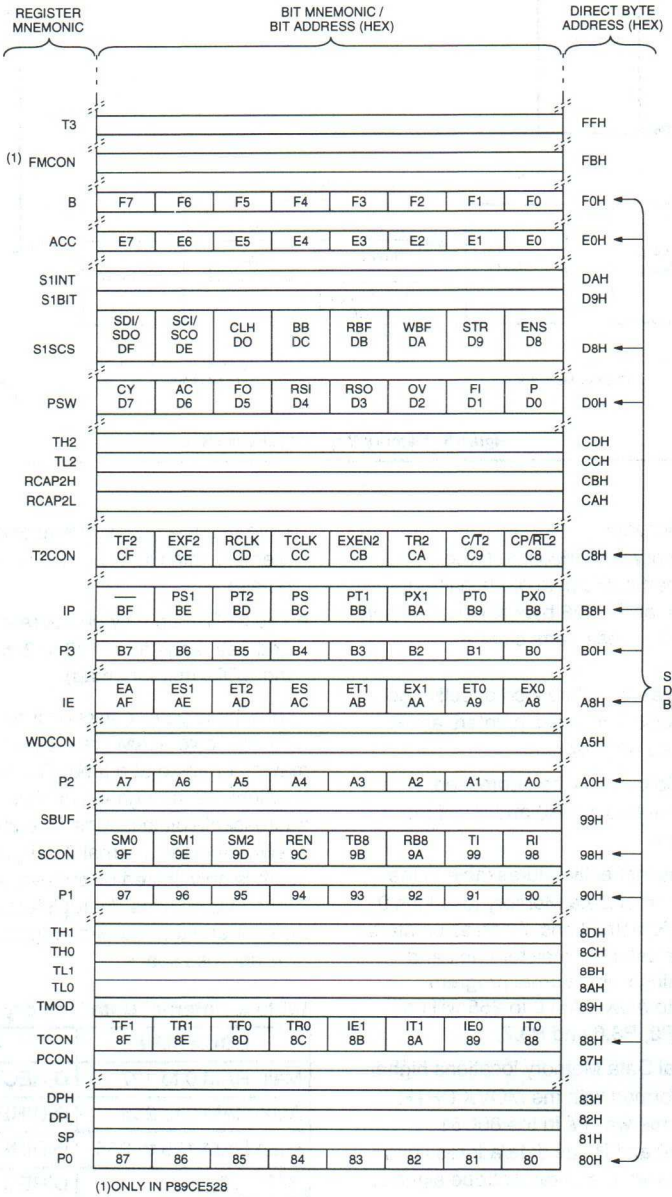



**Figure 6. Special Function Registers (SFR) Memory Map**

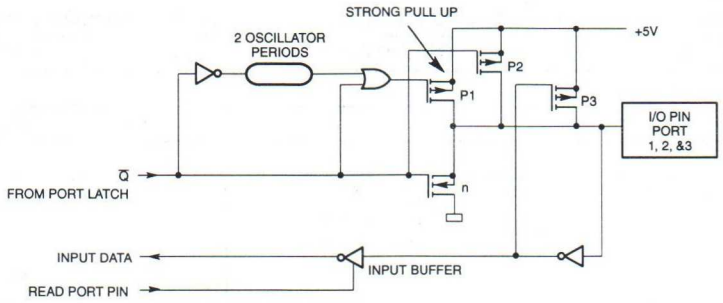



**Figure 7. I/O Buffers in the P8xCE528 (Ports 1, 2, and 3, except P1.6 and P1.7)**

**NOTE:**

For details on the Timers, I/O, UART, WatchDog Timer, $I^2C$ interface, interrupt structure and reduced power modes, please refer to the 8XC528 datasheet in this handbook.

## 7 LIMITING VALUES

In accordance with the Absolute Maximum Rating System (IEC 134).

| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| $V_{DD}$ | Supply voltage range | – | –0.5 | +6.5 | V |
| $V_I$ | All input voltages | – | –0.5 | $V_{DD}$ + 0.5 | V |
| $P_{tot}$ | Total power dissipation | Note 1 | – | 1 | W |
| $T_{stg}$ | Storage temperature range | | –65 | +150 | °C |
| $T_{amb}$ | Operating ambient temperature range: | | | | |
| | Version EBx | | 0 | +70 | °C |
| | Version EFx | | –40 | +85 | °C |

**NOTE:**

1. This value is based on the maximum allowable die temperature and the thermal resistance of the package, not on the device power consumption.

## 8 DC CHARACTERISTICS (EBx)

$V_{DD}$ = 5V ±10%; $V_{SS}$ = 0V; $T_{amb}$ = 0 to +70°C. All voltages with respect to $V_{SS}$ unless otherwise specified.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| **Supply** | | | | | |
| $V_{DD}$ | Supply voltage range | | 4.5 | 5.5 | V |
| **Supply Current** | | | | | |
| $I_{DD}$ | Operating modes | $V_{DD}$ = 5.5V; $f_{CLK}$ = 16MHz Notes 1 and 8 | | | |
| | Version P80CE528 | | – | 40 | mA |
| | Version P83CE528 | | – | 40 | mA |
| $I_{ID}$ | Idle mode | $V_{DD}$ = 5.5V; $f_{CLK}$ = 16MHz Notes 2 and 8 | | | |
| | Version P80CE528 | | – | 8 | mA |
| | Version P83CE528 | | – | 8 | mA |
| $I_{PD}$ | Power-down mode | $2V \le V_{PD} \le V_{DD\ MAX.}$ Note 3 | – | 100 | µA |
| **Inputs** | | | | | |
| $V_{IL}$ | LOW level input voltage (except $\overline{EA}$, P1.6, P1.7) | | –0.5 | $0.2V_{DD} - 0.1$ | V |
| $V_{IL1}$ | LOW level input voltage $\overline{EA}$ | | –0.5 | $0.2V_{DD} - 0.3$ | V |
| $V_{IL2}$ | LOW level input voltage P1.6, P1.7 | Note 6 | –0.5 | $0.3V_{DD}$ | V |
| $V_{IH}$ | HIGH level input voltage (except RST, XTAL1, P1.6, P1.7) | | $0.2V_{DD} + 0.9$ | $V_{DD} + 0.5$ | V |
| $V_{IH1}$ | HIGH level input voltage RST, XTAL1 | | $0.7V_{DD}$ | $V_{DD} + 0.5$ | V |
| $V_{IH2}$ | HIGH level input voltage P1.6, P1.7 | Note 6 | $0.7V_{DD}$ | 6.0 | V |
| $I_{IL}$ | Input current logic 0 Ports 1, 2, and 3 (except P1.6 and P1.7) | $V_I$ = 0.45V | – | –50 | µA |
| $I_{TL}$ | Input current HIGH-to-LOW transition Ports 1, 2, and 3 (except P1.6 and P1.7) | $V_I$ = 2.0V | – | –650 | µA |
| $I_{LI1}$ | Input leakage current Port 0, $\overline{EA}$ | $0.45 < V_I < V_{DD}$ | – | ±10 | µA |
| $I_{LI2}$ | Input leakage current P1.6 and P1.7 | $0V < V_I < 6V$ $0V < V_{DD} < 5.5V$ | – | ±10 | µA |

| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| **Outputs** | | | | | |
| $V_{OL}$ | LOW level output voltage Ports 1, 2, and 3 (except P1.6 and P1.7) | $I_{OL}$ = 1.6mA Notes 4 and 7 | – | 0.45 | V |
| $V_{OL1}$ | LOW level output voltage Port 0, ALE, $\overline{PSEN}$ | $I_{OL}$ = 3.2mA Notes 4 and 7 | – | 0.45 | V |
| $V_{OL2}$ | LOW level output voltage P1.6 and P1.7 | $I_{OL}$ = 3.0mA Note 7 | – | 0.40 | V |
| $V_{OH}$ | HIGH level output voltage Ports 1, 2, 3 | $I_{OH}$ = –60µA; $V_{DD}$ = 5V ±10% | 2.4 | – | V |
| | | $I_{OH}$ = –25µA | $0.75V_{DD}$ | – | V |
| | | $I_{OH}$ = –10µA | $0.9V_{DD}$ | – | V |
| $V_{OH1}$ | HIGH level output voltage Port 0 in external Bus mode, ALE, $\overline{PSEN}$, RST | $I_{OH}$ = –800µA; $V_{DD}$ = 5V ±10% | 2.4 | – | V |
| | | $I_{OH}$ = –300µA | $0.75V_{DD}$ | – | V |
| | | $I_{OH}$ = –80µA Note 5 | $0.9V_{DD}$ | – | V |
| $R_{RST}$ | RST pull-down resistor | | 50 | 150 | kΩ |
| $C_{I/O}$ | Capacitance of input buffer | Test frequency = 1MHz $T_{amb}$ = 25°C | – | 10 | pF |

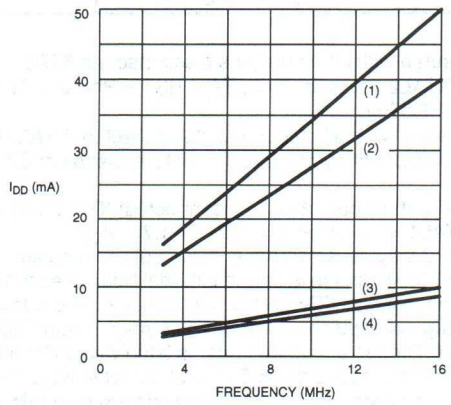
**Notes to DC Characteristics (EBx):**

1. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 5ns; $V_{IL} = V_{SS}$ + 0.5V; $V_{IH} = V_{DD}$ – 0.5V; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = P1.6 = P1.7 = $V_{DD}$; the WDT is disabled (by the external RESET).
2. The Idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 5ns; $V_{IL} = V_{SS}$ + 0.5V; $V_{IH} = V_{DD}$ = 0.5V; XTAL2 not connected; the WDT is disabled; $\overline{EA}$ = RST = $V_{SS}$; Port 0 = P1.6 = P1.7 = $V_{DD}$.
3. The Power-down current is measured with all output pins disconnected; XTAL2 not connected; the WDT is disabled; $\overline{EA}$ = RST = XTAL1 = $V_{SS}$; Port 0 = P1.6 = P1.7 = $V_{DD}$.
4. Capacitive loading on Port 0 and Port 2 may cause spurious noise pulses to be superimposed on the LOW level output voltage of ALE, Port 1 and Port 3. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make a HIGH-to-LOW transition during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE line may exceed 0.8V. In such cases it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.
5. Capacitive loading on Port 0 and Port 2 may cause the HIGH level output voltage on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{DD}$ specification when the address bits are stabilizing.
6. The input threshold voltage of P1.6 and P1.7 (SIO1) meets the $I^2C$-bus specification, so a voltage below $0.3V_{DD}$ will be recognized as a logic 0 while input above $0.7V_{DD}$ will be recognized as a logic 1.
7. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 10mA.
   Maximum $I_{OL}$ per 8-bit port:
   Port 0: 26mA
   Ports 1, 2, and 3: 15mA
   Maximum total $I_{OL}$ for all output pins: 71mA.
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
8. $I_{DD\ max.}$ at other frequencies can be derived from Figure 8, where f is the external oscillator frequency in MHz; $I_{DD\ max.}$ is given in mA.

## 9 DC CHARACTERISTICS (EFx)

$V_{DD}$ = 5V ±10%; $V_{SS}$ = 0V; $T_{amb}$ = –40 to +85°C (extended temperature range). All voltages with respect to $V_{SS}$ unless otherwise specified. DC parameters not included here are the same as for the EBx temperature range data.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| **Inputs** | | | | | |
| $V_{IL}$ | LOW level input voltage (except $\overline{EA}$, P1.6, P1.7) | | –0.5 | $0.2V_{DD} - 0.15$ | V |
| $V_{IL1}$ | LOW level input voltage $\overline{EA}$ | | –0.5 | $0.2V_{DD} - 0.35$ | V |
| $V_{IH}$ | HIGH level input voltage (except RST, XTAL1, P1.6, P1.7) | | $0.2V_{DD} + 1.0$ | $V_{DD} + 0.5$ | V |
| $V_{IH1}$ | HIGH level input voltage RST, XTAL1 | | $0.7V_{DD} + 0.1$ | $V_{DD} + 0.5$ | V |
| $I_{IL}$ | Input current logic 0 Ports 1, 2, and 3 (except P1.6 and P1.7) | $V_I$ = 0.45V | – | –75 | µA |
| $I_{TL}$ | Input current HIGH-to-LOW transition Ports 1, 2, and 3 (except P1.6 and P1.7) | $V_I$ = 2.0V | – | –750 | µA |




$V_{DD}$ = 5.5V
VALID ONLY WITHIN FREQUENCY SPECIFICATIONS OF DEVICE UNDER TEST.
(1)
(2) MAXIMUM OPERATING MODE P83CE528 AND P80CE528.
(3)
(4) MAXIMUM IDLE MODE P83CE528 AND P80CE528.

**Figure 8. Supply Current $I_{DD}$ as a Function of Frequency**

## 10 AC CHARACTERISTICS

EBx: $V_{DD}$ = 5V ±10%; $V_{SS}$ = 0V; $T_{amb}$ = 0 to +70°C; $t_{CLCL}$ min. = 63ns.
EBx: $V_{DD}$ = 5V ±10%; $V_{SS}$ = 0V; $T_{amb}$ = –40 to +85°C; $t_{CLCL}$ min. = 63ns.
All versions Exx: Cl = 100pF for Port 0, ALE and $\overline{PSEN}$; Cl = 80pF for all other outputs unless otherwise specified.
$t_{CLCL}$ min. = $1/f_{MAX}$ (maximum operating frequency).

| SYMBOL | PARAMETER | 16MHz | | 12MHz | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|---|
| | | MIN. | MAX. | MIN. | MAX. | MIN. | MAX. | |
| $1/t_{CLCL}$ | Oscillator frequency | – | – | – | – | 3.5 | 16 | MHz |
| $t_{LHLL}$ | ALE pulse duration | 85 | – | 127 | – | $2t_{CLCL}$–40 | – | ns |
| $t_{AVLL}$ | Address setup time to ALE | 8 | – | 28 | – | $t_{CLCL}$–55 | – | ns |
| $t_{LLAX}$ | Address hold time after ALE | 28 | – | 48 | – | $t_{CLCL}$–35 | – | ns |
| $t_{LLIV}$ | Time from ALE to valid instruction input | – | 150 | – | 233 | – | $4t_{CLCL}$–100 | ns |
| $t_{LLPL}$ | Time from ALE to control pulse $\overline{PSEN}$ | 23 | – | 43 | – | $t_{CLCL}$–40 | – | ns |
| $t_{PLPH}$ | Control pulse duration $\overline{PSEN}$ | 143 | – | 205 | – | $3t_{CLCL}$–45 | – | ns |
| $t_{PLIV}$ | Time from $\overline{PSEN}$ to valid instruction input | – | 83 | – | 145 | – | $3t_{CLCL}$–105 | ns |
| $t_{PXIX}$ | Input instruction hold time after $\overline{PSEN}$ | 0 | – | 0 | – | 0 | – | ns |
| $t_{PXIZ}$ | Input instruction float delay after $\overline{PSEN}$ | – | 38 | – | 59 | – | $t_{CLCL}$–25 | ns |
| $t_{AVIV}$ | Address to valid instruction input | – | 208 | – | 312 | – | $5t_{CLCL}$–105 | ns |
| $t_{PLAZ}$ | Address float time to $\overline{PSEN}$ | – | 10 | – | 10 | – | 10 | ns |
| **Data Memory** | | | | | | | | |
| $t_{RLRH}$ | $\overline{RD}$ pulse duration | 275 | – | 400 | – | $6t_{CLCL}$–100 | – | ns |
| $t_{WLWH}$ | $\overline{WR}$ pulse duration | 275 | – | 400 | – | $6t_{CLCL}$–100 | – | ns |
| $t_{RLDV}$ | $\overline{RD}$ to valid data input | – | 148 | – | 252 | – | $5t_{CLCL}$–165 | ns |
| $t_{RHDX}$ | Data hold time after $\overline{RD}$ | 0 | – | 0 | – | 0 | – | ns |
| $t_{RHDZ}$ | Data float delay after $\overline{RD}$ | – | 55 | – | 97 | – | $2t_{CLCL}$–70 | ns |
| $t_{LLDV}$ | Time from ALE to valid data input | – | 350 | – | 517 | – | $8t_{CLCL}$–150 | ns |
| $t_{AVDV}$ | Address to valid data input | – | 398 | – | 585 | – | $9t_{CLCL}$–165 | ns |
| $t_{LLWL}$ | Time from ALE to $\overline{RD}$ or $\overline{WR}$ | 138 | 238 | 200 | 300 | $3t_{CLCL}$–50 | $3t_{CLCL}$+50 | ns |
| $t_{AVWL}$ | Time from address to $\overline{RD}$ or $\overline{WR}$ | 120 | – | 203 | – | $4t_{CLCL}$–130 | – | ns |
| $t_{QVWX}$ | Data valid to $\overline{WR}$ transition | 3 | – | 23 | – | $t_{CLCL}$–60 | – | ns |
| $t_{WHQX}$ | Data hold time after $\overline{WR}$ | 13 | – | 33 | – | $t_{CLCL}$–50 | – | ns |
| $t_{RLAZ}$ | Address float delay after $\overline{RD}$ | – | 0 | – | 0 | – | 0 | ns |
| $t_{WHLH}$ | Time from $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH | 23 | 103 | 43 | 123 | $t_{CLCL}$–40 | $t_{CLCL}$+40 | ns |

| SYMBOL | PARAMETER | 16MHz | | 12MHz | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|---|
| | | MIN. | MAX. | MIN. | MAX. | MIN. | MAX. | |
| **External Clock** | | | | | | | | |
| $t_{CHCX}$ | HIGH time | 20 | – | – | – | 20 | – | ns |
| $t_{CLCX}$ | LOW time | 20 | – | – | – | 20 | – | ns |
| $t_{CLCH}$ | Rise time | – | 20 | – | – | – | 20 | ns |
| $t_{CHCL}$ | Fall time | – | 20 | – | – | – | 20 | ns |
| **UART Timing Shift Register Mode** | | | | | | | | |
| $t_{XLXL}$ | Serial port clock cycle time | 750 | – | 1000 | – | $12t_{CLCL}$ | – | ns |
| $t_{QVXH}$ | Output data setup to clock rising edge | 492 | – | 700 | – | $10t_{CLCL}-133$ | – | ns |
| $t_{XHQX}$ | Output data hold after clock rising edge | 8 | – | 50 | – | $2_{CLCL}-117$ | – | ns |
| $t_{XHDX}$ | Input data hold after clock rising edge | 0 | – | 0 | – | 0 | – | ns |
| $t_{XHDV}$ | Clock rising edge to input data valid | – | 492 | – | 700 | – | $10t_{CLCL}-133$ | ns |

## 11 I²C-bus CHARACTERISTICS (bit-level)

$t_{CLCL} = 1/f_{CLK}$ = one oscillator period at pin XTAL1. For 63ns < $t_{CLCL}$ < 286ns (16MHz > $f_{OSC}$ > 3.5MHz) the I²C interface meets the I²C-bus specification for bit rates up to 100kbit/s.

| SYMBOL | PARAMETER | INPUT | OUTPUT | I²C-bus Spec. | UNIT |
|---|---|---|---|---|---|
| **SCL Timing** | | | | | |
| $t_{HD;STA}$ | START condition hold time | ≥ 14 $t_{CLCL}$; Note 1 | Note 2 | ≥ 4.0 | µs |
| $t_{LOW}$ | SCL LOW time | ≥ 16 $t_{CLCL}$ | Note 2 | ≥ 4.7 | µs |
| $t_{HIGH}$ | SCL HIGH time | ≥ 14 $t_{CLCL}$; Note 1 | ≥ 80 $t_{CLCL}$; Note 3 | ≥ 4.0 | µs |
| $t_{RC}$ | SCL rise time | ≤ 1µs; Note 4 | Note 5 | ≤ 1.0 | µs |
| $t_{FC}$ | SCL fall time | ≤ 0.3µs; Note 4 | ≤ 0.3µs; Note 6 | ≤ 0.3 | µs |
| **SDA Timing** | | | | | |
| $t_{SU;DAT}$ | Data setup time | ≥ 250ns | Note 2 | ≥ 250 | ns |
| $t_{HD;DAT}$ | Data hold time | ≥ 0ns | Note 2 | ≥ 0 | ns |
| $t_{SU;STA}$ | Repeated START setup time | ≥ 14 $t_{CLCL}$; Note 1 | Note 2 | ≥ 4.7 | µs |
| $t_{SU;STO}$ | STOP condition setup time | ≥ 14 $t_{CLCL}$; Note 1 | Note 2 | ≥ 4.0 | µs |
| $t_{BUF}$ | Bus free time | ≥ 14 $t_{CLCL}$; Note 1 | Note 2 | ≥ 4.7 | µs |
| $t_{RD}$ | SDA rise time | ≤ 1µs; Note 4 | Note 5 | ≤ 1.0 | µs |
| $t_{FD}$ | SDA fall time | ≤ 0.3µs; Note 4 | ≤ 0.3µs; Note 6 | ≤ 0.3 | µs |

**Notes to I²C-bus Characteristics (bit-level):**

1. At $f_{CLK}$ = 3.5MHz, this evaluates to 14×286ns = 4µs, i.e., the bit-level I²C-bus interface can respond to the I²C-bus protocol for $f_{CLK}$ ≥ 3.5MHz.
2. This parameter is determined by the user software; it has to comply with the I²C-bus specification.
3. This value gives the auto-clock pulse length which meets the I²C-bus specification for the specified XTAL1 clock frequency range. Alternatively, the SCL pulse may be timed by software.
4. Spikes on SDA and SCL lines with a duration of less than 4×$t_{CLCL}$ will be filtered out.
5. The rise time is determined by the external bus line capacitance and pull-up resistor; it must be ≤ 1µs.
6. The maximum capacitance on bus lines SDA and SCL is 400pF.

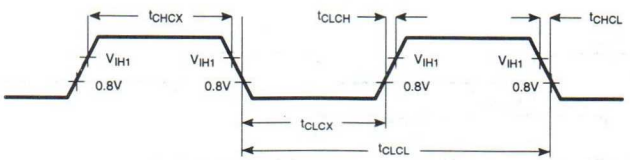
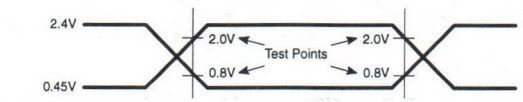
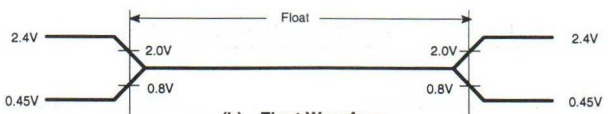



**Figure 9. External Clock Drive XTAL1**




**(a) Output Waveform**




**(b) Float Waveform**

AC testing inputs are driven at 2.4V for a logic 1 and 0.45V for a logic 0.
Timing measurements are taken at 2.0V for a logic 1 and 0.8V for logic 0. See (a).
The float state is defined as the point at which a Port 0 pin sinks 3.2mA or sources 400µA at the voltage test levels. See (b).

**Figure 10. AC Testing Input**

### 11.1 Timing Symbol Definitions

Oscillator:

$f_{CLK}$ = clock frequency
$t_{CLCL}$ = clock period

**Timing symbols (acronyms):**

Each timing symbol has five characters. The first character is always a "t" (= time). The remaining four characters of the symbol (typed in subscript), depending on their relative positions, indicate the name of a signal or the logical status of that signal. The designations are as follows:

A = address
C = clock
D = input data
H = logic level HIGH
I = instruction (program memory contents)
L = logic level LOW or ALE
P = $\overline{\text{PSEN}}$
Q = output data
R = $\overline{\text{RD}}$ signal
t = time
V = valid
W = $\overline{\text{WR}}$ signal
X = no longer a valid logic level
Z = float

Examples:

$t_{AVLL}$ = time for address valid to ALE LOW
$t_{LLPL}$ = time for ALE LOW to $\overline{\text{PSEN}}$ LOW

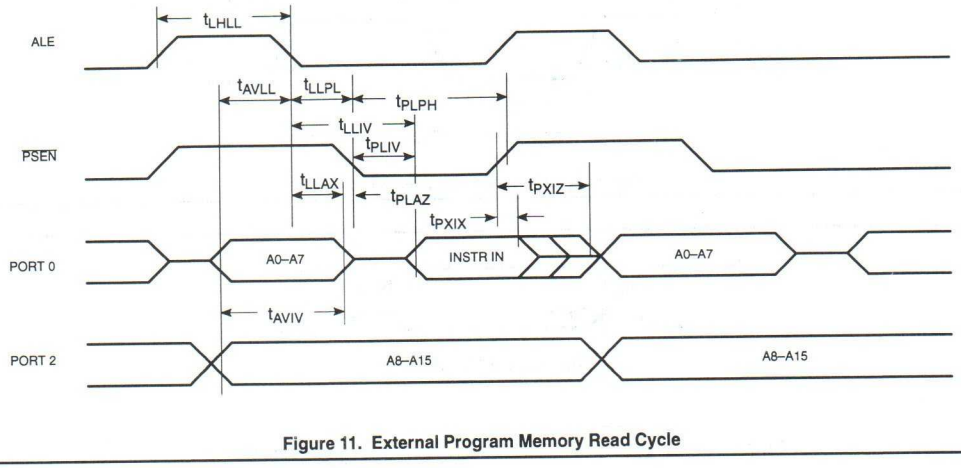



**Figure 11. External Program Memory Read Cycle**

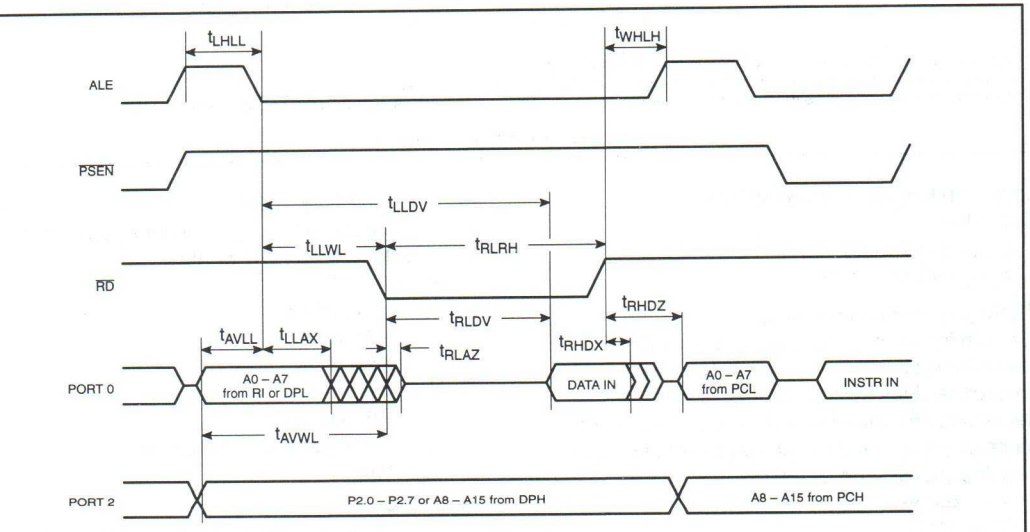



**Figure 12. External Data Memory Read Cycle**

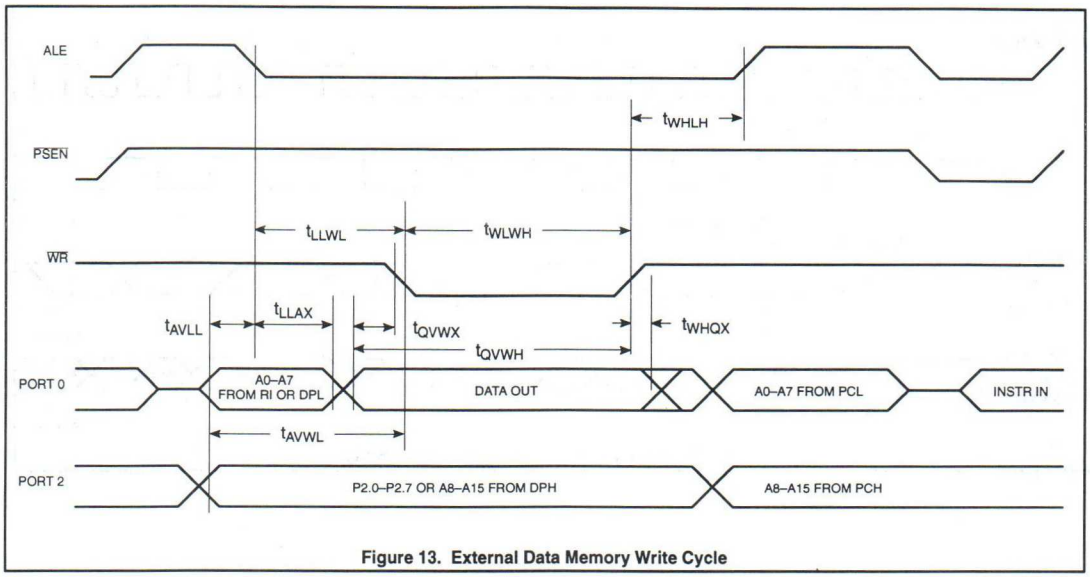



**Figure 13. External Data Memory Write Cycle**

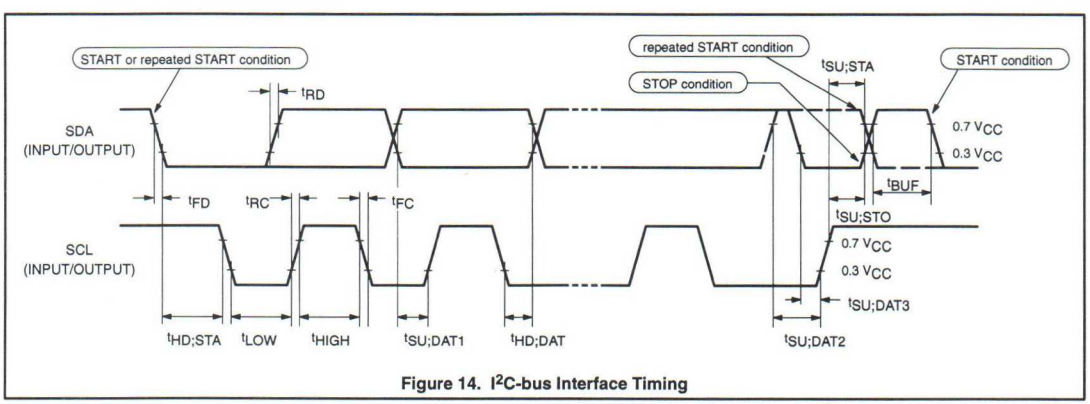



**Figure 14. I²C-bus Interface Timing**

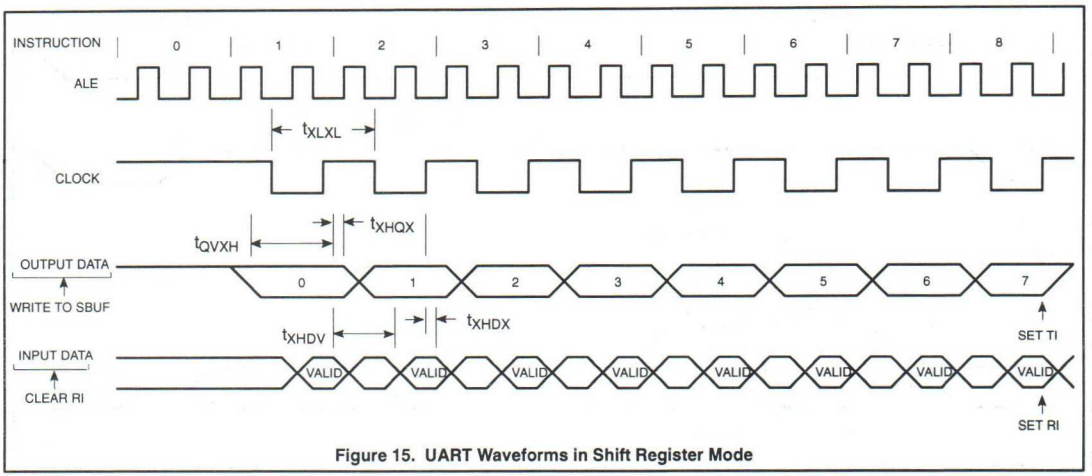



**Figure 15. UART Waveforms in Shift Register Mode**

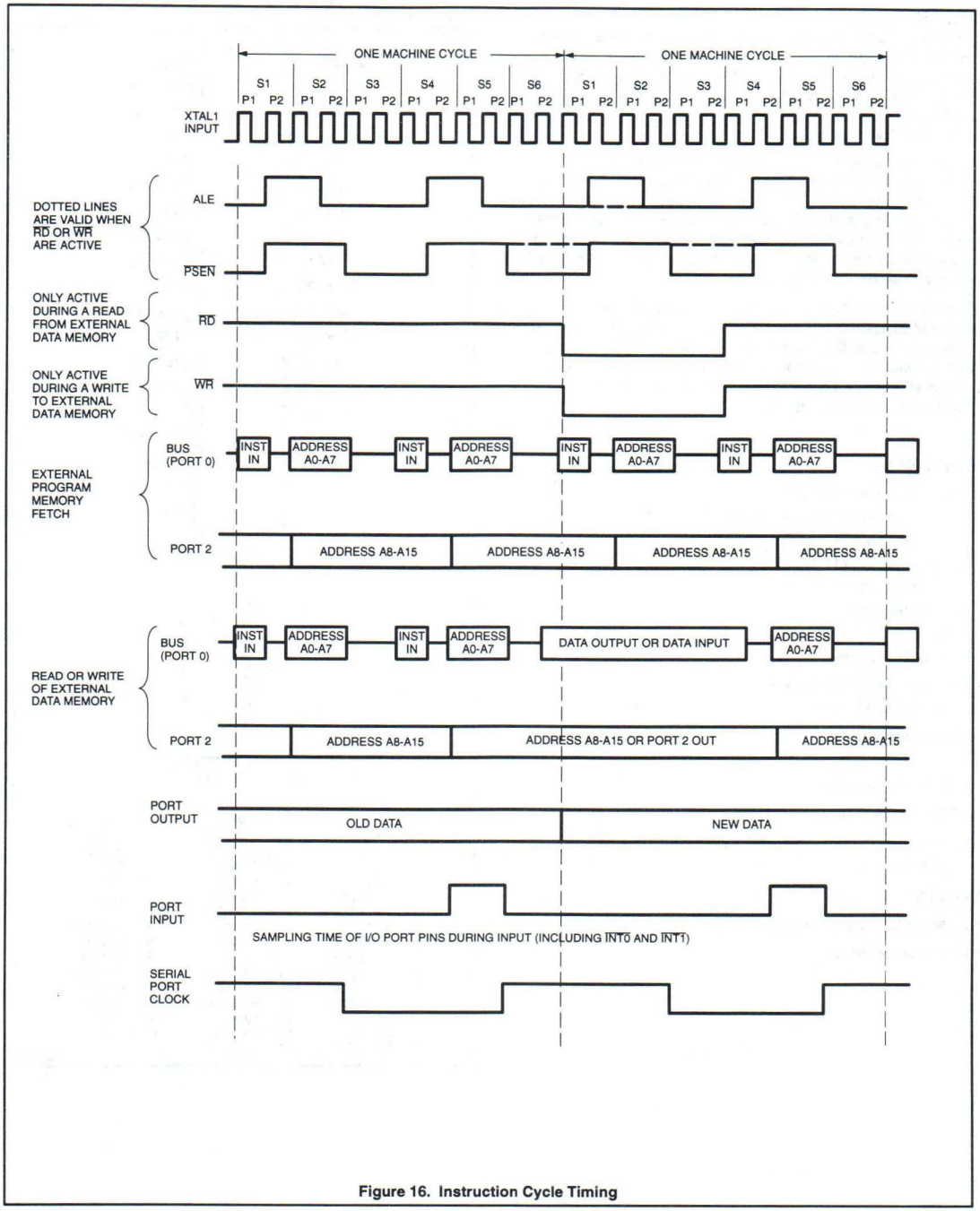



**Figure 16. Instruction Cycle Timing**

# CMOS single-chip 8-bit microcontroller with A/D and watchdog timer

# 80C550/83C550/87C550

## DESCRIPTION

The Philips 8XC550 is a high-performance microcontroller fabricated with Philips high-density CMOS technology. This Philips CMOS technology combines the high speed and density characteristics of HMOS with the low power attributes of CMOS. Philips epitaxial substrate minimizes latch-up sensitivity. The CMOS 8XC550 has the same instruction set as the 80C51.

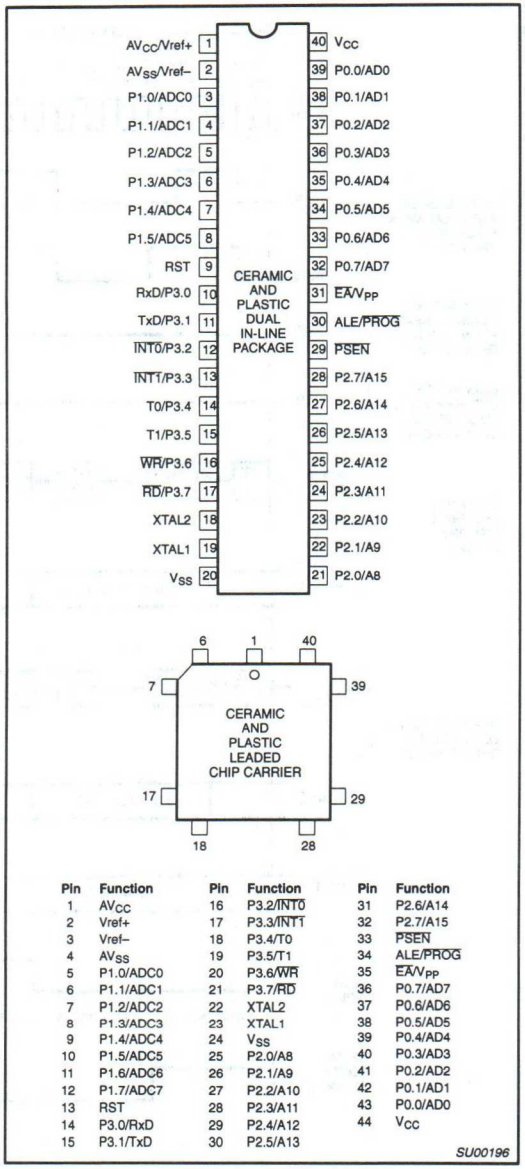
The 8XC550 contains a 4k × 8 EPROM (87C550)/ROM (83C550)/ROMless (80C550 has no program memory on-chip), a 128 × 8 RAM, 8 channels of 8-bit A/D, four 8-bit ports (port 1 is input only), a watchdog timer, two 16-bit counter/timers, a seven-source, two-priority level nested interrupt structure, a serial I/O port for either multi-processor communications, I/O expansion or full duplex UART, and an on-chip oscillator and clock circuits.

In addition, the 8XC550 has two software selectable modes of power reduction—idle mode and power-down mode. The idle mode freezes the CPU while allowing the RAM, timers, serial port, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

## FEATURES

- 80C51 based architecture
  - 4k × 8 EPROM (87C550)/ROM (83C550)
  - 128 × 8 RAM
  - Eight channels of 8-bit A/D
  - Two 16-bit counter/timers
  - Watchdog timer
  - Full duplex serial channel
  - Boolean processor
- Memory addressing capability
  - 64k ROM and 64k RAM
- Power control modes:
  - Idle mode
  - Power-down mode
- CMOS and TTL compatible
- One speed range at $V_{CC}$ = 5V ±10%
  - 3.5 to 16MHz
- Four package styles
- Extended temperature ranges
- OTP package available

## PIN CONFIGURATIONS




| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | $AV_{CC}$ | 16 | P3.2/INT0 | 31 | P2.6/A14 |
| 2 | Vref+ | 17 | P3.3/INT1 | 32 | P2.7/A15 |
| 3 | Vref– | 18 | P3.4/T0 | 33 | PSEN |
| 4 | $AV_{SS}$ | 19 | P3.5/T1 | 34 | ALE/PROG |
| 5 | P1.0/ADC0 | 20 | P3.6/WR | 35 | EA/$V_{PP}$ |
| 6 | P1.1/ADC1 | 21 | P3.7/RD | 36 | P0.7/AD7 |
| 7 | P1.2/ADC2 | 22 | XTAL2 | 37 | P0.6/AD6 |
| 8 | P1.3/ADC3 | 23 | XTAL1 | 38 | P0.5/AD5 |
| 9 | P1.4/ADC4 | 24 | $V_{SS}$ | 39 | P0.4/AD4 |
| 10 | P1.5/ADC5 | 25 | P2.0/A8 | 40 | P0.3/AD3 |
| 11 | P1.6/ADC6 | 26 | P2.1/A9 | 41 | P0.2/AD2 |
| 12 | P1.7/ADC7 | 27 | P2.2/A10 | 42 | P0.1/AD1 |
| 13 | RST | 28 | P2.3/A11 | 43 | P0.0/AD0 |
| 14 | P3.0/RxD | 29 | P2.4/A12 | 44 | $V_{CC}$ |
| 15 | P3.1/TxD | 30 | P2.5/A13 | | |

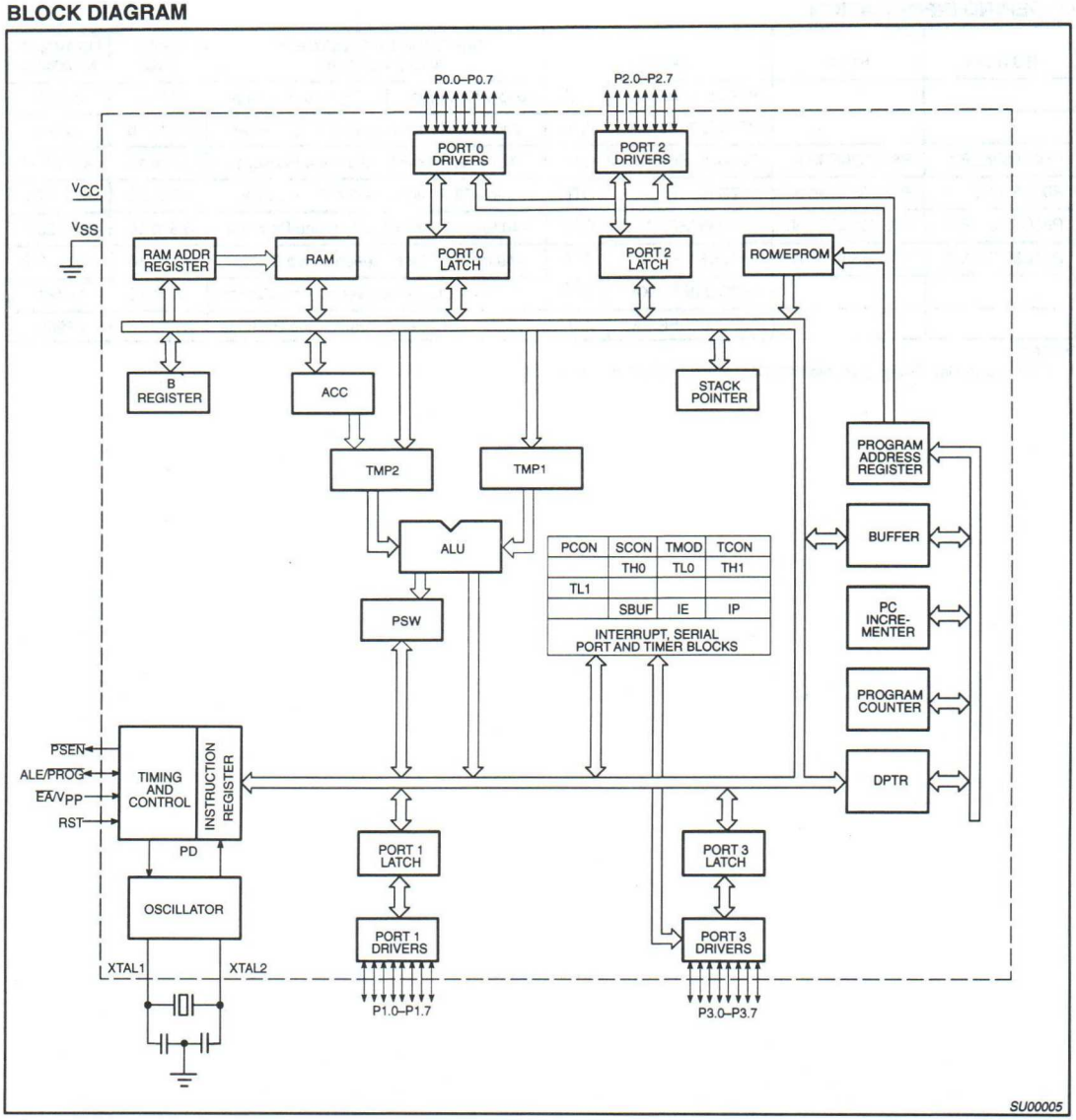
SU00196

## ORDERING INFORMATION

| ROMless | ROM | EPROM | | TEMPERATURE RANGE °C AND PACKAGE [1] | FREQ MHz | DRAWING NUMBER |
|---|---|---|---|---|---|---|
| | | P87C550EBF FA | UV | 0 to +70, Ceramic Dual In-Line Package | 3.5 to 16 | 0590B |
| | | P87C550EBL KA | UV | 0 to +70, Ceramic Leaded Chip Carrier | 3.5 to 16 | 1472A |
| P80C550EBP N | P83C550EBP N | P87C550EBP N | OTP | 0 to +70, Plastic Dual In-Line Package | 3.5 to 16 | SOT129-1 |
| P80C550EBA A | P83C550EBA A | P87C550EBA A | OTP | 0 to +70, Plastic Leaded Chip Carrier | 3.5 to 16 | SOT187-2 |
| P80C550EFP N | P83C550EFP N | P87C550EFP N | OTP | –40 to +85, Plastic Dual In-Line Package | 3.5 to 16 | SOT129-1 |
| P80C550EFA A | P83C550EFA A | P87C550EFA A | OTP | –40 to +85, Plastic Leaded Chip Carrier | 3.5 to 16 | SOT187-2 |
| | | P87C550EFL KA | UV | –40 to +85, Ceramic Leaded Chip Carrier | 3.5 to 16 | 1472A |
| | | P87C550EFF FA | UV | –40 to +85, Ceramic Dual In-Line Package | 3.5 to 16 | 0590B |

**NOTES:**
1. OTP = One Time Programmable EPROM. UV = UV Erasable EPROM.

## BLOCK DIAGRAM



## PIN DESCRIPTION

| MNEMONIC | PIN NO. DIP | PIN NO. LCC | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|
| $V_{SS}$ | 20 | 24 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| $AV_{CC}$ | 1 | 1 | I | **Analog Power Supply:** Analog supply voltage. |
| $AV_{SS}$ | 2 | 4 | I | **Analog Ground:** Analog 0V reference. |
| Vref+ | | 2 | I | **Vref:** A/D converter reference level inputs. Note that these references are combined with $AV_{CC}$ and |
| Vref– | | 3 | I | $AV_{SS}$ in the 40-pin DIP package. |
| P0.0–0.7 | 39–32 | 43–36 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification in the S87C550. External pull-ups are required during program verification. |
| P1.0–P1.7 | 3–8 | 5–12 | I | **Port 1:** Port 1 is an 8-bit input only port (6-bit in the DIP package; bits P1.6 and P1.7 are not implemented). Port 1 digital input can be read out any time. |
| ADC0–ADC7 | 3–8 | 5–12 | | **ADCx:** Inputs to the analog multiplexer input of the 8-bit A/D. There are only six A/D inputs in the DIP package. |
| P2.0–P2.7 | 21–28 | 25–32 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |
| P3.0–P3.7 | 10–17 | 14–21 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the SC80C51 family, as listed below: |
| | 10 | 14 | I | **RxD (P3.0):** Serial input port |
| | 11 | 15 | O | **TxD (P3.1):** Serial output port |
| | 12 | 16 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 17 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 18 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 19 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 20 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 21 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 13 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{CC}$. |
| ALE/$\overline{PROG}$ | 30 | 34 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. |
| $\overline{PSEN}$ | 29 | 33 | O | **Program Store Enable:** The read strobe to external program memory. When the device is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}$/$V_{PP}$ | 31 | 35 | I | **External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H to 0FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 0FFFH. For the 80C550 ROMless part, $\overline{EA}$ must be held low for the part to operate properly. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. |
| XTAL1 | 19 | 23 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 22 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**Table 1. 8XC550 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| ADAT# | A/D result | C6H | | | | | | | | | xxH |
| ADCON# | A/D control | C5H | – | – | – | ADCI | ADCS | AADR2 | AADR1 | AADR0 | xxx00000B |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes): | | | | | | | | | | |
| DPH | High byte | 83H | | | | | | | | | 00H |
| DPL | Low byte | 82H | | | | | | | | | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP*# | Interrupt priority | B8H | – | PWD | PAD | PS | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | EWD | EAD | ES | ET1 | EX1 | ET0 | EX0 | 00H |
| P0* | Port 0 | 80H | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | FFH |
| P1* | Port 1 | 90H | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | FFH |
| P2* | Port 2 | A0H | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | FFH |
| P3* | Port 3 | B0H | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B 0 | FFH |
| PCON# | Power control | 87H | SMOD | SIDL | – | – | GF1 | GF0 | PD | IDL | 00xx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| SBUF | Serial data buffer | 99H | | | | | | | | | xxH |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial port control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | 00H |
| TCON* | Timer counter/control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| TMOD | Timer/counter mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |
| TH0 | Timer 0 high byte | 8CH | | | | | | | | | 00H |
| TH1 | Timer 1 high byte | 8DH | | | | | | | | | 00H |
| TL0 | Timer 0 low byte | 8AH | | | | | | | | | 00H |
| TL1 | Timer 1 low byte | 8BH | | | | | | | | | 00H |
| | | | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 | |
| WDCON*# | Watchdog timer control | C0H | PRE2 | PRE1 | PRE0 | – | – | WDRUN | WDTOF | WDMOD | 000xx000B** |
| WDL# | Watchdog timer reload | C1H | | | | | | | | | FFH** |
| WFEED1# | Watchdog timer feed 1 | C2H | | | | | | | | | xxH |
| WFEED2# | Watchdog timer feed 2 | C3H | | | | | | | | | xxH |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
**This value is not valid for a masked ROM part (83C550) when running from internal memory ($\overline{EA}$ = 1). See data sheet for details.

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| SMOD | SIDL | X | X | GF1 | GF0 | PD | IDL |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| PCON.7 | SMOD | Double baud rate |
| PCON.6 | SIDL | Serial port idle |
| PCON.5 | X | Reserved for future use |
| PCON.4 | X | Reserved for future use |
| PCON.3 | GF1 | General purpose flag bit |
| PCON.2 | GF0 | General purpose flag bit |
| PCON.1 | PD | Power down bit |
| PCON.0 | IDL | Idle mode bit |

**NOTE:**
The PCON register is at SFR byte address 87H. Its contents following a reset are 00XX0000.

SU00197

**Figure 1. Power Control Register (PCON)**

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| X | X | X | ADCI | ADCS | AADR2 | AADR1 | AADR0 |

| INPUT CHANNEL SELECTION | | | |
|---|---|---|---|
| ADDR2 | ADDR1 | ADDR0 | INPUT PIN |
| 0 | 0 | 0 | ADC0 |
| 0 | 0 | 1 | ADC1 |
| 0 | 1 | 0 | ADC2 |
| 0 | 1 | 1 | ADC3 |
| 1 | 0 | 0 | ADC4 |
| 1 | 0 | 1 | ADC5 |
| 1 | 1 | 0 | ADC6 |
| 1 | 1 | 1 | ADC7 |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| ADCON.7 | — | Not used |
| ADCON.6 | — | Not used |
| ADCON.5 | — | Not used |
| ADCON.4 | ADCI | ADC Interrupt flag.<br>This flag is set when an ADC conversion result is ready to be read. An interrupt is invoked if the A/D interrupt is enabled. The flag must be cleared by software. It cannot be set by software. |
| ADCON.3 | ADCS | ADC Start and Status.<br>Setting this flag starts an A/D conversion. The ADC logic insures that this signal is high while the ADC is busy. On completion of the conversion, ADCS is reset at the same time the interrupt flag ADCI is set. ADCS cannot be reset by software. |
| ADCON.2 | ADDR2 | Analog Input Select 2 |
| ADCON.1 | ADDR1 | Analog Input Select 1 |
| ADCON.0 | ADDR0 | Analog Input Select 0 |

SU00198

**Figure 2. A/D Control Register (ADCON)**

## A/D CONVERTER

The analog input circuitry consists of an 8-input analog multiplexer and an analog-to-digital converter with 8-bit resolution. In the LCC package, the analog reference voltage and analog power supplies are connected via separate input pins; in the DIP package, Vref+ is combined with $AV_{CC}$ and Vref– is combined with $AV_{SS}$. The analog inputs are alternate functions to port 1, which is an input only port. Digital input to port 1 can be read any time during an A/D conversion. Care should be exercised in mixing analog and digital signals on port 1, because cross talk from the digital input signals can degrade the A/D conversion accuracy of the analog input. An A/D conversion requires 40 machine cycles.

The A/D converter is controlled by the ADCON special function register. The input channel to be converted is selected by the analog multiplexer by setting ADCON register bits, ADDR2–ADDR0 (see Figure 2). These bits can only be changed when ADCI and ADCS are both low.

The completion of the 8-bit ADC conversion is flagged by ADCI in the ADCON register and the result is stored in the special function register ADAT.

An ADC conversion in progress is unaffected by a software ADC start. The result of a completed conversion remains unaffected provided ADCI remains at a logic 1. While ADCS is a logic 1 or ADCI is a logic 1, a new ADC START will be blocked and consequently lost. An A/D conversion in progress will be aborted when the idle or power-down mode is entered. The result of a completed conversion (ADCI = logic 1) remains unaffected when entering the idle mode, but will be lost if power-down mode is entered. See Figure 3 for the A/D input equivalent circuit.

The analog input pins ADC0-ADC7 may still be used as digital inputs. The analog input channel that is selected by the ADDR2-ADDR0 bits in ADCON cannot be used as a digital input. Reading the selected A/D channel as a digital input will always return a 1. The unselected A/D inputs may always be used as digital inputs.

On RESET the A/D port pins are set to the Digital mode and will work as a normal port and need no further initialization. To use the A/D converter a single byte should be written to ADCON which selects the A/D mux and concurrently sets the ADCS bit to start the A/D conversion. The 40 machine cycles of the A/D conversion include time for signal settling after the mux is selected and before the Sample and Hold procedure is completed.

The circuitry which disables the digital buffer from the port pin is updated at the start of an A/D conversion by setting the ADCS bit in ADCON. After powerup, problems will occur the first time that ADCON is written to if ADCS is not set; in this case, the digital signal disable registers contain random data and some o the 8 port pins will have their digital buffers disabled. When read, these disabled buffers will ignore their input and only return a 1. This condition will be corrected by writing a 1 to ADCS in ADCON which starts and A/D conversion.

Thus, there are two operating modes:

1. DIGITAL ONLY - No Analog inputs are used and ADCON is never written to. In this case pins ADC0-ADC7 are configured as digital inputs.
2. A/D CONVERTER USED - The input multiplexer select field must be written to and ADCS must be set in ADCON. This allows unselected A/D inputs to be used as digital inputs.

**ADCON Register**

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| X | X | X | ADCI | SDCS | AADR2 | AADR1 | AADR0 |

| ADCI | ADCS | Operation |
|---|---|---|
| 0 | 0 | ADC not busy, a conversion can be started. |
| 0 | 1 | ADC busy, start of a new conversion is blocked. |
| 1 | 0 | Conversion completed, start of a new is blocked. |
| 1 | 1 | Not possible. |

| INPUT CHANNEL SELECTION | | | |
|---|---|---|---|
| ADDR2 | ADDR1 | ADDR0 | INPUT PIN |
| 0 | 0 | 0 | P1.0 |
| 0 | 0 | 1 | P1.1 |
| 0 | 1 | 0 | P1.2 |
| 0 | 1 | 1 | P1.3 |
| 1 | 0 | 0 | P1.4 |
| 1 | 0 | 1 | P1.5 |
| 1 | 1 | 0 | P1.6* |
| 1 | 1 | 1 | P1.7* |

*Not present on 40-pin DIP versions.

| Symbol | Position | Function |
|---|---|---|
| ADCI | ADCON.4 | ADC interrupt flag. This flag is set when an ADC conversion is complete. If IE.5 = 1, an interrupt is requested when ADCI = 1. The ADCI flag must be cleared by software after A/D data is read, before the next conversion can begin. |
| ADCS | ADCON.3 | ADC start and status. Setting this bit starts an A/D conversion. Once set, ADCS remains high throughout the conversion cycle. On completion of the conversion, it is reset at the same time the ADCI interrupt flag is set. ADCS cannot be reset by software. |
| AADR2 | ADCON.2 | Analog input selects. |
| AADR1 | ADCON.1 | Binary coded address |
| AADR0 | ADCON.0 | selects one of the five analog input port pins of P1 to be input to the converter. It can only be changed when ADCI and ADCS are both low. AADR2 is the most significant bit. |

**Sample A/D Routines**

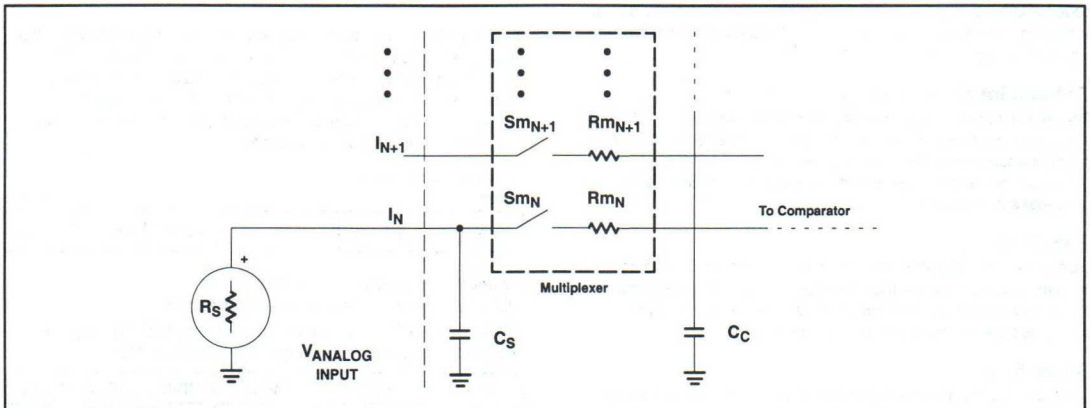
The following routines demonstrate two methods of operating the A/D converter. The first method uses polling to determine when the A/D conversion is complete. The second method uses the A/D interrupt to flag the end of conversion.

The routine ReadAD will start a read of the A/D channel identified by R7, and wait for the conversion to complete, polling the A/D interrupt flag. The result is returned in the accumulator.

```
ReadAD: MOV  A,#08h          ;Basic A/D start command.
        ORL  A,R7            ;Add channel # to be read.
        MOV  ADCON,A;        ;Start A/D.
ADLoop: MOV  A,ADCON         ;Get A/D status.
        JNB  ACC.4,ADLoop;Wait for ADCI (A/D ;finished).
        MOV  A,ADAT          ;Get conversion result
        MOV  ADCON,#0        ;Clear ADCI.
        RET
```

The routine StartAD will start a read of the A/D channel identified by R7 and exit back to the calling program. When the conversion is complete, the A/D interrupt occurs, calling the A/D interrupt service routine. The result of the conversion is returned in register R6.

```
StartAD: MOV  A,#08h         ;Basic A/D start command.
         ORL  A,R7           ;Add channel # to be read.
         MOV  ADCON,A        ;Start A/D.
         RET
          .
          .
          .
         ORG  2Bh            ;A/D interrupt address.
ADInt:   MOV  R6,ADAT        ;Get conversion result.
         MOV  ADCON,#0       ;Clear ADCI.
         RETI
```




Rm = 0.5 - 3 kΩ
CS + CC = 15pF maximum
RS = Recommended < 9.6 kΩ for 1 LSB @ 12MHz

**NOTE:**
Because the analog to digital converter has a sampled-data comparator, the input looks capacitive to a source. When a conversion is initiated, switch Sm closes for 8tcy (8µs @ 12MHz crystal frequency) during which time capacitance Cs + Cc is charged. It should be noted that the sampling causes the analog input to present a varying load to an analog source.

SU00199

**Figure 3. A/D Input: Equivalent Circuit**

## A/D CONVERTER PARAMETER DEFINITIONS

The following definitions are included to clarify some specifications given and do not represent a complete set of A/D parameter definitions.

### Absolute Accuracy Error

Absolute accuracy error of a given output is the difference between the theoretical analog input voltage to produce a given output and the actual analog input voltage required to produce the same code. Since the same output code is produced by a band of input voltages, the "required input voltage" is defined as the midpoint of the band of input voltage that will produce that code. Absolute accuracy error not specified with a code is the maximum over all codes.

### Nonlinearity

If a straight line is drawn between the end points of the actual converter characteristics such that zero offset and full scale errors are removed, then non-linearity is the maximum deviation of the code transitions of the actual characteristics from that of the straight line so constructed. This is also referred to as relative accuracy and also integral non-linearity.

### Differential Non-Linearity

Differential non-linearity is the maximum difference between the actual and ideal code widths fo the converter. The code widths are the differences expressed in LSB between the code transition points, as the input voltage is varied through the range for the complete set of codes.

### Gain Error

Gain error is the deviation between the ideal and actual analog input voltage required to cause the final code transition to a full-scale output code after the offset error has been removed. This may sometimes be referred to as full scale error.

### Offset Error

Offset error is the difference between the actual input voltage that causes the first code transition and the ideal value to cause the first code transition. This ideal value is 1/2 LSB above $V_{ref-}$.

### Channel to Channel Matching

Channel to channel matching is the maximum difference between the corresponding code transitions of the actual characteristics taken from different channels under the same temperature, voltage and frequency conditions.

### Crosstalk

Crosstalk is the measured level of a signal at the output of the converter resulting from a signal applied to one deselected channel.

### Total Error

Maximum deviation of any step point from a line connecting the ideal first transition point to the ideal last transition point.

### Relative Accuracy

Relative accuracy error is the deviation of the ADC's actual code transition points from the ideal code transition points on a straight line which connects the ideal first code transition point and the final code transition point, after nullifying offset error and gain error. It is generally expressed in LSBs or in percent of FSR.

## WATCHDOG TIMER

The purpose of the watchdog timer is to reset the microcontroller within a reasonable amount of time if it enters an erroneous state, possibly due to a programming error, electrical noise, or RFI. When enabled, the watchdog circuit will generate a system reset if the user program fails to "feed" (or reload) the watchdog within a predetermined amount of time.

The watchdog timer implemented on the 8XC550 has a programmable interval and can thus be fine tuned to a particular application. If the watchdog function is not used, the timer may still be used as a versatile general purpose timer.

The watchdog function consists of a programmable 13-bit prescaler, and an 8-bit main timer. The main timer is clocked by a tap taken from one of the top 8 bits of the prescaler. The prescaler is incremented once every machine cycle, or 1/12 of the oscillator frequency. Thus, the main counter can be clocked as often as once every 64 machine cycles or as seldom as once every 8192 machine cycles.

When clocked, the main counter decrements. If the main watchdog counter reaches zero, a system reset will occur. To prevent the watchdog timer from under-flowing, the watchdog must be fed before it counts down to zero. When the watchdog is fed, the contents of the WDL register are loaded into the main watchdog counter and the prescaler is cleared.

**WDCON Register**

MSB LSB

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| PRE2 | PRE1 | PRE0 | X | X | WDRUN | WDTOF | WDMOD |

| Symbol | Position | Function |
|---|---|---|
| WDCON.7 | PRE2 | Prescaler select (read/write). |
| WDCON.6 | PRE1 | These bits select theprescaler divide ratio |
| WDCON.5 | PRE0 | according to the following table: |

| PRE2 | PRE1 | PRE0 | DIVISOR (FROM $f_{OSC}$) |
|---|---|---|---|
| 0 | 0 | 0 | 12 × 64 |
| 0 | 0 | 1 | 12 × 64 × 2 |
| 0 | 1 | 0 | 12 × 64 × 4 |
| 0 | 1 | 1 | 12 × 64 × 8 |
| 1 | 0 | 0 | 12 × 64 × 16 |
| 1 | 0 | 1 | 12 × 64 × 32 |
| 1 | 1 | 0 | 12 × 64 × 64 |
| 1 | 1 | 1 | 12 × 64 × 128 |

| | | |
|---|---|---|
| WDCON.4 | – | Not used |
| WDCON.3 | – | Not used |
| WDCON.2 | WDRUN | Run control (read/write). This bit turns the timer on (WDRUN = 1) or off (WDRUN = 0) if the timer mode has been selected. |
| WDCON.1 | WDTOF | Timeout flag (read/write). This bit is set when the watchdog timer underflows. It is cleared by an external reset and can be cleared by software. |
| WDCON.0 | WDMOD | Mode selection (read/write). When WDMOD = 1, the watchdog is selected; when WDMOD = 0, the timer is selected. Selecting the watchdog mode automatically disables power-down mode. WDMOD is cleared by external reset. Once the watchdog mode is selected, this bit can only be cleared by writing a 0 to this bit and then performing a feed operation. |

A very specific sequence of events must take place to feed the watchdog timer; it cannot be fed accidentally by a runaway program. The following routines demonstrate setting up and feeding the watchdog timer. These routines apply to all versions of the 8XC550 except the ROM part when running from internal program memory.

This routine sets up and starts the watchdog timer. This is not necessary for internal ROM operation, because setup of the watchdog timer on masked ROM parts is accomplished directly via ROM mask options.

```
SetWD:  MOV   WDL,#0FFh    ;Set watchdog reload value.
        MOV   WDCON,#0E5;Set up timer prescaler, mode, and
                           ;run bits.
        ACALL FeedWD       ;Start watchdog with a feed
                           ;operation.
        RET
```

This routine executes a watchdog timer feed operation, causing the timer to reload from WDL. Interrupts must be disabled during this operation due to the fact that the two feed registers must be loaded on consecutive instruction cycles, or a system reset will occur immediately.

```
FeedWD:CLR  EA             ;This sequence must not be
                           ;interrupted.
       MOV  WFEED1,#0A5h;First instruction of feed sequence.
       MOV  WFEED2,#05Ah;Second instruction of feed
                           ;sequence.
       SETB EA             ;Turn interrupts back on.
       RET
```

An interrupt is available to allow the watchdog timer to be used as a general purpose timer in applications where the watchdog function is not needed. The timer operates in the same manner when used as a general purpose timer except that the timer interrupt is generated on timer underflow instead of a chip reset. Refer to the 87C550 data sheet for additional information on watchdog timer operation.

## Programming the Watchdog Timer

Both the EPROM and ROM devices have a set of SFRs for holding the watchdog autoload values and the control bits. The watchdog time-out flag is present in the watchdog control register and operates the same in all versions. In the EPROM device, the watchdog parameters (autoload value and control) are always taken from the SFRs. In the ROM device, the watchdog parameters can be mask programmed or taken from the SFRs. The selection to take the watchdog parameters from the SFRs or from the mask programmed values is controlled by EA (external access). When EA is high (internal ROM access), the watchdog parameters are taken from the mask programmed values. If the watchdog is masked programmed to the timer mode, then the autoload values and the pre-scaler taps are taken from the SFRs. When EA is low (external access), the watchdog parameters are taken from the SFRs. The user should be able to leave code in his program which initializes the watchdog SFRs even though he has migrated to the mask ROM part. This allows no code changes from EPROM prototyping to ROM coded production parts.

## Watchdog Detailed Operation

**EPROM Device (and ROMless Operation: EA = 0)**

In the ROMless operation (ROM part, EA = 0) and in the EPROM device, the watchdog operates in the following manner.

Whether the watchdog is in the watchdog or timer mode, when external RESET is applied, the following takes place:

- Watchdog mode bit set to timer mode.
- Watchdog run control bit set to OFF.
- Autoload register set to FF (max count).
- Watchdog time-out flag cleared.
- Prescaler is cleared.
- Prescaler tap set to the highest divide.
- Autoload takes place.

The watchdog can be fed even though it is in the timer mode.

Note that the operational concept is for the watchdog mode of operation, when coming out of a hardware reset, the software should load the autoload registers, set the mode to watchdog, and then feed the watchdog (cause an autoload). The watchdog will now be starting at a known point.

If the watchdog is in the watchdog mode and running and happens to underflow at the time the external RESET is applied, the watchdog time-out flag will be cleared.

When the watchdog is in the watchdog mode and the watchdog underflows, the following action takes place:

- Autoload takes place.
- Watchdog time-out flag is set
- Timer mode interrupt flag unchanged.
- Mode bit unchanged.
- Watchdog run bit unchanged.
- Autoload register unchanged.
- Prescaler tap unchanged.
- All other device action same as external reset.

Note that if the watchdog underflows, the program counter will start from 00H as in the case of an external reset. The watchdog time-out flag can be examined to determine if the watchdog has caused the reset condition. The watchdog time-out flag bit can be cleared by software.

When the watchdog is in the timer mode and the timer software underflows, the following action takes place:

- Autoload takes place.
- Watchdog time-out flag is set
- Mode bit unchanged.
- Watchdog run bit unchanged.
- Autoload register unchanged.
- Prescaler tap unchanged.

The timer mode interrupt flag is cleared when the interrupt routine is invoked. This bit can also be cleared directly by software without a software feed operation.

**Mask ROM Device (EA = 1)**

In the mask ROM device, the watchdog mode bit (WDMOD) is mask programmed and the bit in the watchdog command register is read only and reflects the mask programmed selection. If the mask programmed mode bit selects the timer mode, then the watchdog run bit (WDRUN) operates as described under EPROM Device. If the mask programmed bit selects the watchdog mode, then the watchdog run bit has no effect on the timer operation.

### Watchdog Function

The watchdog consists of a programmable prescaler and the main timer. The prescaler derives its clock from the on-chip oscillator. The prescaler consists of a divide by 12 followed by a 13 stage counter with taps from stage 6 through stage 13. The tap selection is programmable. The watchdog main counter is a down counter clocked (decremented) each time the programmable prescaler underflows. The watchdog generates an underflow signal (and is autoloaded) when the watchdog is at count 0 and the clock to decrement the watchdog occurs. The watchdog is 8 bits long and the autoload value can range from 0 to FFH. (The autoload value of 0 is permissible since the prescaler is cleared upon autoload).

This leads to the following user design equations. Definitions: $t_{OSC}$ is the oscillator period, N is the selected prescaler tap value, W is the main counter autoload value, $t_{MIN}$ is the minimum watchdog time-out value (when the autoload value is 0), $t_{MAX}$ is the maximum time-out value (when the autoload value is FFH), $t_D$ is the design time-out value.

$t_{MIN} = t_{OSC} \times 12 \times 64$
$t_{MAX} = t_{MIN} \times 128 \times 256$
$t_D = t_{MIN} \times 2^{PRESCALER} \times W$
(where prescaler = 0, 1, 2, 3, 4, 5, 6, or 7)

Note that the design procedure is anticipated to be as follows. A $t_{MAX}$ will be chosen either from equipment or operation considerations and will most likely be the next convenient value higher than $t_D$. (If the watchdog were inadvertently to start from FFH, an overflow would be guaranteed, barring other anomalies, to occur within $t_{MAX}$). Then the value for the prescaler would be chosen from:

$$\text{prescaler} = \log2\ (t_{MAX} / (t_{OSC} \times 12 \times 256)) - 6$$

This then also fixes $t_{MIN}$. An autoload value would then be chosen from:

$$W = t_D / t_{MIN} - 1$$

The software must be written so that a feed operation takes place every $t_D$ seconds from the last feed operation. Some tradeoffs may need to be made. It is not advisable to include feed operations in minor loops or in subroutines unless the feed operation is a specific subroutine.

### Interrupts

The 8XC550 interrupt structure is a seven-source, two-priority level interrupt system similar to that of the standard 80C51 microcontroller. The interrupt sources are listed below in the order of their internal polling sequence. This is the order in which simultaneous interrupts of the same priority level would be serviced.

**Interrupt Priorities**

| PRIORITY | SOURCE | VECTOR ADDRESS | FUNCTION |
|---|---|---|---|
| Highest | INT0 | 0003H | External interrupt 0 |
| | TF0 | 000BH | Counter/timer 0 overflow |
| | INT1 | 0013H | External interrupt 1 |
| | TF1 | 001BH | Counter/timer 1 overflow |
| | TI & RI | 0023H | Serial port transmit/receive |
| | ADCI | 002BH | A/D converter conversion complete |
| Lowest | WDTOF | 0033H | Watchdog timer overflow (only when not in watchdog mode) |

### Interrupt Control Registers

The standard 80C51 interrupt enable and priority registers have been modified slightly to take into account the additional interrupt sources of the 8XC550.

**Interrupt Enable Register**

MSB LSB

| EA | EWD | EAD | ES | ET1 | EX1 | ET0 | EX0 |
|---|---|---|---|---|---|---|---|

| Symbol | Position | Function |
|---|---|---|
| EA | IE.7 | Global interrupt enable |
| EWD | IE.6 | Watchdog timer overflow |
| EAD | IE.5 | A/D conversion complete |
| ES | IE.4 | Serial port transmit or receive |
| ET1 | IE.3 | Timer 1 overflow |
| EX1 | IE.2 | External interrupt 1 |
| ET0 | IE.1 | Timer 0 overflow |
| EX0 | IE.0 | External interrupt 0 |

**Interrupt Priority Register**

MSB LSB

| – | PWD | PAD | PS | PT1 | PX1 | PT0 | PX0 |
|---|---|---|---|---|---|---|---|

| Symbol | Position | Function |
|---|---|---|
| PWD | IP.6 | Watchdog timer |
| PAD | IP.5 | A/D conversion |
| PS | IP.4 | Serial port interrupt |
| PT1 | IP.3 | Timer 1 interrupt |
| PX1 | IP.2 | External interrupt 1 |
| PT0 | IP.1 | Timer 0 interrupt |
| PX0 | IP.0 | External interrupt 0 |

### Power-Down and Idle Modes

The 8XC550 includes the standard 80C51 power-down and idle modes of reduced power consumption. In addition, the 8XC550 includes an option to separately turn off the serial port for extra power savings when it is not needed. Also, the individual functional blocks such as the counter/timers are automatically disabled when they are not running. This actually turns off the clocks to the block in question, resulting in additional power savings. Note that when the watchdog timer is operating, the processor is inhibited from entering the power-down mode. This is due to the fact that the oscillator is stopped in the power-down mode, which would effectively turn off the watchdog timer. In keeping with the purpose of the watchdog timer, the processor is prevented from accidentally entering power-down due to some erroneous operation.

**Power Control Register**

MSB LSB

| SMOD | SIDL | – | – | GF1 | GF0 | PD | IDL |
|---|---|---|---|---|---|---|---|

| Symbol | Position | Function |
|---|---|---|
| SMOD | PCON.7 | Double baud rate bit. When set to a 1 and Timer 1 is used to generate baud rate, and the serial port is used in modes 1, 2, or 3. |
| SIDL | PCON.6 | Separately idles the serial port for additional power savings. |
| – | PCON.5 | Reserved |
| – | PCON.4 | Reserved |
| GF1 | PCON.3 | General-purpose flag bit. |
| GF0 | PCON.2 | General-purpose flag bit. |
| PD | PCON.1 | Power-down bit. Starting this bit activates power-down operation. |
| IDL | PCON.0 | Idle mode bit. Setting this bit activates idle mode operation. |

If 1s are written to PD and IDL at the same time, PD takes precedence.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the Block Diagram, page 3-452).

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals except the A/D stay active. the instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. An A/D conversion in progress will be aborted when idle mode is entered. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

### Programmable Idle Modes

The programmable idle modes have been dispersed throughout the functional blocks. Each block has its own ability to be disabled. For example, if timer 0 is not commanded to be running (TR = 0), then the clock to the timer is disabled resulting in an idle mode power saving. An additional idle control bit has been added to the serial communications port.

### A/D Operation in Idle Mode

When in the idle mode, the A/D converter will be disabled. However, the current through the $V_{REF}$ pins will be present and will not be reduced internally in either the idle or the power-down modes. It is the responsibility of the user to disconnect $V_{REF}$ to reduce power supply current.

MSB LSB

| PRE2 | PRE1 | PRE0 | X | X | WDRUN | WDTOF | WDMOD |
|---|---|---|---|---|---|---|---|

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| WDCON.7 | PRE2 | Prescaler Select (Read/Write). |
| WDCON.6 | PRE1 | Prescaler Select (Read/Write). |
| WDCON.5 | PRE0 | Prescaler Select (Read/Write).<br>Thses bits select the prescaler divide ratio according to the following table: |

| PRE2 | PRE1 | PRE0 | DIVISOR (from $f_{OSC}$) |
|---|---|---|---|
| 0 | 0 | 0 | 12 X 64 |
| 0 | 0 | 1 | 12 X 64 X 2 |
| 0 | 1 | 0 | 12 X 64 X 4 |
| 0 | 1 | 1 | 12 X 64 X 8 |
| 1 | 0 | 0 | 12 X 64 X 16 |
| 1 | 0 | 1 | 12 X 64 X 32 |
| 1 | 1 | 0 | 12 X 64 X 64 |
| 1 | 1 | 1 | 12 X 64 X 128 |

| | | |
|---|---|---|
| WDCON.4 | — | Not used. |
| WDCON.3 | — | Not used. |
| WDCON.2 | WDRUN | Run Control (Read/Write).<br>This bit turns the timer on (WDRUN = 1) or off (WDRUN = 0) if the timer mode has been selected. |
| WDCON.1 | WDTOF | Timeout Flag (Read/Write).<br>This bit is set when the watchdog timer underflows. It is cleared by an external reset and can be cleared by software. |
| WDCON.0 | WDMOD | Mode Selection (Read/Write).<br>When WDMOD = 1, the watchdog mode is selected; when WDMOD = 0, the timer mode is selected. Selecting the watchdog mode automatically disables power-down mode. WDMOD is cleared by external reset. Once the watchdog mode is selected, this bit can only be cleared by writing a 0 to this bit and then performing a feed operation. |

SU00200

**Figure 4. Watchdog Control Register (WDCON)**

## DESIGN CONSIDERATIONS

At power-on, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up.

When the idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when idle is terminated by reset, the instruction following the one that invokes idle should not be one that writes to a port pin or to external memory. Table 2 shows the state of I/O ports during low current operating modes.

### Encryption Table

The encryption table is a feature of the 83C550 and 87C550 that protects the code from being easily read by anyone other than the programmer. The encryption table is 32 bytes of code that are exclusive NORed with the program code data as it is read out. The first byte is XNORed with the first location read, the second with the second read, etc.

After the encryption table has been programmed, the user has to know its contents in order to correctly decode the program code data. The encryption table itself cannot be read out.

For the EPROM (87C550) part, the encryption table is programmed in the same manner as the program memory, but using the "Pgm Encryption Table" levels specified in Table 4. After the encryption table is programmed, verification cycles will produce only encrypted information.

For the ROM part (83C550) the encryption table information is submitted with the ROM code as shown in Table 3.

### Security Bits

There are two security bits on the 83C550 and 87C550 that, when set, prevent the program data memory from being read out or programmed further.

After the first security bit is programmed, the external MOVC instruction is disabled, and for the 87C550, further programming of the code memory or the encryption table is disabled. The other security bit can of course still be programmed. With only security bit one programmed, the memory can still be read out for program verification. After the second security bit is programmed, it is no longer possible to read out (verify) the program memory.

To program the security bits for the 87C550, repeat the programming sequence using the "Pgm Security Bit" levels specified in Table 4. For the masked ROM 83C550 the security bit information is submitted with the ROM code as shown in Table 3.

### ROM Code Submission

When submitting a ROM code for the 83C550, the following must be specified:

1. The 4k byte user ROM program.
2. The 32 byte ROM encryption key.
3. The ROM security bits.
4. The watchdog timer parameters.

This information can be submitted in an EPROM (2764) or hex file with the format specified in Table 3.

**Table 2. External Pin Status During Idle and Power-Down Modes**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

**Table 3. ROM Code Submittal Requirements**

| ADDRESS | CONTENT | BIT(s) | COMMENT |
|---|---|---|---|
| 0000H to 0FFFH | Data | 7:0 | User ROM data |
| 1000H to 101FH | Key | 7:0 | ROM encryption key; FFH = no encryption |
| 1020H | Security bit | 0 | ROM security bit 1 |
| 1020H | Security bit | 1 | ROM security bit 2<br>0 = enable security feature<br>1 = disable security feature |
| 1030H | WDCON[1] | 7:5 | PRE2:0 |
| 1030H | WDCON[1] | 4 | Not used |
| 1030H | WDCON[1] | 3 | Not used |
| 1030H | WDCON[1] | 2 | WDRUN = 0, not ROM coded |
| 1030H | WDCON[1] | 1 | WDTOF = 0, not ROM coded |
| 1030H | WDCON[1] | 0 | WDMOD |
| 1031H | Not used | | |
| 1032H | WD | 7:0 | Watchdog autoload value<br>(see specification) |

**NOTE:**
1. See Watchdog Timer Specification for definition of WDL and WDCON bits.

**Electrical Deviations from Commercial Specifications for Extended Temperature Range**
DC and AC parameters not included here are the same as in the commercial temperature range table.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = –40°C to +85°C, $V_{CC}$ = 5V ±10% (87C550), $V_{CC}$ = 5V ±20% (80/83C550), $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$ | | –0.5 | $0.2V_{CC}$–0.15 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | | 0 | $0.2V_{CC}$–0.35 | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST | | $0.2V_{CC}$+1 | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage to XTAL1, RST | | $0.7V_{CC}$+0.1 | $V_{CC}$+0.5 | V |
| $I_{IL}$ | Logical 0 input current, ports 2, 3 | $V_{IN}$ = 0.45V | | –75 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 2, 3 | $V_{IN}$ = 2.0V | | –750 | µA |
| $I_{CC}$ | Power supply current:<br>Active mode<br>Idle mode<br>Power down mode | $V_{CC}$ = 4.5–5.5V,<br>Frequency range =<br>3.5 to 16MHz | | <br>35<br>6<br>50 | <br>mA<br>mA<br>µA |

## ADC DC ELECTRICAL CHARACTERISTICS

$AV_{CC}$ = 5V ±10%, $AV_{SS}$ = 0V, $T_{amb}$ = –40°C to 85°C, unless otherwise specified

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $AV_{CC}$ | Analog supply | $AV_{CC} = V_{CC} \pm 0.2$ | 4.5 | 5.5 | V |
| $V_{REF}$ | Analog reference; $AV_{REF+}$ $AV_{REF-}$ | | $AV_{SS}$ – 0.2 | $AV_{CC}$ + 0.2 | V |
| $AI_{CC}$ | Analog operating supply current | See note 1 | | 3.0 | mA |
| $AV_{IN}$ | Analog input voltage | | $AV_{SS}$ – 0.2 | $AV_{CC}$ + 0.2 | V |
| $A_{IC}$, $C_{IA}$ | Analog input capacitance | | | 15 | pF |
| $t_{ADS}$ | Sampling time | | | $8t_{CY}$ | |
| $t_{ADC}$ | Conversion time | | | $40t_{CY}$ | |
| Ae | Absolute voltage error | | | ±1.5 | LSB |
| $E_{RA}$ | Relative accuracy | | | ±1 | LSB |
| OSe | Offset error | See note 1 | | ±1 | LSB |
| Ge | Gain error | See note 1 | | 0.4 | % |
| $M_{CTC}$ | Channel-to-channel matching | | | ±1 | LSB |
| Ct | Crosstalk | 0 – 100kHz | | –60 | dB |
| Rref | Resistance between $AV_{REF+}$ and $AV_{REF-}$ | | 1.0 | 10.0 | KΩ |
| $AI_{ID}$ | Idle mode supply current | See note 4 | | 50 | µA |
| $AI_{PD}$ | Power down supply current | See note 4 | | 50 | µA |

**NOTES:**
1. Conditions: $V_{REF+}$ = 4.99712V, $V_{REF-}$ = 0V. $AI_{CC}$ value does not include the resistor ladder current. For the 40-pin package, where the $V_{REF-}$ inputs are connected to $AV_{CC}$ and $AV_{SS}$, the current $AI_{CC}$ will be increased by the register ladder current and may exceed the maximum shown here.
2. The resistor ladder network is not disconnected in the power-down or idle modes. Thus to conserve power, the user must remove $AV_{CC}$ and $V_{REF+}$.
3. If the A/D function is not required, or if the A/D function is only needed periodically, $AV_{CC}$ can be removed without affecting the operation of the digital circuitry. Contents of ADCON and ADAT are not guaranteed to be valid. Digital inputs P1.0 to P1.7 will not function normally. No digital outputs are present on these pins.
4. For this test, the Analog inputs must be at the supplies (either $V_{DD}$ or $V_{SS}$).

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | –40 to +85 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ pin to $V_{SS}$ (87C550 only) | 0 to +13.0 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Input, output current on any two I/O pins | ±10 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.5 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10% (87C550), $V_{CC}$ = 5V ±20% (80/83C550), $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYPICAL[1] | MAX | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$[7] | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$[7] | | 0 | | $0.2V_{CC}$–0.3 | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST[7] | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST[7] | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 2, 3 | $I_{OL}$ = 1.6mA[2] | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$ | $I_{OL}$ = 3.2mA[2] | | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 2, 3, ALE, $\overline{PSEN}$[3] | $I_{OH}$ = –60µA,<br>$I_{OH}$ = –25µA<br>$I_{OH}$ = –10µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$ | | | V<br>V<br>V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode) | $I_{OH}$ = –800µA,<br>$I_{OH}$ = –300µA<br>$I_{OH}$ = –80µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$ | | | V<br>V<br>V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3[7] | $V_{IN}$ = 0.45V | | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3[7] | See note 4 | | | –650 | µA |
| $I_{LI}$ | Input leakage current, port 0 | $V_{IN}$ = $V_{IL}$ or $V_{IH}$ | | | ±10 | µA |
| $I_{CC}$ | Power supply current (does not include $AI_{CC}$):[7]<br>Active mode @ 16MHz[5]<br>Idle mode @ 16MHz<br>Power down mode | See note 6 | | <br>11.5<br>1.3<br>3 | <br>25<br>5<br>50 | <br>mA<br>mA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | 50 | | 300 | kΩ |
| $C_{IO}$ | Pin capacitance (I/O pins only) | | | | 10 | pF |

**NOTES:**
1. Typical ratings are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
4. Pins of ports 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
5. $I_{CC}$MAX at other frequencies is given by: Active mode; $I_{CC}$MAX = 1.43 × FREQ + 1.90: Idle mode; $I_{CC}$MAX = 0.14 × FREQ +2.31, where FREQ is the external oscillator frequency in MHz. $I_{CC}$MAX is given in mA. See Figure 12.
6. See Figures 13 through 16 for $I_{CC}$ test conditions.
7. These values apply only to $T_{amb}$ = 0°C to +70°C. For $T_{amb}$ = –40°C to +85°C. See table on previous page.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10% (87C550), $V_{CC}$ = 5V ±20% (80/83C550), $V_{SS}$ = 0V[1, 2]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 5 | Oscillator frequency: **Speed Versions** S8XC550 Exx | | | 3.5 | 16 | MHz |
| $t_{LHLL}$ | 5 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 5 | Address valid to ALE low | 7 | | $t_{CLCL}-55$ | | ns |
| $t_{LLAX}$ | 5 | Address hold after ALE low | 27 | | $t_{CLCL}-35$ | | ns |
| $t_{LLIV}$ | 5 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 5 | ALE low to $\overline{PSEN}$ low | 22 | | $t_{CLCL}-40$ | | ns |
| $t_{PLPH}$ | 5 | $\overline{PSEN}$ pulse width | 142 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 5 | $\overline{PSEN}$ low to valid instruction in | | 82 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 5 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 5 | Input instruction float after $\overline{PSEN}$ | | 37 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 5 | Address to valid instruction in | | 207 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 5 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 6, 7 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 6, 7 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 6, 7 | $\overline{RD}$ low to valid data in | | 212 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 6, 7 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 6, 7 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-70$ | ns |
| $t_{LLDV}$ | 6, 7 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 6, 7 | Address to valid data in | | 397 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 6, 7 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 137 | 247 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 6, 7 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 120 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 6, 7 | Data valid to $\overline{WR}$ transition | 12 | | $t_{CLCL}-50$ | | ns |
| $t_{WHQX}$ | 6, 7 | Data hold after $\overline{WR}$ | 12 | | $t_{CLCL}-50$ | | ns |
| $t_{RLAZ}$ | 6, 7 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 6, 7 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 22 | 102 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 9 | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | 9 | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | 9 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 9 | Fall time | | 20 | | 20 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 8 | Serial port clock cycle time | 750 | | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 8 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 8 | Output data hold after clock rising edge | 8 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 8 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 8 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}-133$ | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{PSEN}$ low.

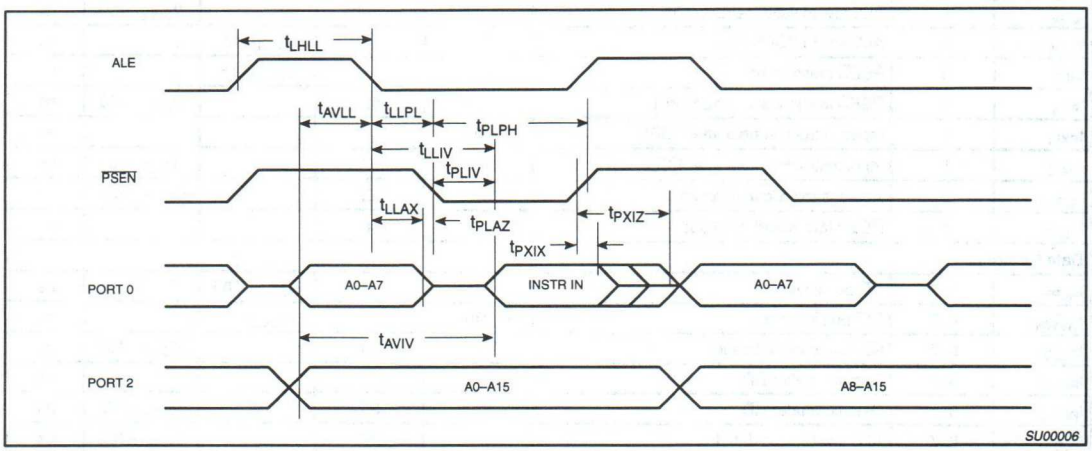



**Figure 5. External Program Memory Read Cycle**

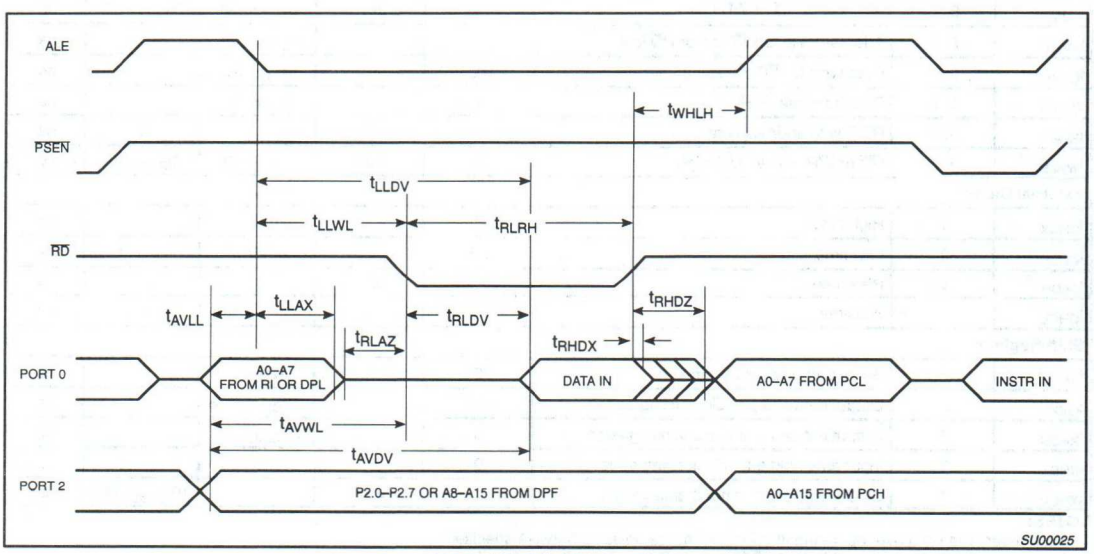



**Figure 6. External Data Memory Read Cycle**

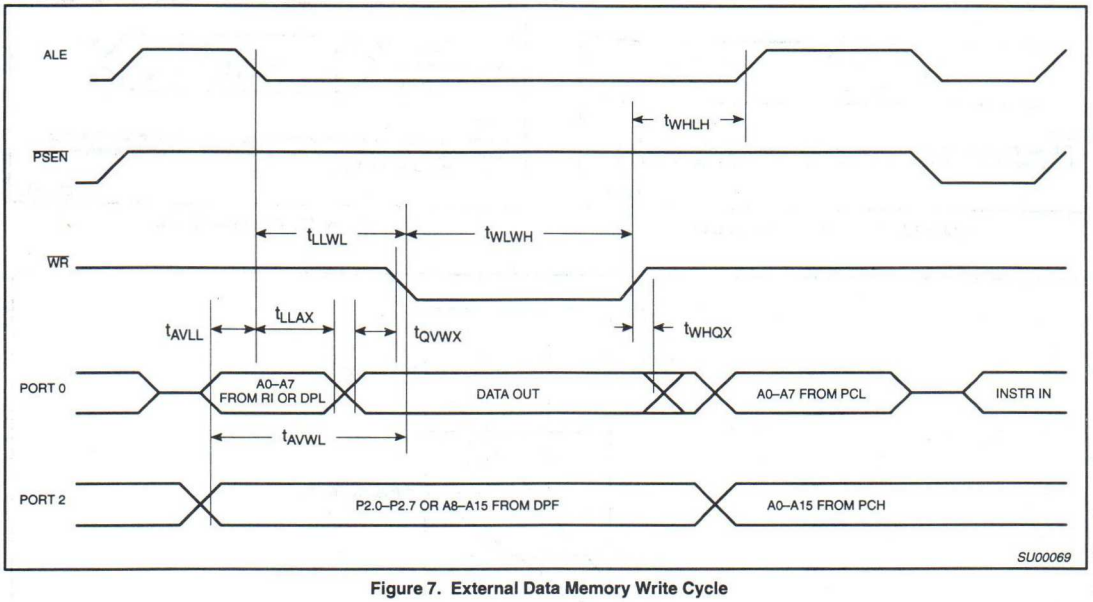



**Figure 7. External Data Memory Write Cycle**

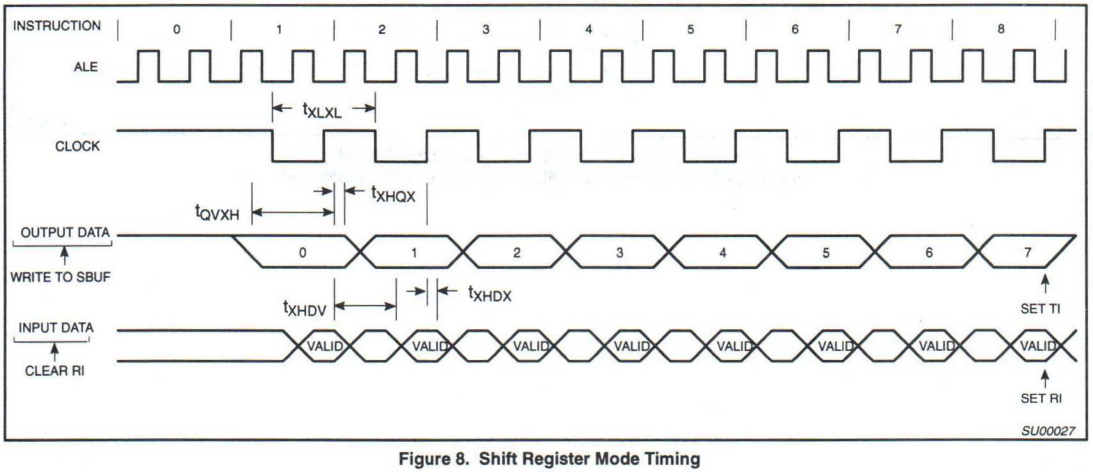



**Figure 8. Shift Register Mode Timing**

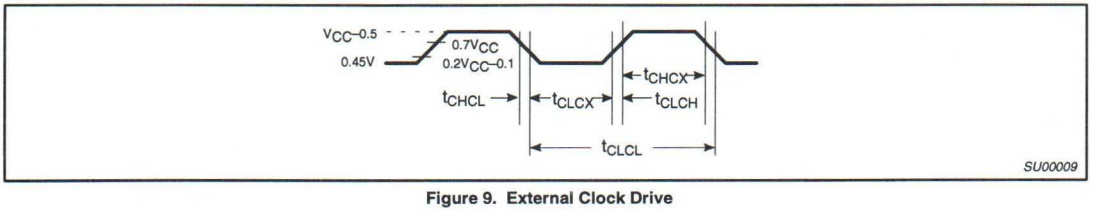



**Figure 9. External Clock Drive**

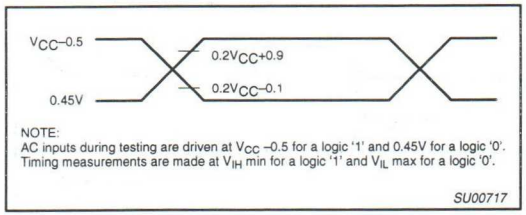



**Figure 10. AC Testing Input/Output**

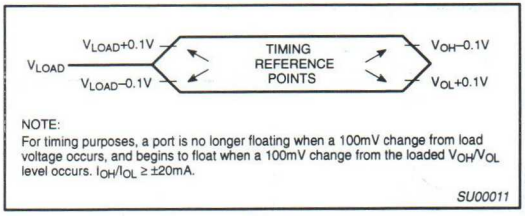



**Figure 11. Float Waveform**

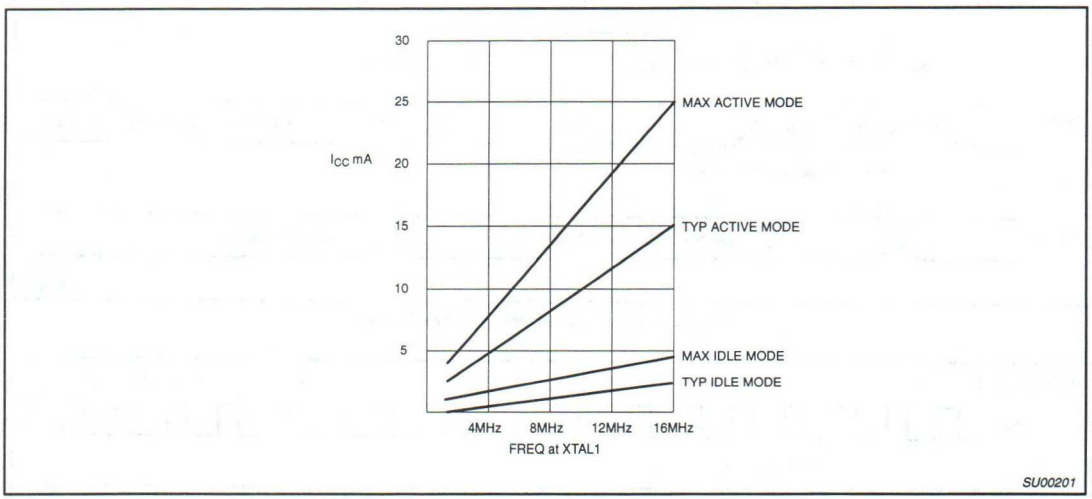



**Figure 12. $I_{CC}$ vs. FREQ (Commercial Temp. Range)**
**Valid only within frequency specifications of the device under test**

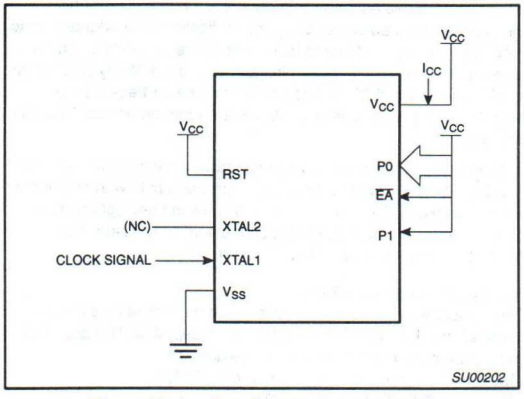



**Figure 13. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

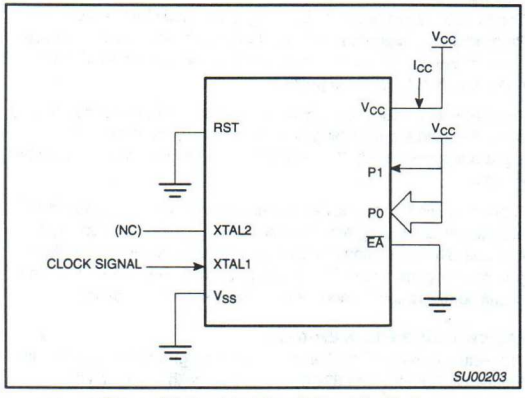



**Figure 14. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

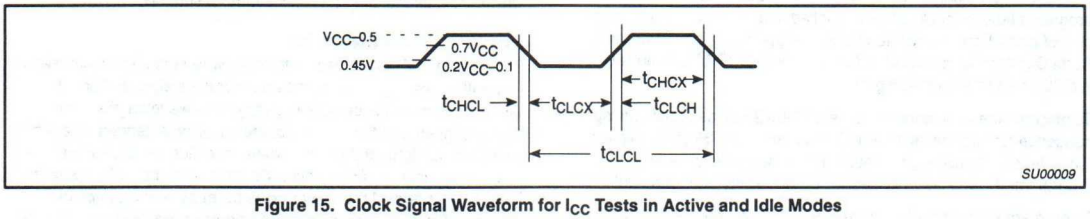



**Figure 15. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

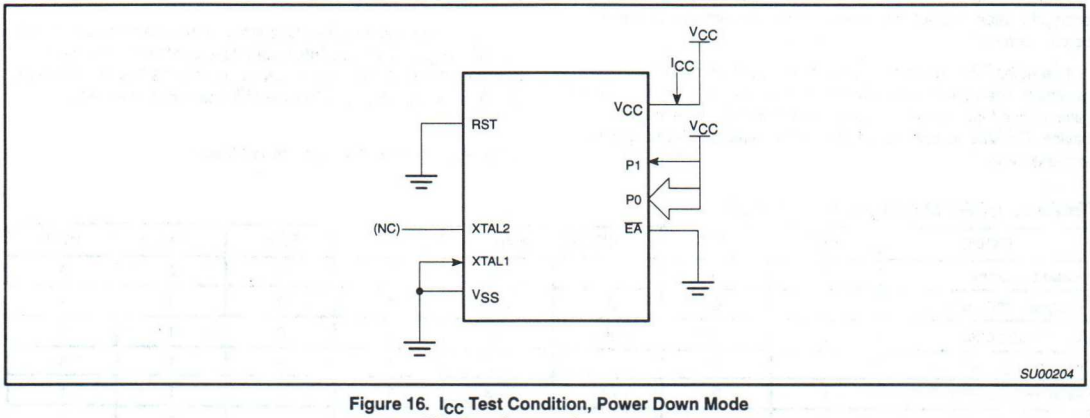



**Figure 16. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected.**
**$V_{CC}$ = 2V to 5.5V.**

## EPROM CHARACTERISTICS

The 87C550 is programmed by using a modified Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C550 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an S87C550 manufactured by Philips.

Table 4 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the lock bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 17 and 18. Figure 19 shows the circuit configuration for normal program memory verification.

### Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 17. Note that the 87C550 is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 2 and 3, as shown in Figure 17. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 1 and 2 specified in Table 4 are held at the 'Program Code Data' levels indicated in Table 4. The ALE/$\overline{PROG}$ is pulsed low 25 times as shown in Figure 18.

To program the encryption table, repeat the 25 pulse programming sequence for addresses 0 through 1FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the security bits, repeat the 25 pulse programming sequence using the 'Pgm Security Bit' levels. After one security bit is programmed, further programming of the code memory and encryption table is disabled. However, the other security bit can still be programmed.

Note that the $\overline{EA}$/$V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

#### Program Verification

If security bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 2 and 3 as shown in Figure 19. The other pins are held at the 'Verify Code Data' levels indicated in Table 4. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

#### Reading the Signature Bytes

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P1.0 and P1.1 need to be pulled to a logic low. The values are:
(030H) = 15H indicates manufactured by Philips
(031H) = 96H indicates S87C550

### Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 4, and which satisfies the timing specifications, is suitable.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-sec/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000uW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

**Table 4. EPROM Programming Modes**

| MODE | RST | $\overline{PSEN}$ | ALE/$\overline{PROG}$ | $\overline{EA}$/$V_{PP}$ | P2.7 | P2.6 | P1.1 | P1.0 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm security bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm security bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V±10% during programming and verification.

* ALE/$\overline{PROG}$ receives 25 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

™Trademark phrase of Intel Corporation.

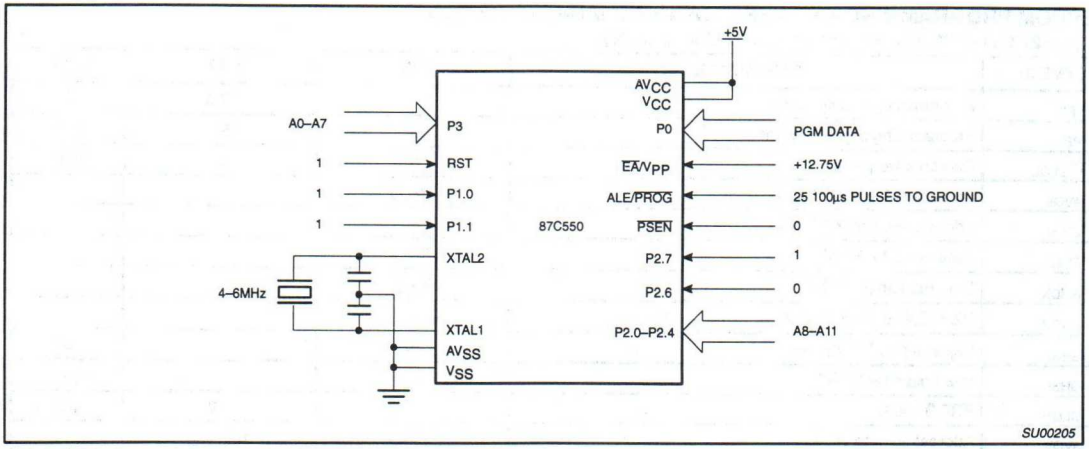



**Figure 17. Programming Configuration**

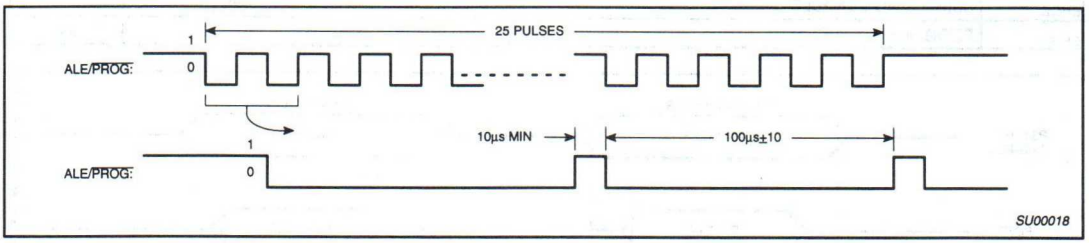



**Figure 18. PROG Waveform**

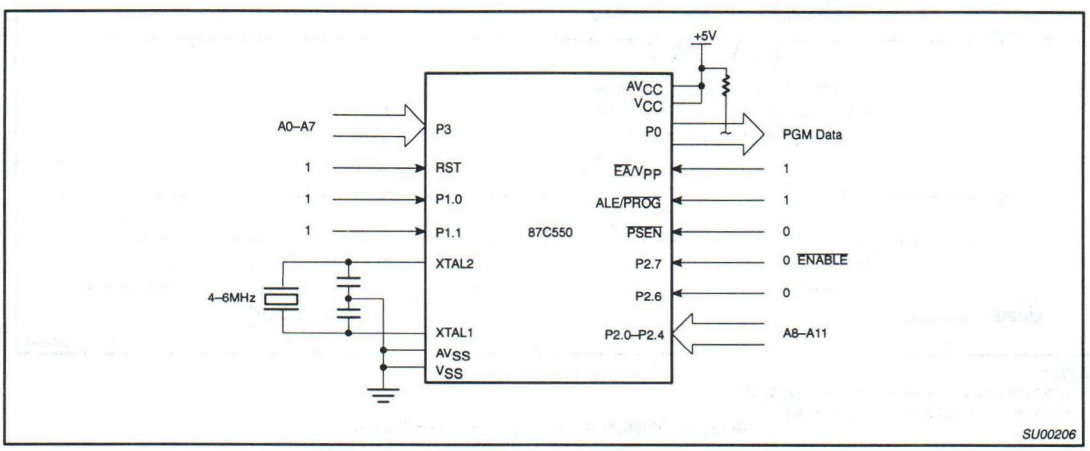
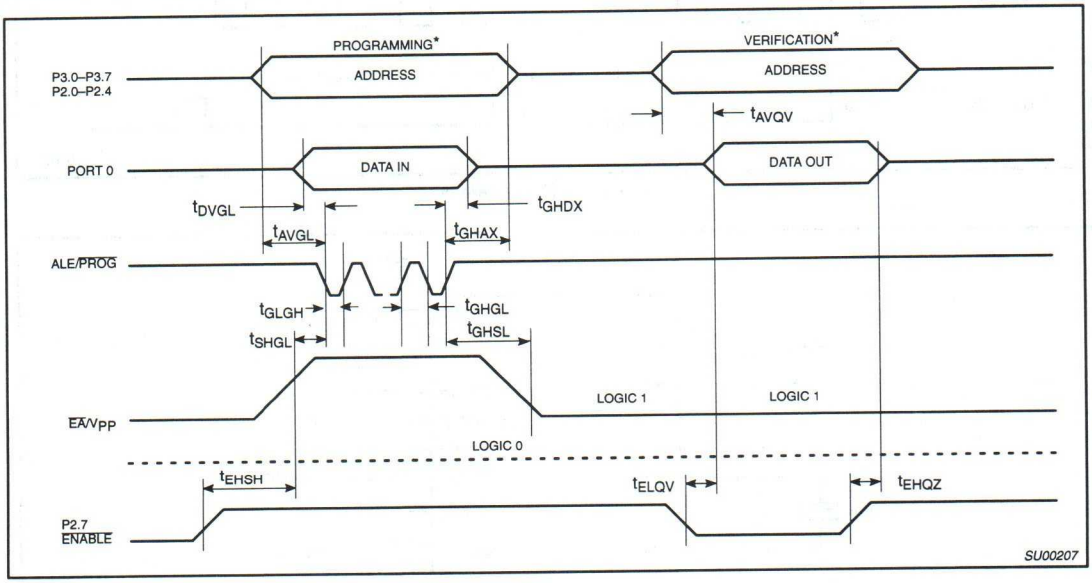



**Figure 19. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (See Figure 20)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




**NOTE:**

* FOR PROGRAMMING VERIFICATION, SEE FIGURE 17.
FOR VERIFICATION CONDITIONS, SEE FIGURE 19.

**Figure 20. EPROM Programming and Verification**

## 8XC552 OVERVIEW

The 8XC552 is a stand-alone high-performance microcontroller designed for use in real-time applications such as instrumentation, industrial control, and automotive control applications such as engine management and transmission control. The device provides, in addition to the 80C51 standard functions, a number of dedicated hardware functions for these applications.

The 8XC552 single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 8XC552 uses the powerful instruction set of the 80C51. Additional special function registers are incorporated to control the on-chip peripherals. Three versions of the derivative exist although the generic term "8XC552" is used to refer to family members:

83C552: 8k bytes mask-programmable ROM, 256 bytes RAM

87C552: 8k bytes EPROM, 256 bytes RAM

80C552: ROMless version of the 83C552

The 8XC552 contains a nonvolatile 8k × 8 read-only program memory, a volatile 256 × 8 read/write data memory, five 8-bit I/O ports and one 8-bit input port, two 16-bit timer/event counters (identical to the timers of the 80C51), an additional 16-bit timer coupled to capture and compare latches, a fifteen-source, two-priority-level, nested interrupt structure, an 8-input ADC, a dual DAC pulse width modulated interface, two serial interfaces (UART and $I^2C$ bus), a "watchdog" timer, and on-chip oscillator and timing circuits. For systems that require extra capability, the 8XC552 can be expanded using standard TTL compatible memories and logic

The 8XC552 has two software selectable modes of reduced activity for further power reduction—Idle and Power-down. The idle mode freezes the CPU and resets Timer T2 and the ADC and PWM circuitry but allows the other timers, RAM, serial ports, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to become inoperative.

## 83C562 OVERVIEW

The 83C562 has been derived from the 8XC552 with the following changes:

- The SIO1 ($I^2C$) interface has been omitted.
- The output of port lines P1.6 and P1.7 have a standard configuration instead of open drain.
- The resolution of the A/D converter is decreased from 10 bits to 8 bits.
- The time of an A/D conversion has decreased from 50 machine cycles to 24 machine cycles.

All other functions, pinning and packaging are unchanged.

This chapter of the users' guide can be used for the 83C562 by omitting or changing the following:

- Disregard the description of SIO1 ($I^2C$).
- The SFRs for the interface: S1ADR, S1DAT, S1STA, and S1CON are not implemented. The two SIO1 related flags ES1 in SFR IEN0 and PS1 in SFR IP0 are also not implemented. These two flag locations are undefined after RESET. The interrupt vector for SIO1 is not used.
- Port lines P1.6 and P1.7 are not open drain but have the same standard configuration and electrical characteristics as P1.0-P1.5. Port lines P1.6 and P1.7 have alternative functions.
- The A/D converter has a resolution of 8 bits instead of 10 bits and consequently the two high-order bits 6 and 7 of SFR ADCON are not implemented. These two locations are undefined after RESET. The 8-bit result of an A/D conversion is present in SFR ADCH. The result can always be calculated from the formula:

$$256 \times \frac{V_{IN} - AV_{ref-}}{AV_{ref+} - AV_{ref-}}$$

  The A/D conversion time is 24 machine cycles instead of 50 machine cycles, and the sampling time is 6 machine cycles instead of 8 machine cycles. The conversion time takes 3 machine cycles per bit.
- The serial I/O function SIO0 and its SFRs S0BUF and S0CON are renamed to SIO, SBUF, and SCON. The interrupt related flags ES0 and PS0 are renamed ES and PS. Interrupt source S0 is renamed S. The serial I/O function remains the same.

## Differences From the 80C51

### Program Memory

The 8XC552 contains 8k bytes of on-chip program memory which can be extended to 64k bytes with external memories (see Figure 1). When the $\overline{EA}$ pin is held high, the 8XC552 fetches instructions from internal ROM unless the address exceeds 1FFFH. Locations 2000H to FFFFH are fetched from external program memory. When the $\overline{EA}$ pin is held low, all instruction fetches are from external memory. ROM locations 0003H to 0073H are used by interrupt service routines.

### Data Memory

The internal data memory is divided into 3 sections: the lower 128 bytes of RAM, the upper 128 bytes of RAM, and the 128-byte special function register areas. The lower 128 bytes of RAM are directly and indirectly addressable. While RAM locations 128 to 255 and the special function register area share the same address space, they are accessed through different addressing modes. RAM locations 128 to 255 are only indirectly addressable, and the special function registers are only directly addressable. All other aspects of the internal RAM are identical to the 8051.

The stack may be located anywhere in the internal RAM by loading the 8-bit stack pointer. Stack depth is 256 bytes maximum.

### Special Function Registers

The special function registers (directly addressable only) contain all of the 8XC552 registers except the program counter and the four register banks. Most of the 56 special function registers are used to control the on-chip peripheral hardware. Other registers include arithmetic registers (ACC, B, PSW), stack pointer (SP), and data pointer registers (DHP, DPL). Sixteen of the SFRs contain 128 directly addressable bit locations. Table 1 lists the 8XC552's special function registers.

The standard 80C51 SFRs are present and function identically in the 8XC552 except where noted in the following sections.

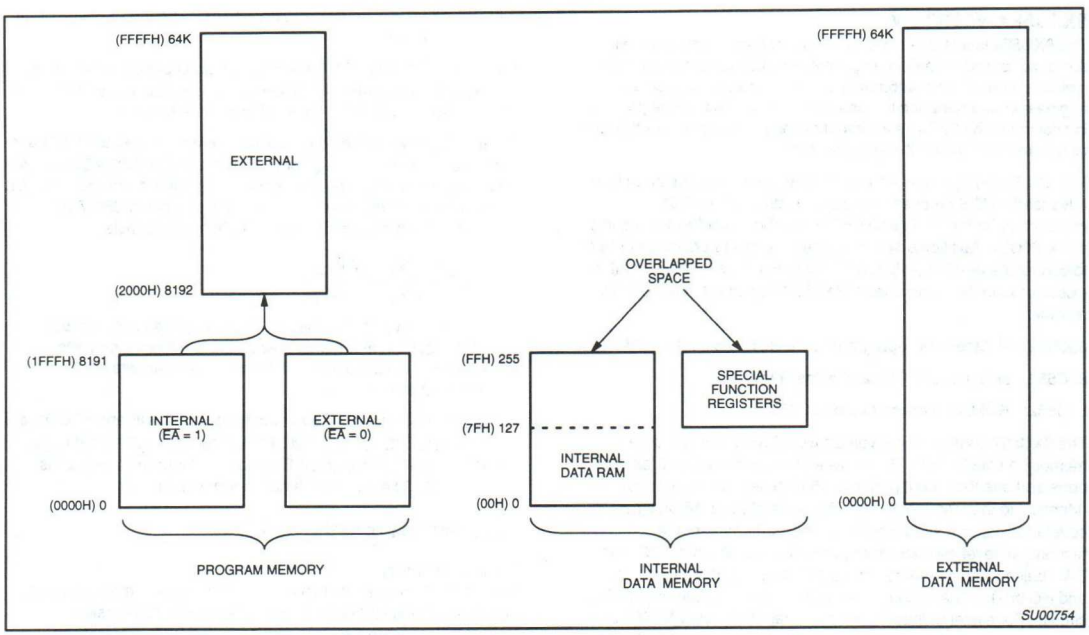



**Figure 1. Memory Map**

**Timer T2**

Timer T2 is a 16-bit timer consisting of two registers TMH2 (HIGH byte) and TML2 (LOW byte). The 16-bit timer/counter can be switched off or clocked via a prescaler from one of two sources: $f_{OSC}/12$ or an external signal. When Timer T2 is configured as a counter, the prescaler is clocked by an external signal on T2 (P1.4). A rising edge on T2 increments the prescaler, and the maximum repetition rate is one count per machine cycle (1MHz with a 12MHz oscillator).

The maximum repetition rate for Timer T2 is twice the maximum repetition rate for Timer 0 and Timer 1. T2 (P1.4) is sampled at S2P1 and again at S5P1 (i.e., twice per machine cycle). A rising edge is detected when T2 is LOW during one sample and HIGH during the next sample. To ensure that a rising edge is detected, the input signal must be LOW for at least 1/2 cycle and then HIGH for at least 1/2 cycle. If a rising edge is detected before the end of S2P1, the timer will be incremented during the following cycle; otherwise it will be incremented one cycle later. The prescaler has a programmable division factor of 1, 2, 4, or 8 and is cleared if its division factor or input source is changed, or if the timer/counter is reset.

Timer T2 may be read "on the fly" but possesses no extra read latches, and software precautions may have to be taken to avoid misinterpretation in the event of an overflow from least to most significant byte while Timer T2 is being read. Timer T2 is not loadable and is reset by the RST signal or by a rising edge on the input signal RT2, if enabled. RT2 is enabled by setting bit T2ER (TM2CON.5).

When the least significant byte of the timer overflows or when a 16-bit overflow occurs, an interrupt request may be generated. Either or both of these overflows can be programmed to request an interrupt. In both cases, the interrupt vector will be the same. When the lower byte (TML2) overflows, flag T2B0 (TM2CON) is set and flag T20V (TM2IR) is set when TMH2 overflows. These flags are set one cycle after an overflow occurs. Note that when T20V is set, T2B0 will also be set. To enable the byte overflow interrupt, bits ET2 (IEN1.7, enable overflow interrupt, see Figure 2) and T2IS0 (TM2CON.6, byte overflow interrupt select) must be set. Bit TWB0 (TM2CON.4) is the Timer T2 byte overflow flag.

To enable the 16-bit overflow interrupt, bits ET2 (IE1.7, enable overflow interrupt) and T2IS1 (TM2CON.7, 16-bit overflow interrupt select) must be set. Bit T2OV (TM2IR.7) is the Timer T2 16-bit overflow flag. All interrupt flags must be reset by software. To enable both byte and 16-bit overflow, T2IS0 and T2IS1 must be set and two interrupt service routines are required. A test on the overflow flags indicates which routine must be executed. For each routine, only the corresponding overflow flag must be cleared.

Timer T2 may be reset by a rising edge on RT2 (P1.5) if the Timer T2 external reset enable bit (T2ER) in T2CON is set. This reset also clears the prescaler. In the idle mode, the timer/counter and prescaler are reset and halted. Timer T2 is controlled by the TM2CON special function register (see Figure 3).

**Table 1. 8XC552 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| ADCH# | A/D converter high | C6H | | | | | | | | | xxxxxxxxB |
| ADCON# | Adc control | C5H | ADC.1 | ADC.0 | ADEX | ADCI | ADCS | AADR2 | AADR1 | AADR0 | xx000000B |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| CTCON# | Capture control | EBH | CTN3 | CTP3 | CTN2 | CTP2 | CTN1 | CTP1 | CTN0 | CTP0 | 00H |
| CTH3# | Capture high 3 | CFH | | | | | | | | | xxxxxxxxB |
| CTH2# | Capture high 2 | CEH | | | | | | | | | xxxxxxxxB |
| CTH1# | Capture high 1 | CDH | | | | | | | | | xxxxxxxxB |
| CTH0# | Capture high 0 | CCH | | | | | | | | | xxxxxxxxB |
| CMH2# | Compare high 2 | CBH | | | | | | | | | 00H |
| CMH1# | Compare high 1 | CAH | | | | | | | | | 00H |
| CMH0# | Compare high 0 | C9H | | | | | | | | | 00H |
| CTL3# | Capture low 3 | AFH | | | | | | | | | xxxxxxxxB |
| CTL2# | Capture low 2 | AEH | | | | | | | | | xxxxxxxxB |
| CTL1# | Capture low 1 | ADH | | | | | | | | | xxxxxxxxB |
| CTL0# | Capture low 0 | ACH | | | | | | | | | xxxxxxxxB |
| CML2# | Compare low 2 | ABH | | | | | | | | | 00H |
| CML1# | Compare low 1 | AAH | | | | | | | | | 00H |
| CML0# | Compare low 0 | A9H | | | | | | | | | 00H |
| DPTR: | Data pointer (2 bytes) | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IEN0*# | Interrupt enable 0 | A8H | EA | EAD | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | EF | EE | ED | EC | EB | EA | E9 | E8 | |
| IEN1*# | Interrupt enable 1 | E8H | ET2 | ECM2 | ECM1 | ECM0 | ECT3 | ECT2 | ECT1 | ECT0 | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP0*# | Interrupt priority 0 | B8H | – | PAD | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | FF | FE | FD | FC | FB | FA | F9 | F8 | |
| IP1*# | Interrupt priority 1 | F8H | PT2 | PCM2 | PCM1 | PCM0 | PCT3 | PCT2 | PCT1 | PCT0 | 00H |
| P5# | Port 5 | C4H | ADC7 | ADC6 | ADC5 | ADC4 | ADC3 | ADC2 | ADC1 | ADC0 | xxxxxxxxB |
| | | | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 | |
| P4# | Port 4 | C0H | CMT1 | CMT0 | CMSR5 | CMSR4 | CMSR3 | CMSR2 | CMSR1 | CMSR0 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TXD | RXD | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | SDA | SCL | RT2 | T2 | CT3I | CT2I | CT1I | CT0I | FFH |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| PCON# | Power control | 87H | SMOD | – | – | WLE | GF1 | GF0 | PD | IDL | 00xx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | F1 | P | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

**Table 1. 8XC552 Special Function Registers** (Continued)

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PWMP# | PWM prescaler | FEH | | | | | | | | | 00H |
| PWM1# | PWM register 1 | FDH | | | | | | | | | 00H |
| PWM0# | PWM register 0 | FCH | | | | | | | | | 00H |
| RTE# | Reset/toggle enable | EFH | TP47 | TP46 | RP45 | RP44 | RP43 | RP42 | RP41 | RP40 | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| S0BUF | Serial 0 data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| S0CON* | Serial 0 control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| S1ADR# | Serial 1 address | DBH | SLAVE ADDRESS | | | | | | | GC | 00H |
| SIDAT# | Serial 1 data | DAH | | | | | | | | | 00H |
| S1STA# | Serial 1 status | D9H | SC4 | SC3 | SC2 | SC1 | SC0 | 0 | 0 | 0 | F8H |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| SICON#* | Serial 1 control | D8H | CR2 | ENS1 | STA | ST0 | SI | AA | CR1 | CR0 | 00H |
| STE# | Set enable | EEH | TG47 | TG46 | SP45 | SP44 | SP43 | SP42 | SP41 | SP40 | C0H |
| TH1 | Timer high 1 | 8DH | | | | | | | | | 00H |
| TH0 | Timer high 0 | 8CH | | | | | | | | | 00H |
| TL1 | Timer low 1 | 8BH | | | | | | | | | 00H |
| TL0 | Timer low 0 | 8AH | | | | | | | | | 00H |
| TMH2# | Timer high 2 | EDH | | | | | | | | | 00H |
| TML2# | Timer low 2 | ECH | | | | | | | | | 00H |
| TMOD | Timer mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| TM2CON# | Timer 2 control | EAH | T2IS1 | T2IS0 | T2ER | T2B0 | T2P1 | T2P0 | T2MS1 | T2MS0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| TM2IR#* | Timer 2 int flag reg | C8H | T20V | CMI2 | CMI1 | CMI0 | CTI3 | CTI2 | CTI1 | CTI0 | 00H |
| T3# | Timer 3 | FFH | | | | | | | | | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

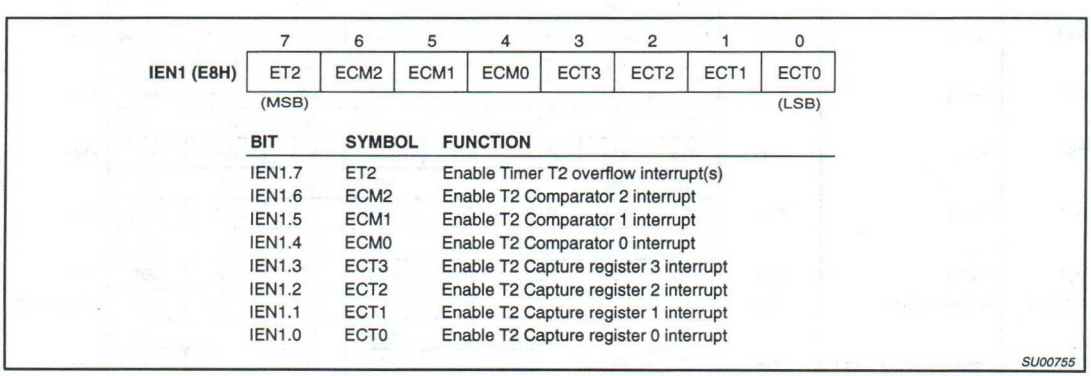



**Figure 2. Timer T2 Interrupt Enable Register (IEN1)**

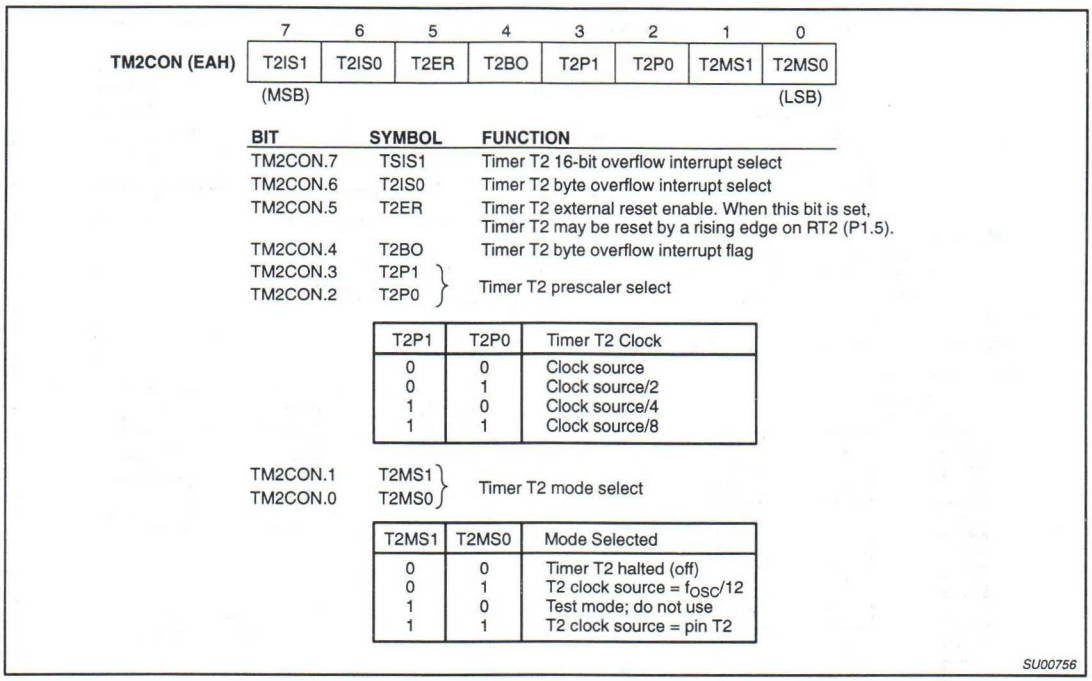



**Figure 3. T2 Control Register (TM2CON)**

**Timer T2 Extension:** When a 12MHz oscillator is used, a 16-bit overflow on Timer T2 occurs every 65.5, 131, 262, or 524 ms, depending on the prescaler division ratio; i.e., the maximum cycle time is approximately 0.5 seconds. In applications where cycle times are greater than 0.5 seconds, it is necessary to extend Timer T2. This is achieved by selecting fosc/12 as the clock source (set T2MS0, reset T2MS1), setting the prescaler division ration to 1/8 (set T2P0, set T2P1), disabling the byte overflow interrupt (reset T2IS0) and enabling the 16-bit overflow interrupt (set T2IS1). The following software routine is written for a three-byte extension which gives a maximum cycle time of approximately 2400 hours.

```
OVINT: PUSH   ACC      ;save accumulator
       PUSH   PSW      ;save status
       INC    TIMEX1   ;increment first byte (low order)
                       ;of extended timer
       MOV    A,TIMEX1
       JNZ    INTEX    ;jump to INTEX if ;there is no overflow

       INC    TIMEX2   ;increment second byte
       MOV    A,TIMEX2
       JNZ    INTEX    ;jump to INTEX if there is no overflow
       INC    TIMEX3   ;increment third byte (high order)

INTEX: CLR    T2OV     ;reset interrupt flag
       POP    PSW      ;restore status
       POP    ACC      ;restore accumulator
       RETI            ;return from interrupt
```

**Timer T2, Capture and Compare Logic:** Timer T2 is connected to four 16-bit capture registers and three 16-bit compare registers. A capture register may be used to capture the contents of Timer T2 when a transition occurs on its corresponding input pin. A compare register may be used to set, reset, or toggle port 4 output pins at certain pre-programmable time intervals.

The combination of Timer T2 and the capture and compare logic is very powerful in applications involving rotating machinery, automotive injection systems, etc. Timer T2 and the capture and compare logic are shown in Figure 4.

**Capture Logic:** The four 16-bit capture registers that Timer T2 is connected to are: CT0, CT1, CT2, and CT3. These registers are loaded with the contents of Timer T2, and an interrupt is requested upon receipt of the input signals CT0I, CT1I, CT2I, or CT3I. These input signals are shared with port 1. The four interrupt flags are in the Timer T2 interrupt register (TM2IR special function register). If the capture facility is not required, these inputs can be regarded as additional external interrupt inputs.

Using the capture control register CTCON (see Figure 5), these inputs may capture on a rising edge, a falling edge, or on either a rising or falling edge. The inputs are sampled during S1P1 of each cycle. When a selected edge is detected, the contents of Timer T2 are captured at the end of the cycle.

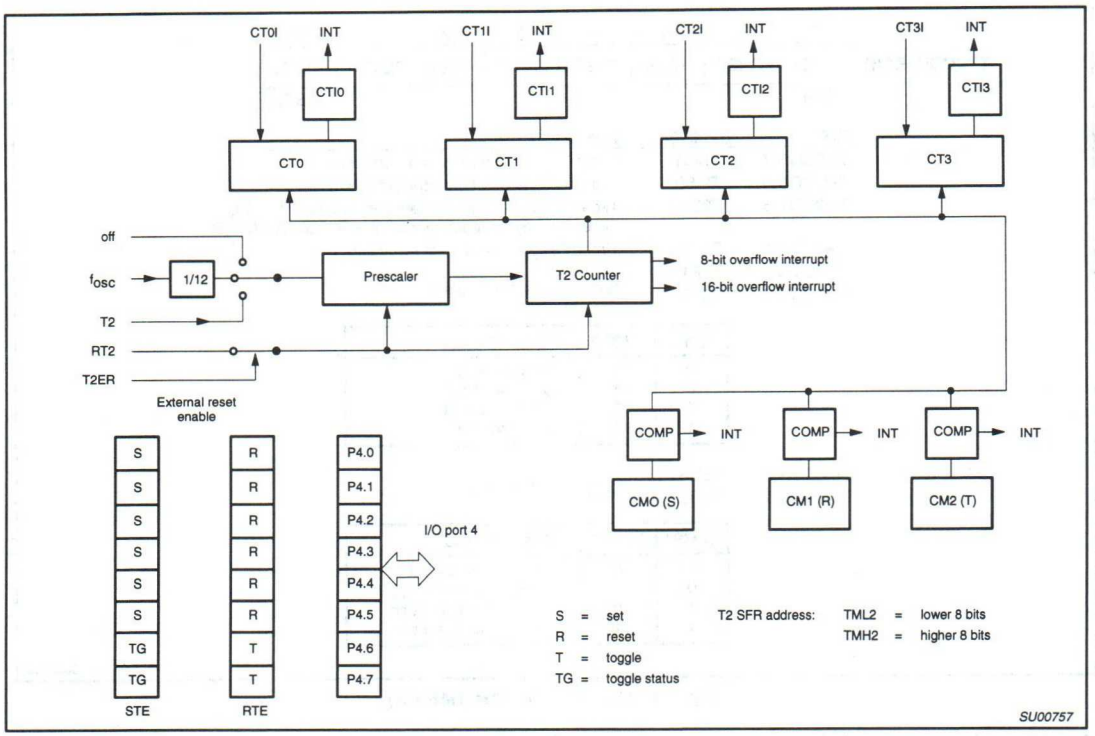



**Figure 4. Block Diagram of Timer 2**

**Measuring Time Intervals Using Capture Registers:** When a recurring external event is represented in the form of rising or falling edges on one of the four capture pins, the time between two events can be measured using Timer T2 and a capture register. When an event occurs, the contents of Timer T2 are copied into the relevant capture register and an interrupt request is generated. The interrupt service routine may then compute the interval time if it knows the previous contents of Timer T2 when the last event occurred. With a 12MHz oscillator, Timer T2 can be programmed to overflow every 524ms. When event interval times are shorter than this, computing the interval time is simple, and the interrupt service routine is short. For longer interval times, the Timer T2 extension routine may be used.

**Compare Logic:** Each time Timer T2 is incremented, the contents of the three 16-bit compare registers CM0, CM1, and CM2 are compared with the new counter value of Timer T2. When a match is found, the corresponding interrupt flag in TM2IR is set at the end of the following cycle. When a match with CM0 occurs, the controller sets bits 0-5 of port 4 if the corresponding bits of the set enable register STE are at logic 1.

When a match with CM1 occurs, the controller resets bits 0-5 of port 4 if the corresponding bits of the reset/toggle enable register RTE are at logic 1 (see Figure 6 for RTE register function). If RTE is "0", then P4.n is not affected by a match between CM1 or CM2 and Timer 2. When a match with CM2 occurs, the controller "toggles" bits 6 and 7 of port 4 if the corresponding bits of the RTE are at logic 1. The port latches of bits 6 and 7 are not toggled.

Two additional flip-flops store the last operation, and it is these flip-flops that are toggled.

Thus, if the current operation is "set," the next operation will be "reset" even if the port latch is reset by software before the "reset" operation occurs. The first "toggle" after a chip RESET will set the port latch. The contents of these two flip-flops can be read at STE.6 and STE.7 (corresponding to P4.6 and P4.7, respectively). Bits STE.6 and STE.7 are read only (see Figure 7 for STE register function). A logic 1 indicates that the next toggle will set the port latch; a logic 0 indicates that the next toggle will reset the port latch. CM0, CM1, and CM2 are reset by the RST signal.

The modified port latch information appears at the port pin during S5P1 of the cycle following the cycle in which a match occurred. If the port is modified by software, the outputs change during S1P1 of the following cycle. Each port 4 bit can be set or reset by software at any time. A hardware modification resulting from a comparator match takes precedence over a software modification in the same cycle. When the comparator results require a "set" and a "reset" at the same time, the port latch will be reset.

**Timer T2 Interrupt Flag Register TM2IR:** Eight of the nine Timer T2 interrupt flags are located in special function register TM2IR (see Figure 8). The ninth flag is TM2CON.4.

The CT0I and CT1I flags are set during S4 of the cycle in which the contents of Timer T2 are captured. CT0I is scanned by the interrupt logic during S2, and CT1I is scanned during S3. CT2I and CT3I are set during S6 and are scanned during S4 and S5. The associated

interrupt requests are recognized during the following cycle. If these flags are polled, a transition at CT0I or CT1I will be recognized one cycle before a transition on CT2I or CT3I since registers are read during S5. The CMI0, CMI1, and CMI2 flags are set during S6 of the cycle following a match. CMI0 is scanned by the interrupt logic during S2; CMI1 and CMI2 are scanned during S3 and S4. A match will be recognized by the interrupt logic (or by polling the flags) two cycles after the match takes place.

The 16-bit overflow flag (T2OV) and the byte overflow flag (T2BO) are set during S6 of the cycle in which the overflow occurs. These flags are recognized by the interrupt logic during the next cycle.

Special function register IP1 (Figure 8) is used to determine the Timer T2 interrupt priority. Setting a bit high gives that function a high priority, and setting a bit low gives the function a low priority. The functions controlled by the various bits of the IP1 register are shown in Figure 8.

**CTCON (EBH)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| CTN3 | CTP3 | CTN2 | CTP2 | CTN1 | CTP1 | CTN1 | CTP0 |
| (MSB) | | | | | | | (LSB) |

| BIT | SYMBOL | CAPTURE/INTERRUPT ON: |
|---|---|---|
| CTCON.7 | CTN3 | Capture Register 3 triggered by a falling edge on CT3I |
| CTCON.6 | CTP3 | Capture Register 3 triggered by a rising edge on CT3I |
| CTCON.5 | CTN2 | Capture Register 2 triggered by a falling edge on CT2I |
| CTCON.4 | CTP2 | Capture Register 2 triggered by a rising edge on CT2I |
| CTCON.3 | CTN1 | Capture Register 1 triggered by a falling edge on CT1I |
| CTCON.2 | CTP1 | Capture Register 1 triggered by a rising edge on CT1I |
| CTCON.1 | CTN0 | Capture Register 0 triggered by a falling edge on CT0I |
| CTCON.0 | CTP0 | Capture Register 0 triggered by a rising edge on CT0I |

SU00758

**Figure 5. Capture Control Register (CTCON)**

**RTE (EFH)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| TP47 | TP46 | RP45 | RP44 | RP43 | RP42 | RO41 | RP40 |
| (MSB) | | | | | | | (LSB) |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| RTE.7 | TP47 | If "1" then P4.7 toggles on a match between CM2 and Timer T2 |
| RTE.6 | TP46 | If "1" then P4.6 toggles on a match between CM2 and Timer T2 |
| RTE.5 | RP45 | If "1" then P4.5 is reset on a match between CM2 and Timer T2 |
| RTE.4 | RP44 | If "1" then P4.4 is reset on a match between CM2 and Timer T2 |
| RTE.3 | RP43 | If "1" then P4.3 is reset on a match between CM2 and Timer T2 |
| RTE.2 | RP42 | If "1" then P4.2 is reset on a match between CM2 and Timer T2 |
| RTE.1 | RP41 | If "1" then P4.1 is reset on a match between CM2 and Timer T2 |
| RTE.0 | RP40 | If "1" then P4.0 is reset on a match between CM2 and Timer T2 |

SU00759

**Figure 6. Reset/Toggle Enable Register (RTE)**

**STE (EEH)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| TG47 | TG46 | SP45 | SP44 | SP43 | SP42 | SP41 | SP40 |
| (MSB) | | | | | | | (LSB) |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| STE.7 | TG47 | Toggle flip-flops |
| STE.6 | TG46 | Toggle flip-flops |
| STE.5 | SP45 | If "1" then P4.5 is set on a match between CM0 and Timer T2 |
| STE.4 | SP44 | If "1" then P4.4 is set on a match between CM2 and Timer T2 |
| STE.3 | SP43 | If "1" then P4.3 is set on a match between CM2 and Timer T2 |
| STE.2 | SP42 | If "1" then P4.2 is set on a match between CM2 and Timer T2 |
| STE.1 | SP41 | If "1" then P4.1 is set on a match between CM2 and Timer T2 |
| STE.0 | SP40 | If "1" then P4.0 is set on a match between CM2 and Timer T2 |

SU00760

**Figure 7. Set Enable Register (STE)**

**TM2IR (C8H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| T2OV | CMI2 | CMI1 | CMI0 | CTI3 | CTI2 | CTI1 | CTI0 |
| (MSB) | | | | | | | (LSB) |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| TM2IR.7 | T2OV | Timer T2 16-bit overflow interrupt flag |
| TM2IR.6 | CMI2 | CM2 interrupt flag |
| TM2IR.5 | CMI1 | CM1 interrupt flag |
| TM2IR.4 | CMI0 | CM0 interrupt flag |
| TM2IR.3 | CTI3 | CT3 interrupt flag |
| TM2IR.2 | CTI2 | CT2 interrupt flag |
| TM2IR.1 | CTI1 | CT1 interrupt flag |
| TM2IR.0 | CTI0 | CT0 interrupt flag |

**Interrupt Flag Register (TM2IR)**

**IP1 (F8H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| PT2 | PCM2 | PCM1 | PCM0 | PCT3 | PCT2 | PCT1 | PCT0 |
| (MSB) | | | | | | | (LSB) |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IP1.7 | PT2 | Timer T2 overflow interrupt(s) priority level |
| IP1.6 | PCM2 | Timer T2 comparator 2 interrupt priority level |
| IP1.5 | PCM1 | Timer T2 comparator 1 interrupt priority level |
| IP1.4 | PCM0 | Timer T2 comparator 0 interrupt priority level |
| IP1.3 | PCT3 | Timer T2 capture register 3 interrupt priority level |
| IP1.2 | PCT2 | Timer T2 capture register 2 interrupt priority level |
| IP1.1 | PCT1 | Timer T2 capture register 1 interrupt priority level |
| IP1.0 | PCT0 | Timer T2 capture register 0 interrupt priority level |

**Timer 2 Interrupt Priority Register *IP1)**

SU00761

**Figure 8. Interrupt Flag Register (TM2IR) and Timer T2 Interrupt Priority Register (IP1)**

**Timer T3, The Watchdog Timer**

In addition to Timer T2 and the standard timers, a watchdog timer is also incorporated on the 8XC552. The purpose of a watchdog timer is to reset the microcontroller if it enters erroneous processor states (possibly caused by electrical noise or RFI) within a reasonable period of time. An analogy is the "dead man's handle" in railway locomotives. When enabled, the watchdog circuitry will generate a system reset if the user program fails to reload the watchdog timer within a specified length of time known as the "watchdog interval."

**Watchdog Circuit Description:** The watchdog timer (Timer T3) consists of an 8-bit timer with an 11-bit prescaler as shown in Figure 9. The prescaler is fed with a signal whose frequency is 1/12 the oscillator frequency (1MHz with a 12MHz oscillator). The 8-bit timer is incremented every "t" seconds, where:

$t = 12 \times 2048 \times 1/f_{OSC}$
(= 1.5ms at $f_{OSC}$ = 16MHz; = 1ms at $f_{OSC}$ = 24MHz)

If the 8-bit timer overflows, a short internal reset pulse is generated which will reset the 8XC552. A short output reset pulse is also generated at the RST pin. This short output pulse (3 machine cycles) may be destroyed if the RST pin is connected to a capacitor. This would not, however, affect the internal reset operation.

Watchdog operation is activated when external pin $\overline{EW}$ is tied low. When $\overline{EW}$ is tied low, it is impossible to disable the watchdog operation by software.

**How to Operate the Watchdog Timer:** The watchdog timer has to be reloaded within periods that are shorter than the programmed watchdog interval; otherwise the watchdog timer will overflow and a system reset will be generated. The user program must therefore continually execute sections of code which reload the watchdog timer. The period of time elapsed between execution of these sections of code must never exceed the watchdog interval. When using a 16MHz oscillator, the watchdog interval is programmable between 1.5ms and 392ms. When using a 24MHz oscillator, the watchdog interval is programmable between 1ms and 255ms.

In order to prepare software for watchdog operation, a programmer should first determine how long his system can sustain an erroneous processor state. The result will be the maximum watchdog interval. As the maximum watchdog interval becomes shorter, it becomes more difficult for the programmer to ensure that the user program always reloads the watchdog timer within the watchdog interval, and thus it becomes more difficult to implement watchdog operation.

The programmer must now partition the software in such a way that reloading of the watchdog is carried out in accordance with the above requirements. The programmer must determine the execution times of all software modules. The effect of possible conditional branches, subroutines, external and internal interrupts must all be taken into account. Since it may be very difficult to evaluate the execution times of some sections of code, the programmer should use worst case estimations. In any event, the programmer must make sure that the watchdog is not activated during normal operation.

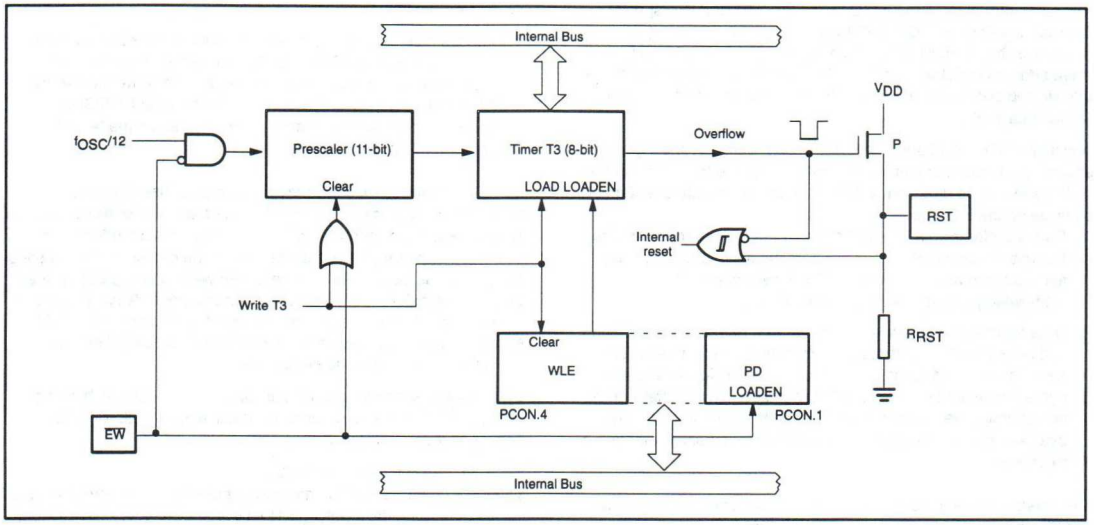



**Figure 9. Watchdog Timer**

The watchdog timer is reloaded in two stages in order to prevent erroneous software from reloading the watchdog. First PCON.4 (WLE) must be set. The T3 may be loaded. When T3 is loaded, PCON.4 (WLE) is automatically reset. T3 cannot be loaded if PCON.4 (WLE) is reset. Reload code may be put in a subroutine as it is called frequently. Since Timer T3 is an up-counter, a reload value of 00H gives the maximum watchdog interval (510ms with a 12MHz oscillator), and a reload value of 0FFH gives the minimum watchdog interval (2ms with a 12MHz oscillator).

In the idle mode, the watchdog circuitry remains active. When watchdog operation is implemented, the power-down mode cannot be used since both states are contradictory. Thus, when watchdog operation is enabled by tying external pin $\overline{EW}$ low, it is impossible to enter the power-down mode, and an attempt to set the power-down bit (PCON.1) will have no effect. PCON.1 will remain at logic 0.

During the early stages of software development/debugging, the watchdog may be disabled by tying the $\overline{EW}$ pin high. At a later stage, $\overline{EW}$ may be tied low to complete the debugging process.

**Watchdog Software Example:** The following example shows how watchdog operation might be handled in a user program.

```
;at the program start:

T3          EQU  0FFH ;address of watchdog timer T3
PCON        EQU  087H ;address of PCON SFR
WATCH-INTV  EQU  156  ;watchdog interval (e.g., 2x100ms)

;to be inserted at each watchdog reload location within
;the user program:

LCALL WATCHDOG

;watchdog service routine:

WATCHDOG: ORL  PCON,#10H ;set condition flag (PCON.4)
          MOV T3,WATCH-INV ;load T3 with  watchdog interval
          RET
```

If it is possible for this subroutine to be called in an erroneous state, then the condition flag WLE should be set at different parts of the main program.

**Serial I/O**

The 8XC552 is equipped with two independent serial ports: SIO0 and SIO1. SIO0 is a full duplex UART port and is identical to the 80C51 serial port. SIO1 accommodates the I²C bus.

**SIO0:** SIO0 is a full duplex serial I/O port identical to that on the 80C51. Its operation is the same, including the use of timer 1 as a baud rate generator.

**SIO1, I²C Serial I/O:** The I²C bus uses two wires (SDA and SCL) to transfer information between devices connected to the bus. The main features of the bus are:

- Bidirectional data transfer between masters and slaves
- Multimaster bus (no central master)
- Arbitration between simultaneously transmitting masters without corruption of serial data on the bus
- Serial clock synchronization allows devices with different bit rates to communicate via one serial bus
- Serial clock synchronization can be used as a handshake mechanism to suspend and resume serial transfer
- The I²C bus may be used for test and diagnostic purposes

The output latches of P1.6 and P1.7 must be set to logic 1 in order to enable SIO1.

The 8XC552 on-chip I²C logic provides a serial interface that meets the I²C bus specification and supports all transfer modes (other than the low-speed mode) from and to the I²C bus. The SIO1 logic handles bytes transfer autonomously. It also keeps track of serial transfers, and a status register (S1STA) reflects the status of SIO1 and the I²C bus.

The CPU interfaces to the I²C logic via the following four special function registers: S1CON (SIO1 control register), S1STA (SIO1 status register), S1DAT (SIO1 data register), and S1ADR (SIO1 slave address register). The SIO1 logic interfaces to the external I²C bus via two port 1 pins: P1.6/SCL (serial clock line) and P1.7/SDA (serial data line).

A typical I²C bus configuration is shown in Figure 10, and Figure 11 shows how a data transfer is accomplished on the bus. Depending on the state of the direction bit (R/W), two types of data transfers are possible on the I²C bus:

1. Data transfer from a master transmitter to a slave receiver. The first byte transmitted by the master is the slave address. Next follows a number of data bytes. The slave returns an acknowledge bit after each received byte.
2. Data transfer from a slave transmitter to a master receiver. The first byte (the slave address) is transmitted by the master. The slave then returns an acknowledge bit. Next follows the data bytes transmitted by the slave to the master. The master returns an acknowledge bit after all received bytes other than the last byte. At the end of the last received byte, a "not acknowledge" is returned.

The master device generates all of the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a repeated START condition. Since a repeated START condition is also the beginning of the next serial transfer, the I²C bus will not be released.

**Modes of Operation:** The on-chip SIO1 logic may operate in the following four modes:

1. Master Transmitter Mode:

   Serial data output through P1.7/SDA while P1.6/SCL outputs the serial clock. The first byte transmitted contains the slave address of the receiving device (7 bits) and the data direction bit. In this case the data direction bit (R/$\overline{W}$) will be logic 0, and we say that a "W" is transmitted. Thus the first byte transmitted is SLA+W. Serial data is transmitted 8 bits at a time. After each byte is transmitted, an acknowledge bit is received. START and STOP conditions are output to indicate the beginning and the end of a serial transfer.

2. Master Receiver Mode:

   The first byte transmitted contains the slave address of the transmitting device (7 bits) and the data direction bit. In this case the data direction bit (R/$\overline{W}$) will be logic 1, and we say that an "R" is transmitted. Thus the first byte transmitted is SLA+R. Serial data is received via P1.7/SDA while P1.6/SCL outputs the serial clock. Serial data is received 8 bits at a time. After each byte is received, an acknowledge bit is transmitted. START and STOP conditions are output to indicate the beginning and end of a serial transfer.

3. Slave Receiver Mode:

   Serial data and the serial clock are received through P1.7/SDA and P1.6/SCL. After each byte is received, an acknowledge bit is transmitted. START and STOP conditions are recognized as the beginning and end of a serial transfer. Address recognition is performed by hardware after reception of the slave address and direction bit.

4. Slave Transmitter Mode:

   The first byte is received and handled as in the slave receiver mode. However, in this mode, the direction bit will indicate that the transfer direction is reversed. Serial data is transmitted via P1.7/SDA while the serial clock is input through P1.6/SCL. START and STOP conditions are recognized as the beginning and end of a serial transfer.

In a given application, SIO1 may operate as a master and as a slave. In the slave mode, the SIO1 hardware looks for its own slave address and the general call address. If one of these addresses is detected, an interrupt is requested. When the microcontroller wishes to become the bus master, the hardware waits until the bus is free before the master mode is entered so that a possible slave action is not interrupted. If bus arbitration is lost in the master mode, SIO1 switches to the slave mode immediately and can detect its own slave address in the same serial transfer.

**SIO1 Implementation and Operation:** Figure 12 shows how the on-chip I²C bus interface is implemented, and the following text describes the individual blocks.

INPUT FILTERS AND OUTPUT STAGES

The input filters have I²C compatible input levels. If the input voltage is less than 1.5V, the input logic level is interpreted as 0; if the input voltage is greater than 3.0V, the input logic level is interpreted as 1. Input signals are synchronized with the internal clock ($f_{OSC}/4$), and spikes shorter than three oscillator periods are filtered out.

The output stages consist of open drain transistors that can sink 3mA at $V_{OUT} < 0.4V$. These open drain outputs do not have clamping diodes to $V_{DD}$. Thus, if the device is connected to the I²C bus and $V_{DD}$ is switched off, the I²C bus is not affected.

ADDRESS REGISTER, S1ADR

This 8-bit special function register may be loaded with the 7-bit slave address (7 most significant bits) to which SIO1 will respond when programmed as a slave transmitter or receiver. The LSB (GC) is used to enable general call address (00H) recognition.

COMPARATOR

The comparator compares the received 7-bit slave address with its own slave address (7 most significant bits in S1ADR). It also compares the first received 8-bit byte with the general call address (00H). If an equality is found, the appropriate status bits are set and an interrupt is requested.

SHIFT REGISTER, S1DAT

This 8-bit special function register contains a byte of serial data to be transmitted or a byte which has just been received. Data in S1DAT is always shifted from right to left; the first bit to be transmitted is the MSB (bit 7) and, after a byte has been received, the first bit of received data is located at the MSB of S1DAT. While data is being shifted out, data on the bus is simultaneously being shifted in; S1DAT always contains the last byte present on the bus. Thus, in the event of lost arbitration, the transition from master transmitter to slave receiver is made with the correct data in S1DAT.

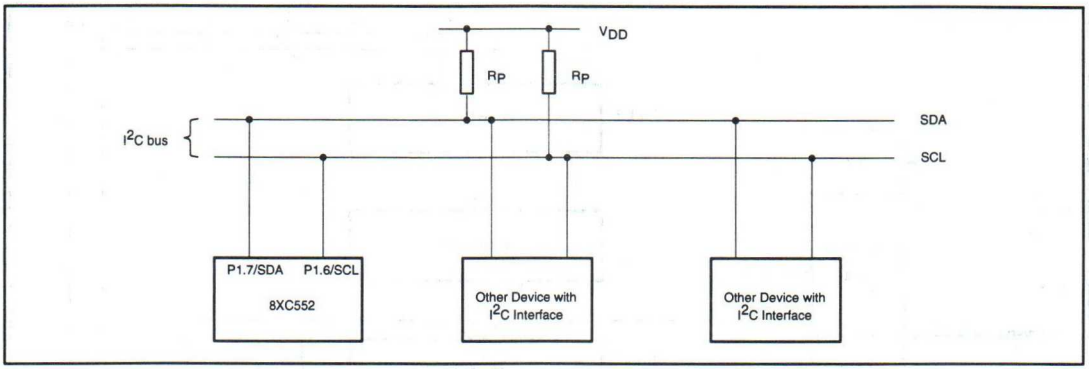



**Figure 10. Typical I²C Bus Configuration**

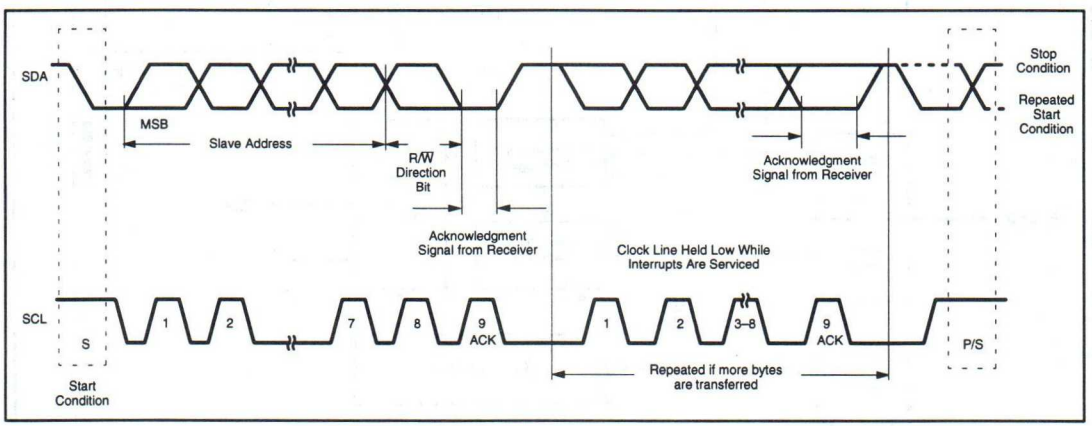



**Figure 11. Data Transfer on the I²C Bus**

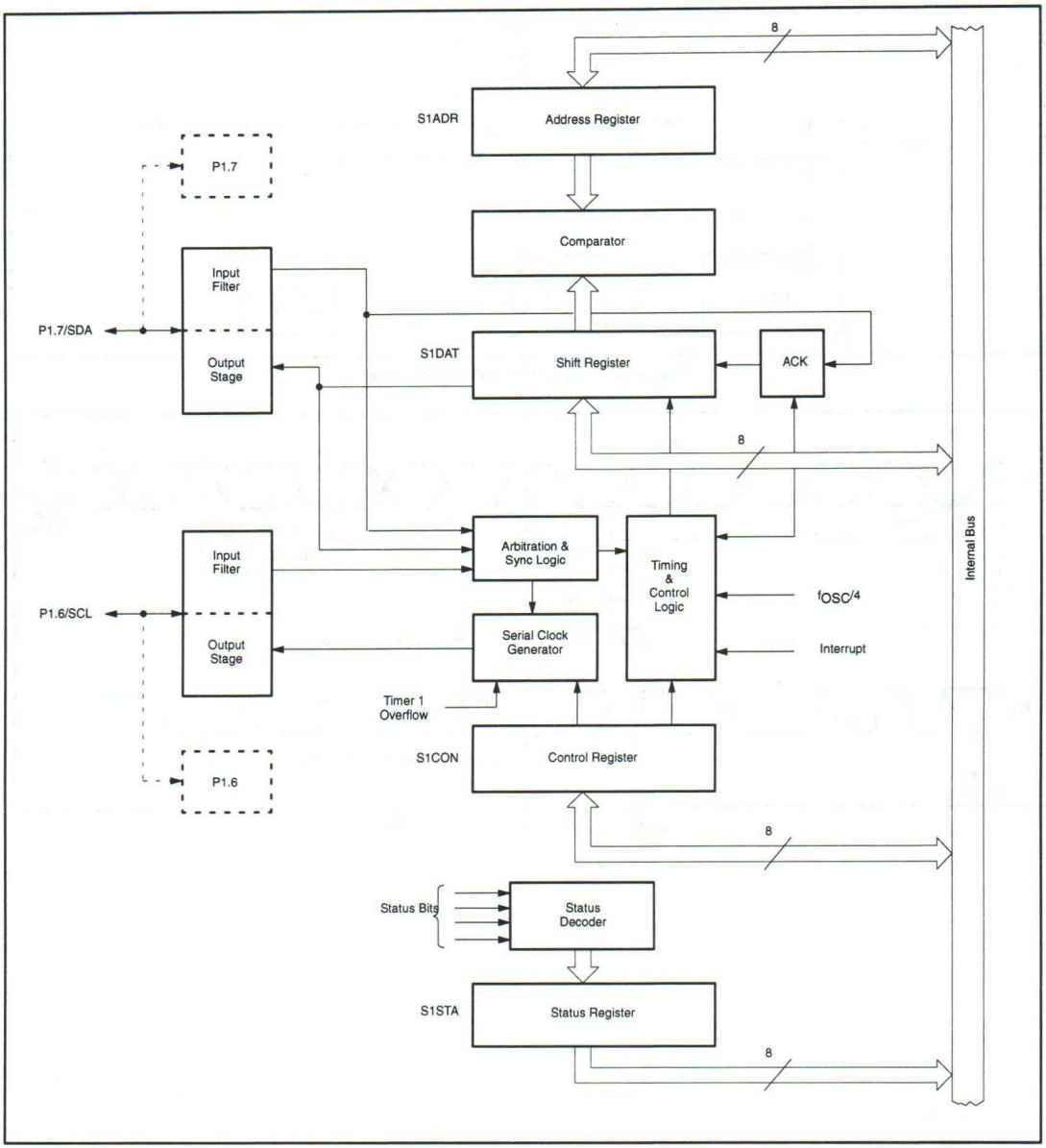
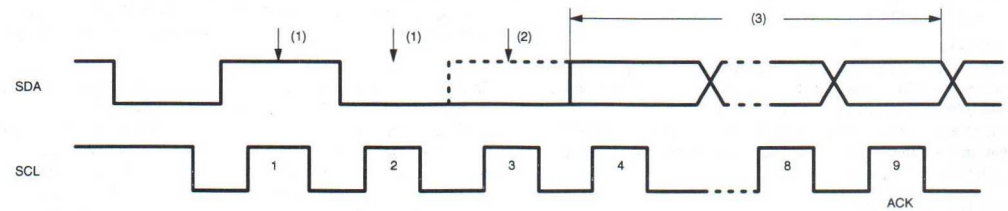
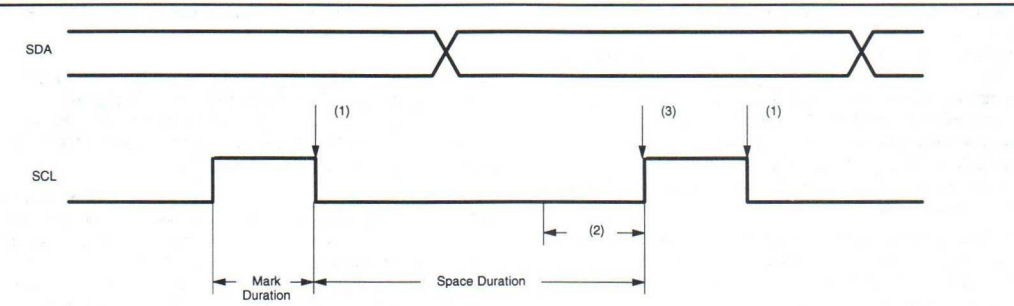



**Figure 12. I²C Bus Serial Interface Block Diagram**




1. Another device transmits identical serial data.
2. Another device overrules a logic 1 (dotted line) transmitted by SIO1 (master) by pulling the SDA line low. Arbitration is lost, and SIO1 enters the slave receiver mode.
3. SIO1 is in the slave receiver mode but still generates clock pulses until the current byte has been transmitted. SIO1 will not generate clock pulses for the next byte. Data on SDA originates from the new master once it has won arbitration.

**Figure 13. Arbitration Procedure**




1. Another service pulls the SCL line low before the SIO1 "mark" duration is complete. The serial clock generator is immediately reset and commences with the "space" duration by pulling SCL low.
2. Another device still pulls the SCL line low after SIO1 releases SCL. The serial clock generator is forced into the wait state until the SCL line is released.
3. The SCL line is released, and the serial clock generator commences with the mark duration.

**Figure 14. Serial Clock Synchronization**

ARBITRATION AND SYNCHRONIZATION LOGIC

In the master transmitter mode, the arbitration logic checks that every transmitted logic 1 actually appears as a logic 1 on the $I^2C$ bus. If another device on the bus overrules a logic 1 and pulls the SDA line low, arbitration is lost, and SIO1 immediately changes from master transmitter to slave receiver. SIO1 will continue to output clock pulses (on SCL) until transmission of the current serial byte is complete.

Arbitration may also be lost in the master receiver mode. Loss of arbitration in this mode can only occur while SIO1 is returning a "not acknowledge: (logic 1) to the bus. Arbitration is lost when another device on the bus pulls this signal LOW. Since this can occur only at the end of a serial byte, SIO1 generates no further clock pulses. Figure 13 shows the arbitration procedure.

The synchronization logic will synchronize the serial clock generator with the clock pulses on the SCL line from another device. If two or more master devices generate clock pulses, the "mark" duration is determined by the device that generates the shortest "marks," and the "space" duration is determined by the device that generates the longest "spaces." Figure 14 shows the synchronization procedure.

A slave may stretch the space duration to slow down the bus master. The space duration may also be stretched for handshaking purposes. This can be done after each bit or after a complete byte transfer. SIO1 will stretch the SCL space duration after a byte has been transmitted or received and the acknowledge bit has been transferred. The serial interrupt flag (SI) is set, and the stretching continues until the serial interrupt flag is cleared.

SERIAL CLOCK GENERATOR

This programmable clock pulse generator provides the SCL clock pulses when SIO1 is in the master transmitter or master receiver mode. It is switched off when SIO1 is in a slave mode. The programmable output clock frequencies are: $f_{OSC}/120$, $f_{OSC}/9600$, and the Timer 1 overflow rate divided by eight. The output clock

pulses have a 50% duty cycle unless the clock generator is synchronized with other SCL clock sources as described above.

TIMING AND CONTROL

The timing and control logic generates the timing and control signals for serial byte handling. This logic block provides the shift pulses for S1DAT, enables the comparator, generates and detects start and stop conditions, receives and transmits acknowledge bits, controls the master and slave modes, contains interrupt request logic, and monitors the $I^2C$ bus status.

CONTROL REGISTER, S1CON

This 7-bit special function register is used by the microcontroller to control the following SIO1 functions: start and restart of a serial transfer, termination of a serial transfer, bit rate, address recognition, and acknowledgment.

STATUS DECODER AND STATUS REGISTER

The status decoder takes all of the internal status bits and compresses them into a 5-bit code. This code is unique for each $I^2C$ bus status. The 5-bit code may be used to generate vector addresses for fast processing of the various service routines. Each service routine processes a particular bus status. There are 26 possible bus states if all four modes of SIO1 are used. The 5-bit status code is latched into the five most significant bits of the status register when the serial interrupt flag is set (by hardware) and remains stable until the interrupt flag is cleared by software. The three least significant bits of the status register are always zero. If the status code is used as a vector to service routines, then the routines are displaced by eight address locations. Eight bytes of code is sufficient for most of the service routines (see the software example in this section).

**The Four SIO1 Special Function Registers:** The microcontroller interfaces to SIO1 via four special function registers. These four SFRs (S1ADR, S1DAT, S1CON, and S1STA) are described individually in the following sections.

**The Address Register, S1ADR:** The CPU can read from and write to this 8-bit, directly addressable SFR. S1ADR is not affected by the SIO1 hardware. The contents of this register are irrelevant when SIO1 is in a master mode. In the slave modes, the seven most significant bits must be loaded with the microcontroller's own slave address, and, if the least significant bit is set, the general call address (00H) is recognized; otherwise it is ignored.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| S1ADR (DBH) | X | X | X | X | X | X | X | GC |

own slave address

The most significant bit corresponds to the first bit received from the $I^2C$ bus after a start condition. A logic 1 in S1ADR corresponds to a high level on the $I^2C$ bus, and a logic 0 corresponds to a low level on the bus.

**The Data Register, S1DAT:** S1DAT contains a byte of serial data to be transmitted or a byte which has just been received. The CPU can read from and write to this 8-bit, directly addressable SFR while it is not in the process of shifting a byte. This occurs when SIO1 is in a defined state and the serial interrupt flag is set. Data in S1DAT remains stable as long as SI is set. Data in S1DAT is always shifted from right to left: the first bit to be transmitted is the MSB (bit 7), and, after a byte has been received, the first bit of received data is located at the MSB of S1DAT. While data is being shifted out, data on the bus is simultaneously being shifted in; S1DAT always contains the last data byte present on the bus. Thus, in the event of lost arbitration, the transition from master transmitter to slave receiver is made with the correct data in S1DAT.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| S1DAT (DAH) | SD7 | SD6 | SD5 | SD4 | SD3 | SD2 | SD1 | SD0 |

shift direction

SD7 - SD0:

Eight bits to be transmitted or just received. A logic 1 in S1DAT corresponds to a high level on the $I^2C$ bus, and a logic 0 corresponds to a low level on the bus. Serial data shifts through S1DAT from right to left. Figure 15 shows how data in S1DAT is serially transferred to and from the SDA line.

S1DAT and the ACK flag form a 9-bit shift register which shifts in or shifts out an 8-bit byte, followed by an acknowledge bit. The ACK flag is controlled by the SIO1 hardware and cannot be accessed by the CPU. Serial data is shifted through the ACK flag into S1DAT on the rising edges of serial clock pulses on the SCL line. When a byte has been shifted into S1DAT, the serial data is available in S1DAT, and the acknowledge bit is returned by the control logic during the ninth clock pulse. Serial data is shifted out from S1DAT via a buffer (BSD7) on the falling edges of clock pulses on the SCL line.

When the CPU writes to S1DAT, BSD7 is loaded with the content of S1DAT.7, which is the first bit to be transmitted to the SDA line (see Figure 16). After nine serial clock pulses, the eight bits in S1DAT will have been transmitted to the SDA line, and the acknowledge bit will be present in ACK. Note that the eight transmitted bits are shifted back into S1DAT.

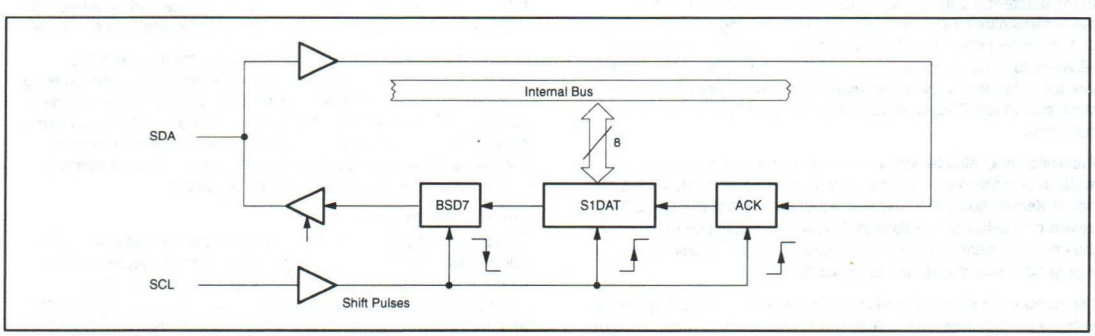



**Figure 15. Serial Input/Output Configuration**

**The Control Register, S1CON:** The CPU can read from and write to this 8-bit, directly addressable SFR. Two bits are affected by the SIO1 hardware: the SI bit is set when a serial interrupt is requested, and the STO bit is cleared when a STOP condition is present on the $I^2C$ bus. The STO bit is also cleared when ENS1 = "0".

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| S1CON (D8H) | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |

ENS1, THE SIO1 ENABLE BIT

ENS1 = "0": When ENS1 is "0", the SDA and SCL outputs are in a high impedance state. SDA and SCL input signals are ignored, SIO1 is in the "not addressed" slave state, and the STO bit in S1CON is forced to "0". No other bits are affected. P1.6 and P1.7 may be used as open drain I/O ports.

ENS1 = "1": When ENS1 is "1", SIO1 is enabled. The P1.6 and P1.7 port latches must be set to logic 1.

ENS1 should not be used to temporarily release SIO1 from the I2C bus since, when ENS1 is reset, the I2C bus status is lost. The AA flag should be used instead (see description of the AA flag in the following text).

In the following text, it is assumed that ENS1 = "1".

STA, THE START FLAG

STA = "1": When the STA bit is set to enter a master mode, the SIO1 hardware checks the status of the I2C bus and generates a START condition if the bus is free. If the bus is not free, then SIO1 waits for a STOP condition (which will free the bus) and generates a START condition after a delay of a half clock period of the internal serial clock generator.

If STA is set while SIO1 is already in a master mode and one or more bytes are transmitted or received, SIO1 transmits a repeated START condition. STA may be set at any time. STA may also be set when SIO1 is an addressed slave.

STA = "0": When the STA bit is reset, no START condition or repeated START condition will be generated.

STO, THE STOP FLAG

STO = "1": When the STO bit is set while SIO1 is in a master mode, a STOP condition is transmitted to the $I^2C$ bus. When the STOP condition is detected on the bus, the SIO1 hardware clears the STO flag. In a slave mode, the STO flag may be set to recover from an error condition. In this case, no STOP condition is transmitted to the $I^2C$ bus. However, the SIO1 hardware behaves as if a STOP condition has been received and switches to the defined "not addressed" slave receiver mode. The STO flag is automatically cleared by hardware.

If the STA and STO bits are both set, the a STOP condition is transmitted to the $I^2C$ bus if SIO1 is in a master mode (in a slave mode, SIO1 generates an internal STOP condition which is not transmitted). SIO1 then transmits a START condition.

STO = "0": When the STO bit is reset, no STOP condition will be generated.

SI, THE SERIAL INTERRUPT FLAG

SI = "1": When the SI flag is set, then, if the EA and ES1 (interrupt enable register) bits are also set, a serial interrupt is requested. SI is set by hardware when one of 25 of the 26 possible SIO1 states is entered. The only state that does not cause SI to be set is state F8H, which indicates that no relevant state information is available.

While SI is set, the low period of the serial clock on the SCL line is stretched, and the serial transfer is suspended. A high level on the SCL line is unaffected by the serial interrupt flag. SI must be reset by software.

SI = "0": When the SI flag is reset, no serial interrupt is requested, and there is no stretching of the serial clock on the SCL line.

AA, THE ASSERT ACKNOWLEDGE FLAG

AA = "1": If the AA flag is set, an acknowledge (low level to SDA) will be returned during the acknowledge clock pulse on the SCL line when:

- The "own slave address" has been received
- The general call address has been received while the general call bit (GC) in S1ADR is set
- A data byte has been received while SIO1 is in the master receiver mode
- A data byte has been received while SIO1 is in the addressed slave receiver mode

AA = "0": if the AA flag is reset, a not acknowledge (high level to SDA) will be returned during the acknowledge clock pulse on SCL when:

- A data has been received while SIO1 is in the master receiver mode
- A data byte has been received while SIO1 is in the addressed slave receiver mode

When SIO1 is in the addressed slave transmitter mode, state C8H will be entered after the last serial is transmitted (see Figure 20). When SI is cleared, SIO1 leaves state C8H, enters the not addressed slave receiver mode, and the SDA line remains at a high level. In state C8H, the AA flag can be set again for future address recognition.

When SIO1 is in the not addressed slave mode, its own slave address and the general call address are ignored. Consequently, no acknowledge is returned, and a serial interrupt is not requested. Thus, SIO1 can be temporarily released from the $I^2C$ bus while the bus status is monitored. While SIO1 is released from the bus, START and STOP conditions are detected, and serial data is shifted in. Address recognition can be resumed at any time by setting the AA flag. If the AA flag is set when the part's own slave address or the general call address has been partly received, the address will be recognized at the end of the byte transmission.

CR0, CR1, AND CR2, THE CLOCK RATE BITS

These three bits determine the serial clock frequency when SIO1 is in a master mode. The various serial rates are shown in Table 2.

A 12.5kHz bit rate may be used by devices that interface to the $I^2C$ bus via standard I/O port lines which are software driven and slow. 100kHz is usually the maximum bit rate and can be derived from a 16MHz, 12MHz, or a 6MHz oscillator. A variable bit rate (0.5kHz to 62.5kHz) may also be used if Timer 1 is not required for any other purpose while SIO1 is in a master mode.

The frequencies shown in Table 2 are unimportant when SIO1 is in a slave mode. In the slave modes, SIO1 will automatically synchronize with any clock frequency up to 100kHz.

**The Status Register, S1STA:** S1STA is an 8-bit read-only special function register. The three least significant bits are always zero. The five most significant bits contain the status code. There are 26 possible status codes. When S1STA contains F8H, no relevant state information is available and no serial interrupt is requested. All other S1STA values correspond to defined SIO1 states. When each of these states is entered, a serial interrupt is requested (SI = "1"). A valid status code is present in S1STA one machine cycle after SI is set by hardware and is still present one machine cycle after SI has been reset by software.

**More Information on SIO1 Operating Modes:** The four operating modes are:

- Master Transmitter
- Master Receiver
- Slave Receiver
- Slave Transmitter

Data transfers in each mode of operation are shown in Figures 17–37. These figures contain the following abbreviations:

| Abbreviation | Explanation |
|---|---|
| S | Start condition |
| SLA | 7-bit slave address |
| R | Read bit (high level at SDA) |
| W | Write bit (low level at SDA) |
| A | Acknowledge bit (low level at SDA) |
| $\overline{A}$ | Not acknowledge bit (high level at SDA) |
| Data | 8-bit data byte |
| P | Stop condition |

In Figures 17-37, circles are used to indicate when the serial interrupt flag is set. The numbers in the circles show the status code held in the S1STA register. At these points, a service routine must be executed to continue or complete the serial transfer. These service routines are not critical since the serial transfer is suspended until the serial interrupt flag is cleared by software.

When a serial interrupt routine is entered, the status code in S1STA is used to branch to the appropriate service routine. For each status code, the required software action and details of the following serial transfer are given in Tables 3-7.

**Master Transmitter Mode:** In the master transmitter mode, a number of data bytes are transmitted to a slave receiver (see Figure 17). Before the master transmitter mode can be entered, S1CON must be initialized as follows:

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| S1CON (D8H) | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |
| | bit rate | 1 | 0 | 0 | 0 | X | bit rate | |

CR0, CR1, and CR2 define the serial bit rate. ENS1 must be set to logic 1 to enable SIO1. If the AA bit is reset, SIO1 will not acknowledge its own slave address or the general call address in the event of another device becoming master of the bus. In other words, if AA is reset, SIO0 cannot enter a slave mode. STA, STO, and SI must be reset.

The master transmitter mode may now be entered by setting the STA bit using the SETB instruction. The SIO1 logic will now test the $I^2C$ bus and generate a start condition as soon as the bus becomes free. When a START condition is transmitted, the serial interrupt flag (SI) is set, and the status code in the status register (S1STA) will be 08H. This status code must be used to vector to an interrupt service routine that loads S1DAT with the slave address and the data direction bit (SLA+W). The SI bit in S1CON must then be reset before the serial transfer can continue.

When the slave address and the direction bit have been transmitted and an acknowledgment bit has been received, the serial interrupt flag (SI) is set again, and a number of status codes in S1STA are possible. There are 18H, 20H, or 38H for the master mode and also 68H, 78H, or B0H if the slave mode was enabled (AA = logic 1). The appropriate action to be taken for each of these status codes is detailed in Table 3. After a repeated start condition (state 10H). SIO1 may switch to the master receiver mode by loading S1DAT with SLA+R).

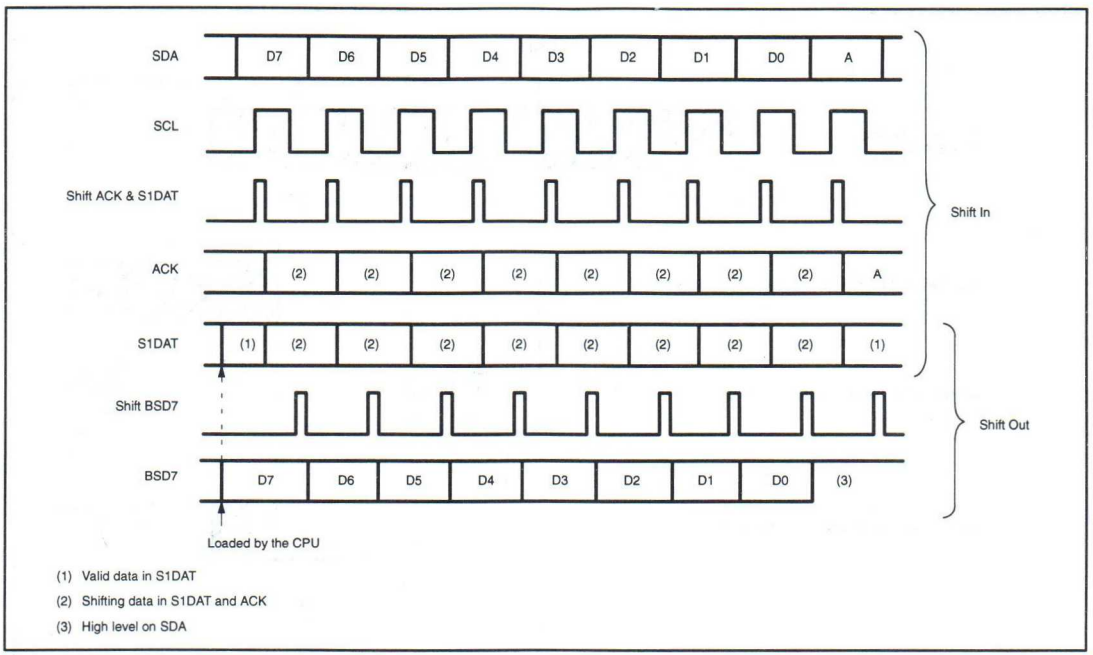



**Figure 16. Shift-in and Shift-out Timing**

## Table 2. Serial Clock Rates

| CR2 | CR1 | CR0 | BIT FREQUENCY (kHz) AT $f_{OSC}$ 6MHz | 12MHz | 16MHz | $f_{OSC}$ DIVIDED BY |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 23 | 47 | 63 | 256 |
| 0 | 0 | 1 | 27 | 54 | 71 | 224 |
| 0 | 1 | 0 | 31 | 63 | 83 | 192 |
| 0 | 1 | 1 | 37 | 75 | 100 | 160 |
| 1 | 0 | 0 | 6.25 | 12.5 | 17 | 960 |
| 1 | 0 | 1 | 50 | 100 | – | 120 |
| 1 | 1 | 0 | 100 | – | – | 60 |
| 1 | 1 | 1 | 0.25 < 62.5 | 0.5 < 62.5 | 0.67 < 56 | 96 × (256 – reload value Timer 1) (Reload value range: 0 – 254 in mode 2) |

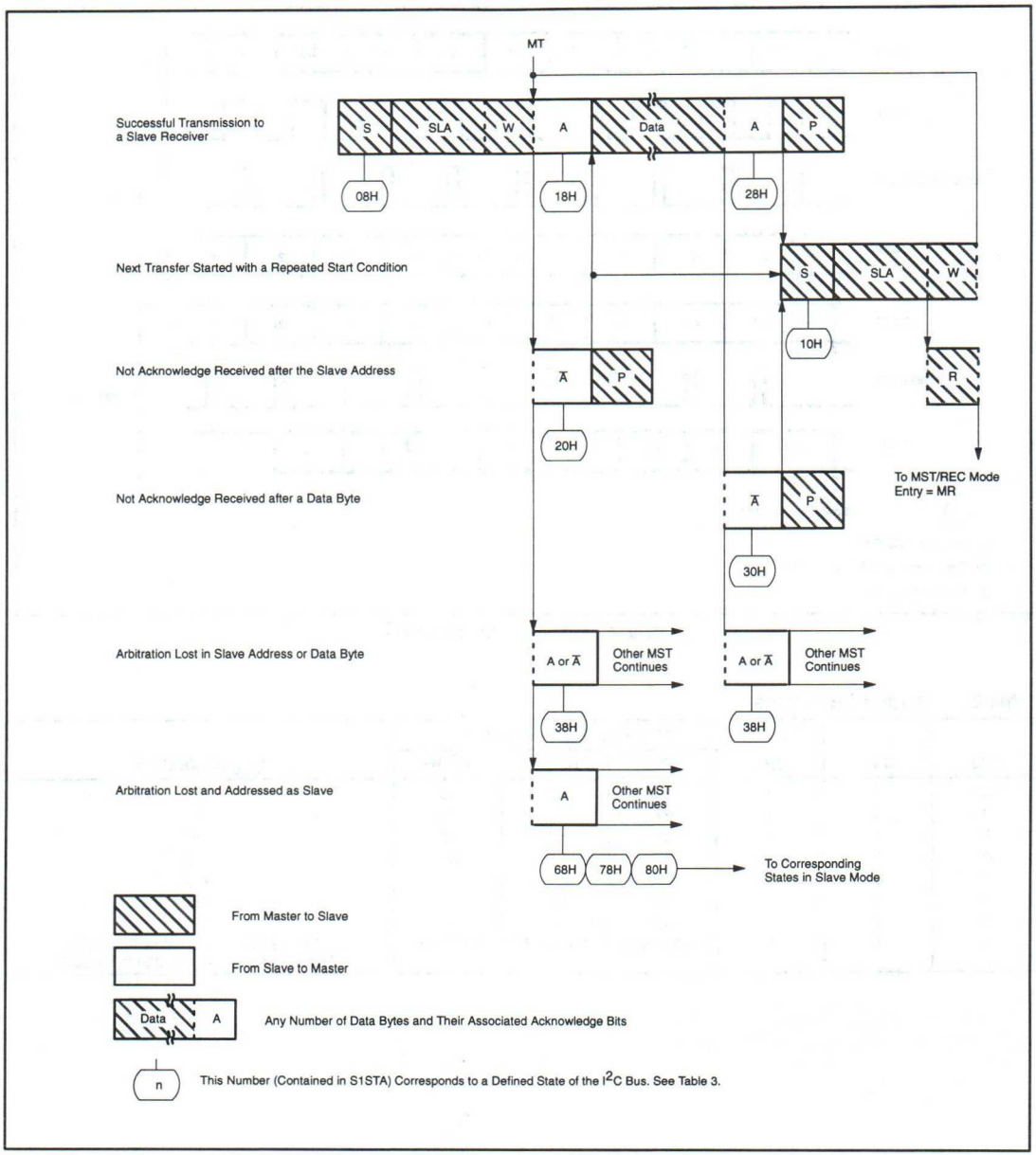



**Figure 17. Format and States in the Master Transmitter Mode**

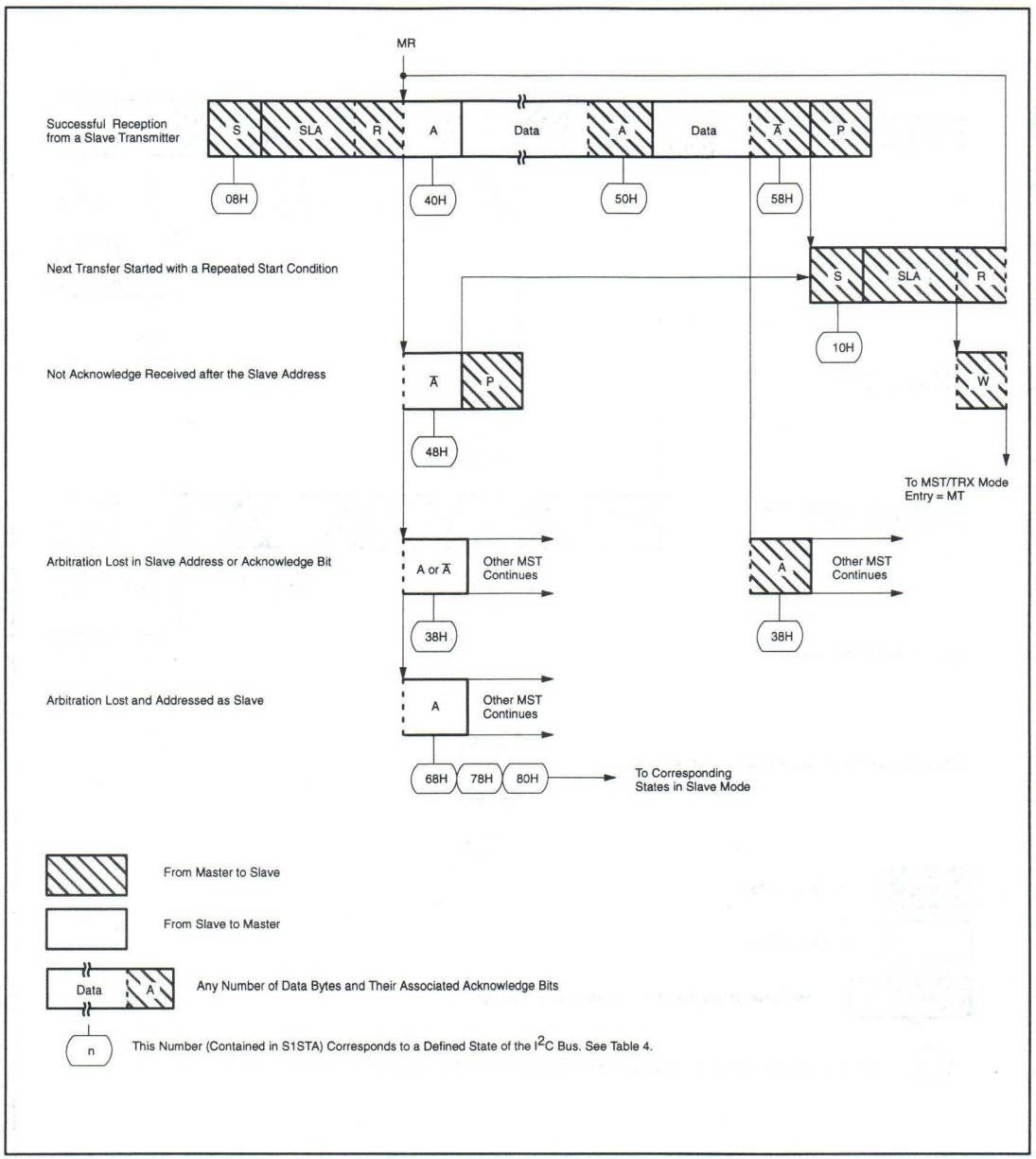



**Figure 18. Format and States in the Master Receiver Mode**

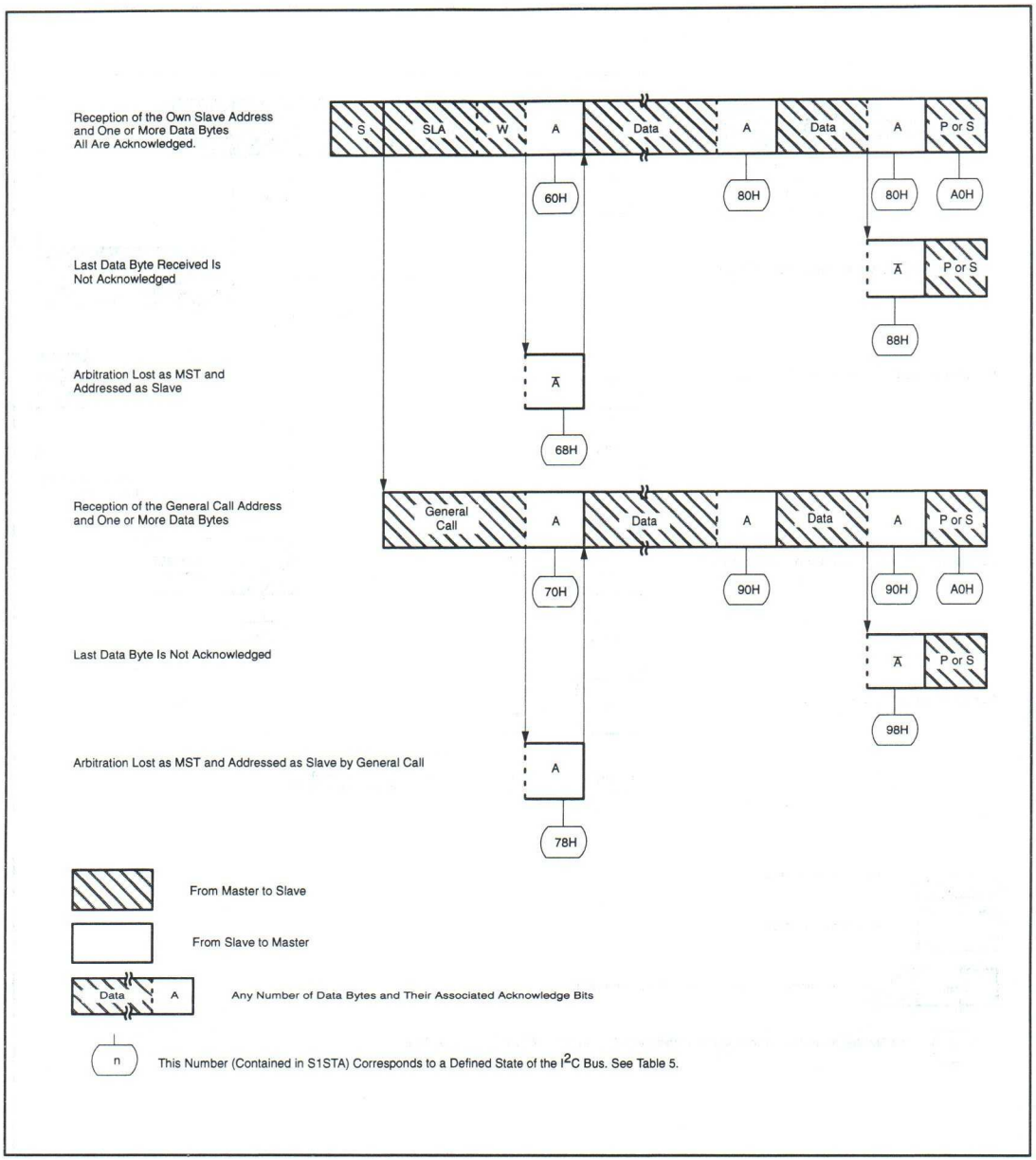



**Figure 19. Format and States in the Slave Receiver Mode**

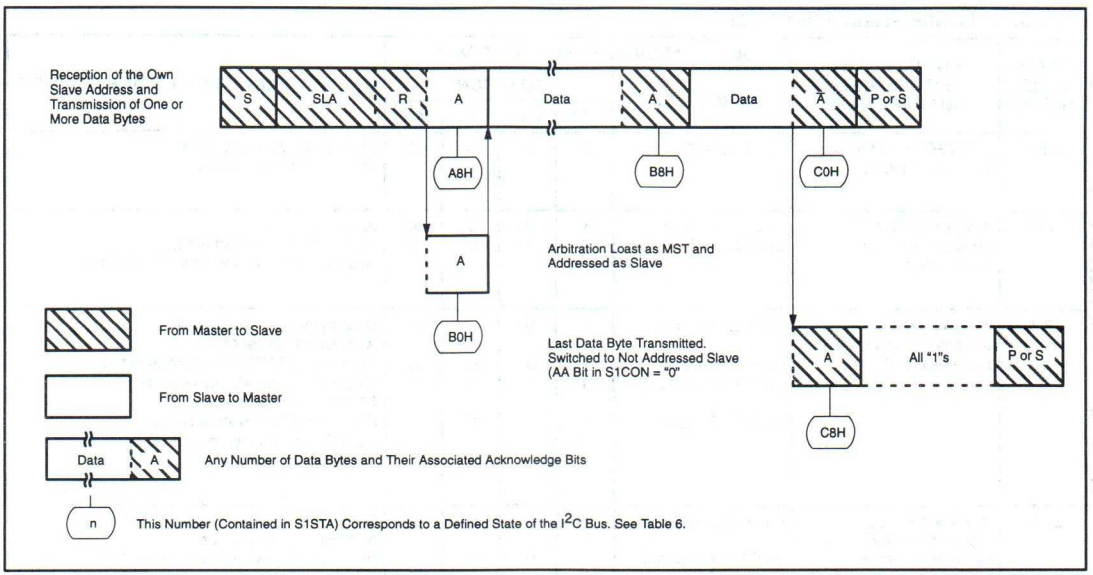



**Figure 20. Format and States of the Slave Transmitter Mode**

**Master Receiver Mode:** In the master receiver mode, a number of data bytes are received from a slave transmitter (see Figure 18). The transfer is initialized as in the master transmitter mode. When the start condition has been transmitted, the interrupt service routine must load S1DAT with the 7-bit slave address and the data direction bit (SLA+R). The SI bit in S1CON must then be cleared before the serial transfer can continue.

When the slave address and the data direction bit have been transmitted and an acknowledgment bit has been received, the serial interrupt flag (SI) is set again, and a number of status codes in S1STA are possible. These are 40H, 48H, or 38H for the master mode and also 68H, 78H, or B0H if the slave mode was enabled (AA = logic 1). The appropriate action to be taken for each of these status codes is detailed in Table 4. ENS1, CR1, and CR0 are not affected by the serial transfer and are not referred to in Table 4. After a repeated start condition (state 10H), SIO1 may switch to the master transmitter mode by loading S1DAT with SLA+W.

**Slave Receiver Mode:** In the slave receiver mode, a number of data bytes are received from a master transmitter (see Figure 19). To initiate the slave receiver mode, S1ADR and S1CON must be loaded as follows:

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| S1ADR (DBH) | X | X | X | X | X | X | X | GC |
| | own slave address | | | | | | | |

The upper 7 bits are the address to which SIO1 will respond when addressed by a master. If the LSB (GC) is set, SIO1 will respond to the general call address (00H); otherwise it ignores the general call address.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| S1CON (D8H) | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |
| | X | 1 | 0 | 0 | 0 | 1 | X | X |

CR0, CR1, and CR2 do not affect SIO1 in the slave mode. ENS1 must be set to logic 1 to enable SIO1. The AA bit must be set to enable SIO1 to acknowledge its own slave address or the general call address. STA, STO, and SI must be reset.

When S1ADR and S1CON have been initialized, SIO1 waits until it is addressed by its own slave address followed by the data direction bit which must be "0" (W) for SIO1 to operate in the slave receiver mode. After its own slave address and the W bit have been received, the serial interrupt flag (I) is set and a valid status code can be read from S1STA. This status code is used to vector to an interrupt service routine, and the appropriate action to be taken for each of these status codes is detailed in Table 5. The slave receiver mode may also be entered if arbitration is lost while SIO1 is in the master mode (see status 68H and 78H).

If the AA bit is reset during a transfer, SIO1 will return a not acknowledge (logic 1) to SDA after the next received data byte. While AA is reset, SIO1 does not respond to its own slave address or a general call address. However, the $I^2C$ bus is still monitored and address recognition may be resumed at any time by setting AA. This means that the AA bit may be used to temporarily isolate SIO1 from the $I^2C$ bus.

**Table 3. Master Transmitter Mode**

| STATUS CODE (S1STA) | STATUS OF THE I²C BUS AND SIO1 HARDWARE | APPLICATION SOFTWARE RESPONSE | | | | | NEXT ACTION TAKEN BY SIO1 HARDWARE |
|---|---|---|---|---|---|---|---|
| | | TO/FROM S1DAT | TO S1CON | | | | |
| | | | STA | STO | SI | AA | |
| 08H | A START condition has been transmitted | Load SLA+W | X | 0 | 0 | X | SLA+W will be transmitted;<br>ACK bit will be received |
| 10H | A repeated START condition has been transmitted | Load SLA+W or<br>Load SLA+R | X<br>X | 0<br>0 | 0<br>0 | X<br>X | As above<br>SLA+W will be transmitted;<br>SIO1 will be switched to MST/REC mode |
| 18H | SLA+W has been transmitted; ACK has been received | Load data byte or<br><br>no S1DAT action or<br>no S1DAT action or<br><br>no S1DAT action | 0<br><br>1<br>0<br><br>1 | 0<br><br>0<br>1<br><br>1 | 0<br><br>0<br>0<br><br>0 | X<br><br>X<br>X<br><br>X | Data byte will be transmitted;<br>ACK bit will be received<br>Repeated START will be transmitted;<br>STOP condition will be transmitted;<br>STO flag will be reset<br>STOP condition followed by a<br>START condition will be transmitted;<br>STO flag will be reset |
| 20H | SLA+W has been transmitted; NOT ACK has been received | Load data byte or<br><br>no S1DAT action or<br>no S1DAT action or<br><br>no S1DAT action | 0<br><br>1<br>0<br><br>1 | 0<br><br>0<br>1<br><br>1 | 0<br><br>0<br>0<br><br>0 | X<br><br>X<br>X<br><br>X | Data byte will be transmitted;<br>ACK bit will be received<br>Repeated START will be transmitted;<br>STOP condition will be transmitted;<br>STO flag will be reset<br>STOP condition followed by a<br>START condition will be transmitted;<br>STO flag will be reset |
| 28H | Data byte in S1DAT has been transmitted; ACK has been received | Load data byte or<br><br>no S1DAT action or<br>no S1DAT action or<br><br>no S1DAT action | 0<br><br>1<br>0<br><br>1 | 0<br><br>0<br>1<br><br>1 | 0<br><br>0<br>0<br><br>0 | X<br><br>X<br>X<br><br>X | Data byte will be transmitted;<br>ACK bit will be received<br>Repeated START will be transmitted;<br>STOP condition will be transmitted;<br>STO flag will be reset<br>STOP condition followed by a<br>START condition will be transmitted;<br>STO flag will be reset |
| 30H | Data byte in S1DAT has been transmitted; NOT ACK has been received | Load data byte or<br><br>no S1DAT action or<br>no S1DAT action or<br><br>no S1DAT action | 0<br><br>1<br>0<br><br>1 | 0<br><br>0<br>1<br><br>1 | 0<br><br>0<br>0<br><br>0 | X<br><br>X<br>X<br><br>X | Data byte will be transmitted;<br>ACK bit will be received<br>Repeated START will be transmitted;<br>STOP condition will be transmitted;<br>STO flag will be reset<br>STOP condition followed by a<br>START condition will be transmitted;<br>STO flag will be reset |
| 38H | Arbitration lost in SLA+R/$\overline{W}$ or Data bytes | No S1DAT action or<br><br>No S1DAT action | 0<br><br>1 | 0<br><br>0 | 0<br><br>0 | X<br><br>X | I²C bus will be released;<br>not addressed slave will be entered<br>A START condition will be transmitted when the bus becomes free |

**Table 4. Master Receiver Mode**

| STATUS CODE (S1STA) | STATUS OF THE I²C BUS AND SIO1 HARDWARE | APPLICATION SOFTWARE RESPONSE | | | | | NEXT ACTION TAKEN BY SIO1 HARDWARE |
|---|---|---|---|---|---|---|---|
| | | TO/FROM S1DAT | TO S1CON | | | | |
| | | | STA | STO | SI | AA | |
| 08H | A START condition has been transmitted | Load SLA+R | X | 0 | 0 | X | SLA+R will be transmitted;<br>ACK bit will be received |
| 10H | A repeated START condition has been transmitted | Load SLA+R or<br>Load SLA+W | X<br>X | 0<br>0 | 0<br>0 | X<br>X | As above<br>SLA+W will be transmitted;<br>SIO1 will be switched to MST/TRX mode |
| 38H | Arbitration lost in NOT ACK bit | No S1DAT action or<br><br>No S1DAT action | 0<br><br>1 | 0<br><br>0 | 0<br><br>0 | X<br><br>X | I²C bus will be released;<br>SIO1 will enter a slave mode<br>A START condition will be transmitted when the bus becomes free |
| 40H | SLA+R has been transmitted; ACK has been received | No S1DAT action or<br><br>no S1DAT action | 0<br><br>0 | 0<br><br>0 | 0<br><br>0 | 0<br><br>1 | Data byte will be received;<br>NOT ACK bit will be returned<br>Data byte will be received;<br>ACK bit will be returned |
| 48H | SLA+R has been transmitted; NOT ACK has been received | No S1DAT action or<br><br>no S1DAT action or<br><br>no S1DAT action | 1<br><br>0<br><br>1 | 0<br><br>1<br><br>1 | 0<br><br>0<br><br>0 | X<br><br>X<br><br>X | Repeated START condition will be transmitted<br><br>STOP condition will be transmitted;<br>STO flag will be reset<br>STOP condition followed by a START condition will be transmitted;<br>STO flag will be reset |
| 50H | Data byte has been received; ACK has been returned | Read data byte or<br><br>read data byte | 0<br><br>0 | 0<br><br>0 | 0<br><br>0 | 0<br><br>1 | Data byte will be received;<br>NOT ACK bit will be returned<br>Data byte will be received;<br>ACK bit will be returned |
| 58H | Data byte has been received; NOT ACK has been returned | Read data byte or<br><br>read data byte or<br><br>read data byte | 1<br><br>0<br><br>1 | 0<br><br>1<br><br>1 | 0<br><br>0<br><br>0 | X<br><br>X<br><br>X | Repeated START condition will be transmitted<br><br>STOP condition will be transmitted;<br>STO flag will be reset<br>STOP condition followed by a START condition will be transmitted;<br>STO flag will be reset |

**Table 5. Slave Receiver Mode**

| STATUS CODE (S1STA) | STATUS OF THE I²C BUS AND SIO1 HARDWARE | APPLICATION SOFTWARE RESPONSE | | | | | NEXT ACTION TAKEN BY SIO1 HARDWARE |
|---|---|---|---|---|---|---|---|
| | | TO/FROM S1DAT | TO S1CON | | | | |
| | | | STA | STO | SI | AA | |
| 60H | Own SLA+W has been received; ACK has been returned | No S1DAT action or | X | 0 | 0 | 0 | Data byte will be received and NOT ACK will be returned |
| | | no S1DAT action | X | 0 | 0 | 1 | Data byte will be received and ACK will be returned |
| 68H | Arbitration lost in SLA+R/$\overline{W}$ as master; Own SLA+W has been received, ACK returned | No S1DAT action or | X | 0 | 0 | 0 | Data byte will be received and NOT ACK will be returned |
| | | no S1DAT action | X | 0 | 0 | 1 | Data byte will be received and ACK will be returned |
| 70H | General call address (00H) has been received; ACK has been returned | No S1DAT action or | X | 0 | 0 | 0 | Data byte will be received and NOT ACK will be returned |
| | | no S1DAT action | X | 0 | 0 | 1 | Data byte will be received and ACK will be returned |
| 78H | Arbitration lost in SLA+R/$\overline{W}$ as master; General call address has been received, ACK has been returned | No S1DAT action or | X | 0 | 0 | 0 | Data byte will be received and NOT ACK will be returned |
| | | no S1DAT action | X | 0 | 0 | 1 | Data byte will be received and ACK will be returned |
| 80H | Previously addressed with own SLV address; DATA has been received; ACK has been returned | Read data byte or | X | 0 | 0 | 0 | Data byte will be received and NOT ACK will be returned |
| | | read data byte | X | 0 | 0 | 1 | Data byte will be received and ACK will be returned |
| 88H | Previously addressed with own SLA; DATA byte has been received; NOT ACK has been returned | Read data byte or | 0 | 0 | 0 | 0 | Switched to not addressed SLV mode; no recognition of own SLA or General call address |
| | | read data byte or | 0 | 0 | 0 | 1 | Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1 |
| | | read data byte or | 1 | 0 | 0 | 0 | Switched to not addressed SLV mode; no recognition of own SLA or General call address. A START condition will be transmitted when the bus becomes free |
| | | read data byte | 1 | 0 | 0 | 1 | Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1. A START condition will be transmitted when the bus becomes free. |
| 90H | Previously addressed with General Call; DATA byte has been received; ACK has been returned | Read data byte or | X | 0 | 0 | 0 | Data byte will be received and NOT ACK will be returned |
| | | read data byte | X | 0 | 0 | 1 | Data byte will be received and ACK will be returned |
| 98H | Previously addressed with General Call; DATA byte has been received; NOT ACK has been returned | Read data byte or | 0 | 0 | 0 | 0 | Switched to not addressed SLV mode; no recognition of own SLA or General call address |
| | | read data byte or | 0 | 0 | 0 | 1 | Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1 |
| | | read data byte or | 1 | 0 | 0 | 0 | Switched to not addressed SLV mode; no recognition of own SLA or General call address. A START condition will be transmitted when the bus becomes free |
| | | read data byte | 1 | 0 | 0 | 1 | Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1. A START condition will be transmitted when the bus becomes free. |

**Table 5. Slave Receiver Mode** (Continued)

| STATUS CODE (S1STA) | STATUS OF THE I²C BUS AND SIO1 HARDWARE | APPLICATION SOFTWARE RESPONSE | | | | | NEXT ACTION TAKEN BY SIO1 HARDWARE |
|---|---|---|---|---|---|---|---|
| | | TO/FROM S1DAT | TO S1CON | | | | |
| | | | STA | STO | SI | AA | |
| A0H | A STOP condition or repeated START condition has been received while still addressed as SLV/REC or SLV/TRX | No STDAT action or | 0 | 0 | 0 | 0 | Switched to not addressed SLV mode; no recognition of own SLA or General call address |
| | | No STDAT action or | 0 | 0 | 0 | 1 | Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1 |
| | | No STDAT action or | 1 | 0 | 0 | 0 | Switched to not addressed SLV mode; no recognition of own SLA or General call address. A START condition will be transmitted when the bus becomes free |
| | | No STDAT action | 1 | 0 | 0 | 1 | Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1. A START condition will be transmitted when the bus becomes free. |

**Table 6. Slave Transmitter Mode**

| STATUS CODE (S1STA) | STATUS OF THE I²C BUS AND SIO1 HARDWARE | APPLICATION SOFTWARE RESPONSE | | | | | NEXT ACTION TAKEN BY SIO1 HARDWARE |
|---|---|---|---|---|---|---|---|
| | | TO/FROM S1DAT | TO S1CON | | | | |
| | | | STA | STO | SI | AA | |
| A8H | Own SLA+R has been received; ACK has been returned | Load data byte or | X | 0 | 0 | 0 | Last data byte will be transmitted and ACK bit will be received |
| | | load data byte | X | 0 | 0 | 1 | Data byte will be transmitted; ACK will be received |
| B0H | Arbitration lost in SLA+R/$\overline{W}$ as master; Own SLA+R has been received, ACK has been returned | Load data byte or | X | 0 | 0 | 0 | Last data byte will be transmitted and ACK bit will be received |
| | | load data byte | X | 0 | 0 | 1 | Data byte will be transmitted; ACK bit will be received |
| B8H | Data byte in S1DAT has been transmitted; ACK has been received | Load data byte or | X | 0 | 0 | 0 | Last data byte will be transmitted and ACK bit will be received |
| | | load data byte | X | 0 | 0 | 1 | Data byte will be transmitted; ACK bit will be received |
| C0H | Data byte in S1DAT has been transmitted; NOT ACK has been received | No S1DAT action or | 0 | 0 | 0 | 0 | Switched to not addressed SLV mode; no recognition of own SLA or General call address |
| | | no S1DAT action or | 0 | 0 | 0 | 1 | Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1 |
| | | no S1DAT action or | 1 | 0 | 0 | 0 | Switched to not addressed SLV mode; no recognition of own SLA or General call address. A START condition will be transmitted when the bus becomes free |
| | | no S1DAT action | 1 | 0 | 0 | 1 | Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1. A START condition will be transmitted when the bus becomes free. |
| C8H | Last data byte in S1DAT has been transmitted (AA = 0); ACK has been received | No S1DAT action or | 0 | 0 | 0 | 0 | Switched to not addressed SLV mode; no recognition of own SLA or General call address |
| | | no S1DAT action or | 0 | 0 | 0 | 1 | Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1 |
| | | no S1DAT action or | 1 | 0 | 0 | 0 | Switched to not addressed SLV mode; no recognition of own SLA or General call address. A START condition will be transmitted when the bus becomes free |
| | | no S1DAT action | 1 | 0 | 0 | 1 | Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1. A START condition will be transmitted when the bus becomes free. |

**Slave Transmitter Mode:** In the slave transmitter mode, a number of data bytes are transmitted to a master receiver (see Figure 20). Data transfer is initialized as in the slave receiver mode. When S1ADR and S1CON have been initialized, SIO1 waits until it is addressed by its own slave address followed by the data direction bit which must be "1" (R) for SIO1 to operate in the slave transmitter mode. After its own slave address and the R bit have been received, the serial interrupt flag (SI) is set and a valid status code can be read from S1STA. This status code is used to vector to an interrupt service routine, and the appropriate action to be taken for each of these status codes is detailed in Table 6. The slave transmitter mode may also be entered if arbitration is lost while SIO1 is in the master mode (see state B0H).

If the AA bit is reset during a transfer, SIO1 will transmit the last byte of the transfer and enter state C0H or C8H. SIO1 is switched to the not addressed slave mode and will ignore the master receiver if it continues the transfer. Thus the master receiver receives all 1s as serial data. While AA is reset, SIO1 does not respond to its own slave address or a general call address. However, the $I^2C$ bus is still monitored, and address recognition may be resumed at any time by setting AA. This means that the AA bit may be used to temporarily isolate SIO1 from the $I^2C$ bus.

**Miscellaneous States:** There are two S1STA codes that do not correspond to a defined SIO1 hardware state (see Table 7). These are discussed below.

**S1STA = F8H:**
This status code indicates that no relevant information is available because the serial interrupt flag, SI, is not yet set. This occurs between other states and when SIO1 is not involved in a serial transfer.

**S1STA = 00H:**
This status code indicates that a bus error has occurred during an SIO1 serial transfer. A bus error is caused when a START or STOP condition occurs at an illegal position in the format frame. Examples of such illegal positions are during the serial transfer of an address byte, a data byte, or an acknowledge bit. A bus error may also be caused when external interference disturbs the internal SIO1 signals. When a bus error occurs, SI is set. To recover from a bus error, the STO flag must be set and SI must be cleared. This causes SIO1 to enter the "not addressed" slave mode (a defined state) and to clear the STO flag (no other bits in S1CON are affected). The SDA and SCL lines are released (a STOP condition is not transmitted).

**Some Special Cases:** The SIO1 hardware has facilities to handle the following special cases that may occur during a serial transfer:

Simultaneous Repeated START Conditions from Two Masters

A repeated START condition may be generated in the master transmitter or master receiver modes. A special case occurs if another master simultaneously generates a repeated START condition (see Figure 21). Until this occurs, arbitration is not lost by either master since they were both transmitting the same data.

If the SIO1 hardware detects a repeated START condition on the $I^2C$ bus before generating a repeated START condition itself, it will release the bus, and no interrupt request is generated. If another master frees the bus by generating a STOP condition, SIO1 will transmit a normal START condition (state 08H), and a retry of the total serial data transfer can commence.

Data Transfer After Loss of Arbitration

Arbitration may be lost in the master transmitter and master receiver modes (see Figure 13). Loss of arbitration is indicated by the following states in S1STA; 38H, 68H, 78H, and B0H (see Figures 17 and 18).

If the STA flag in S1CON is set by the routines which service these states, then, if the bus is free again, a START condition (state 08H) is transmitted without intervention by the CPU, and a retry of the total serial transfer can commence.

Forced Access to the $I^2C$ Bus

In some applications, it may be possible for an uncontrolled source to cause a bus hang-up. In such situations, the problem may be caused by interference, temporary interruption of the bus or a temporary short-circuit between SDA and SCL.

If an uncontrolled source generates a superfluous START or masks a STOP condition, then the $I^2C$ bus stays busy indefinitely. If the STA flag is set and bus access is not obtained within a reasonable amount of time, then a forced access to the $I^2C$ bus is possible. This is achieved by setting the STO flag while the STA flag is still set. No STOP condition is transmitted. The SIO1 hardware behaves as if a STOP condition was received and is able to transmit a START condition. The STO flag is cleared by hardware (see Figure 22).

$I^2C$ Bus Obstructed by a Low Level on SCL or SDA

An $I^2C$ bus hang-up occurs if SDA or SCL is pulled LOW by an uncontrolled source. If the SCL line is obstructed (pulled LOW) by a device on the bus, no further serial transfer is possible, and the SIO1 hardware cannot resolve this type of problem. When this occurs, the problem must be resolved by the device that is pulling the SCL bus line LOW.

If the SDA line is obstructed by another device on the bus (e.g., a slave device out of bit synchronization), the problem can be solved by transmitting additional clock pulses on the SCL line (see Figure 23). The SIO1 hardware transmits additional clock pulses when the STA flag is set, but no START condition can be generated because the SDA line is pulled LOW while the $I^2C$ bus is considered free. The SIO1 hardware attempts to generate a START condition after every two additional clock pulses on the SCL line. When the SDA line is eventually released, a normal START condition is transmitted, state 08H is entered, and the serial transfer continues.

If a forced bus access occurs or a repeated START condition is transmitted while SDA is obstructed (pulled LOW), the SIO1 hardware performs the same action as described above. In each case, state 08H is entered after a successful START condition is transmitted and normal serial transfer continues. Note that the CPU is not involved in solving these bus hang-up problems.

Bus Error

A bus error occurs when a START or STOP condition is present at an illegal position in the format frame. Examples of illegal positions are during the serial transfer of an address byte, a data or an acknowledge bit.

The SIO1 hardware only reacts to a bus error when it is involved in a serial transfer either as a master or an addressed slave. When a bus error is detected, SIO1 immediately switches to the not addressed slave mode, releases the SDA and SCL lines, sets the interrupt flag, and loads the status register with 00H. This status code may be used to vector to a service routine which either attempts the aborted serial transfer again or simply recovers from the error condition as shown in Table 7.

**Table 7. Miscellaneous States**

| STATUS CODE (S1STA) | STATUS OF THE I²C BUS AND SIO1 HARDWARE | APPLICATION SOFTWARE RESPONSE | | | | | NEXT ACTION TAKEN BY SIO1 HARDWARE |
|---|---|---|---|---|---|---|---|
| | | TO/FROM S1DAT | TO S1CON | | | | |
| | | | STA | STO | SI | AA | |
| F8H | No relevant state information available; SI = 0 | No S1DAT action | No S1CON action | | | | Wait or proceed current transfer |
| 00H | Bus error during MST or selected slave modes, due to an illegal START or STOP condition. State 00H can also occur when interference causes SIO1 to enter an undefined state. | No S1DAT action | 0 | 1 | 0 | X | Only the internal hardware is affected in the MST or addressed SLV modes. In all cases, the bus is released and SIO1 is switched to the not addressed SLV mode. STO is reset. |

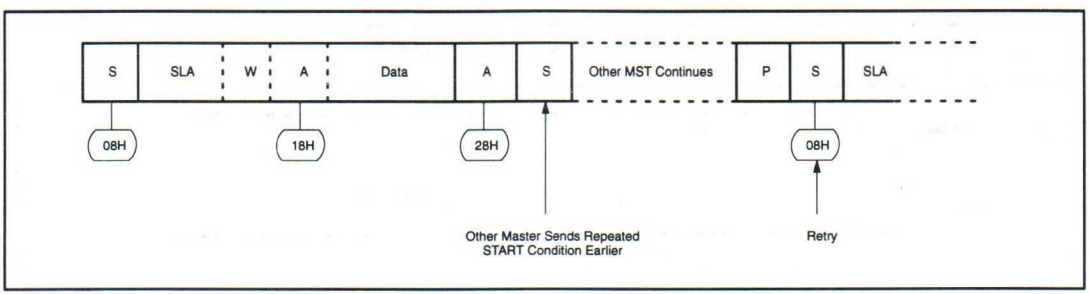



**Figure 21. Simultaneous Repeated START Conditions from 2 Masters**

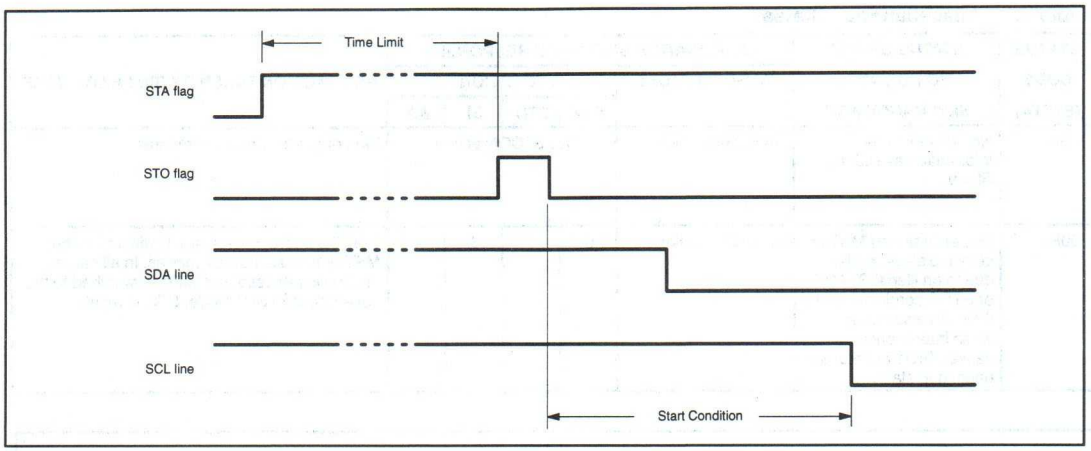
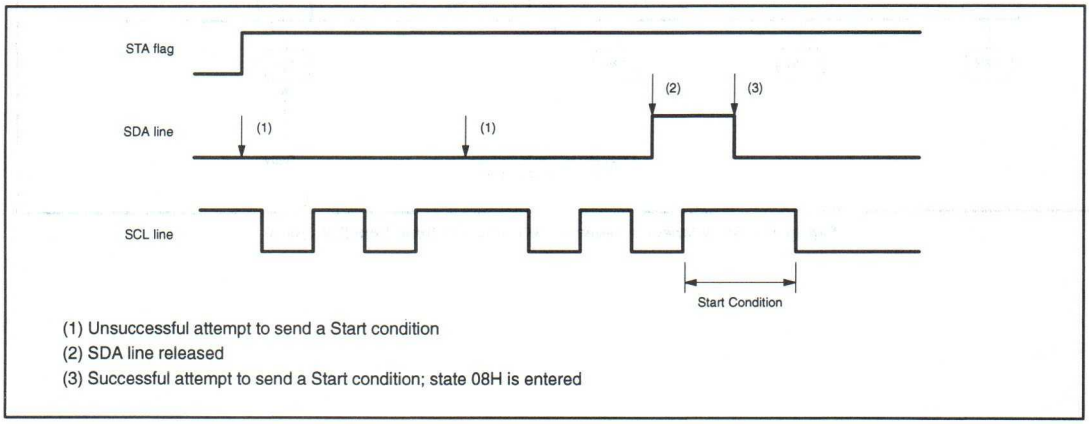



**Figure 22. Forced Access to a Busy I²C Bus**




(1) Unsuccessful attempt to send a Start condition
(2) SDA line released
(3) Successful attempt to send a Start condition; state 08H is entered

**Figure 23. Recovering from a Bus Obstruction Caused by a Low Level on SDA**

**Software Examples of SIO1 Service Routines:** This section consists of a software example for:

- Initialization of SIO1 after a RESET
- Entering the SIO1 interrupt routine
- The 26 state service routines for the
  - Master transmitter mode
  - Master receiver mode
  - Slave receiver mode
  - Slave transmitter mode

INITIALIZATION

In the initialization routine, SIO1 is enabled for both master and slave modes. For each mode, a number of bytes of internal data RAM are allocated to the SIO to act as either a transmission or reception buffer. In this example, 8 bytes of internal data RAM are reserved for different purposes. The data memory map is shown in Figure 24. The initialization routine performs the following functions:

- S1ADR is loaded with the part's own slave address and the general call bit (GC)
- P1.6 and P1.7 bit latches are loaded with logic 1s
- RAM location HADD is loaded with the high-order address byte of the service routines
- The SIO1 interrupt enable and interrupt priority bits are set
- The slave mode is enabled by simultaneously setting the ENS1 and AA bits in S1CON and the serial clock frequency (for master modes) is defined by loading CR0 and CR1 in S1CON. The master routines must be started in the main program.

The SIO1 hardware now begins checking the $I^2C$ bus for its own slave address and general call. If the general call or the own slave address is detected, an interrupt is requested and S1STA is loaded with the appropriate state information. The following text describes a fast method of branching to the appropriate service routine.

SIO1 INTERRUPT ROUTINE

When the SIO1 interrupt is entered, the PSW is first pushed on the stack. Then S1STA and HADD (loaded with the high-order address byte of the 26 service routines by the initialization routine) are pushed on to the stack. S1STA contains a status code which is the lower byte of one of the 26 service routines. The next instruction is RET, which is the return from subroutine instruction. When this instruction is executed, the high and low order address bytes are popped from stack and loaded into the program counter.

The next instruction to be executed is the first instruction of the state service routine. Seven bytes of program code (which execute in eight machine cycles) are required to branch to one of the 26 state service routines.

```
SI      PUSH  PSW     Save PSW
        PUSH  S1STA   Push status code
                      (low order address byte)
        PUSH  HADD    Push high order address byte
        RET           Jump to state service routine
```

The state service routines are located in a 256-byte page of program memory. The location of this page is defined in the initialization routine. The page can be located anywhere in program memory by loading data RAM register HADD with the page number. Page 01 is chosen in this example, and the service routines are located between addresses 0100H and 01FFH.

THE STATE SERVICE ROUTINES

The state service routines are located 8 bytes from each other. Eight bytes of code are sufficient for most of the service routines. A few of the routines require more than 8 bytes and have to jump to other locations to obtain more bytes of code. Each state routine is part of the SIO1 interrupt routine and handles one of the 26 states. It ends with a RETI instruction which causes a return to the main program.

MASTER TRANSMITTER AND MASTER RECEIVER MODES

The master mode is entered in the main program. To enter the master transmitter mode, the main program must first load the internal data RAM with the slave address, data bytes, and the number of data bytes to be transmitted. To enter the master receiver mode, the main program must first load the internal data RAM with the slave address and the number of data bytes to be received. The R/$\overline{W}$ bit determines whether SIO1 operates in the master transmitter or master receiver mode.

Master mode operation commences when the STA bit in S1CION is set by the SETB instruction and data transfer is controlled by the master state service routines in accordance with Table 3, Table 4, Figure 17, and Figure 18. In the example below, 4 bytes are transferred. There is no repeated START condition. In the event of lost arbitration, the transfer is restarted when the bus becomes free. If a bus error occurs, the $I^2C$ bus is released and SIO1 enters the not selected slave receiver mode. If a slave device returns a not acknowledge, a STOP condition is generated.

A repeated START condition can be included in the serial transfer if the STA flag is set instead of the STO flag in the state service routines vectored to by status codes 28H and 58H. Additional software must be written to determine which data is transferred after a repeated START condition.

SLAVE TRANSMITTER AND SLAVE RECEIVER MODES

After initialization, SIO1 continually tests the $I^2C$ bus and branches to one of the slave state service routines if it detects its own slave address or the general call address (see Table 5, Table 6, Figure 19, and Figure 20). If arbitration was lost while in the master mode, the master mode is restarted after the current transfer. If a bus error occurs, the $I^2C$ bus is released and SIO1 enters the not selected slave receiver mode.

In the slave receiver mode, a maximum of 8 received data bytes can be stored in the internal data RAM. A maximum of 8 bytes ensures that other RAM locations are not overwritten if a master sends more bytes. If more than 8 bytes are transmitted, a not acknowledge is returned, and SIO1 enters the not addressed slave receiver mode. A maximum of one received data byte can be stored in the internal data RAM after a general call address is detected. If more than one byte is transmitted, a not acknowledge is returned and SIO1 enters the not addressed slave receiver mode.

In the slave transmitter mode, data to be transmitted is obtained from the same locations in the internal data RAM that were previously loaded by the main program. After a not acknowledge has been returned by a master receiver device, SIO1 enters the not addressed slave mode.

ADAPTING THE SOFTWARE FOR DIFFERENT APPLICATIONS

The following software example shows the typical structure of the interrupt routine including the 26 state service routines and may be used as a base for user applications. If one or more of the four modes are not used, the associated state service routines may be removed but, care should be taken that a deleted routine can never be invoked.

This example does not include any time-out routines. In the slave modes, time-out routines are not very useful since, in these modes, SIO1 behaves essentially as a passive device. In the master modes, an internal timer may be used to cause a time-out if a serial transfer is not complete after a defined period of time. This time period is defined by the system connected to the $I^2C$ bus.

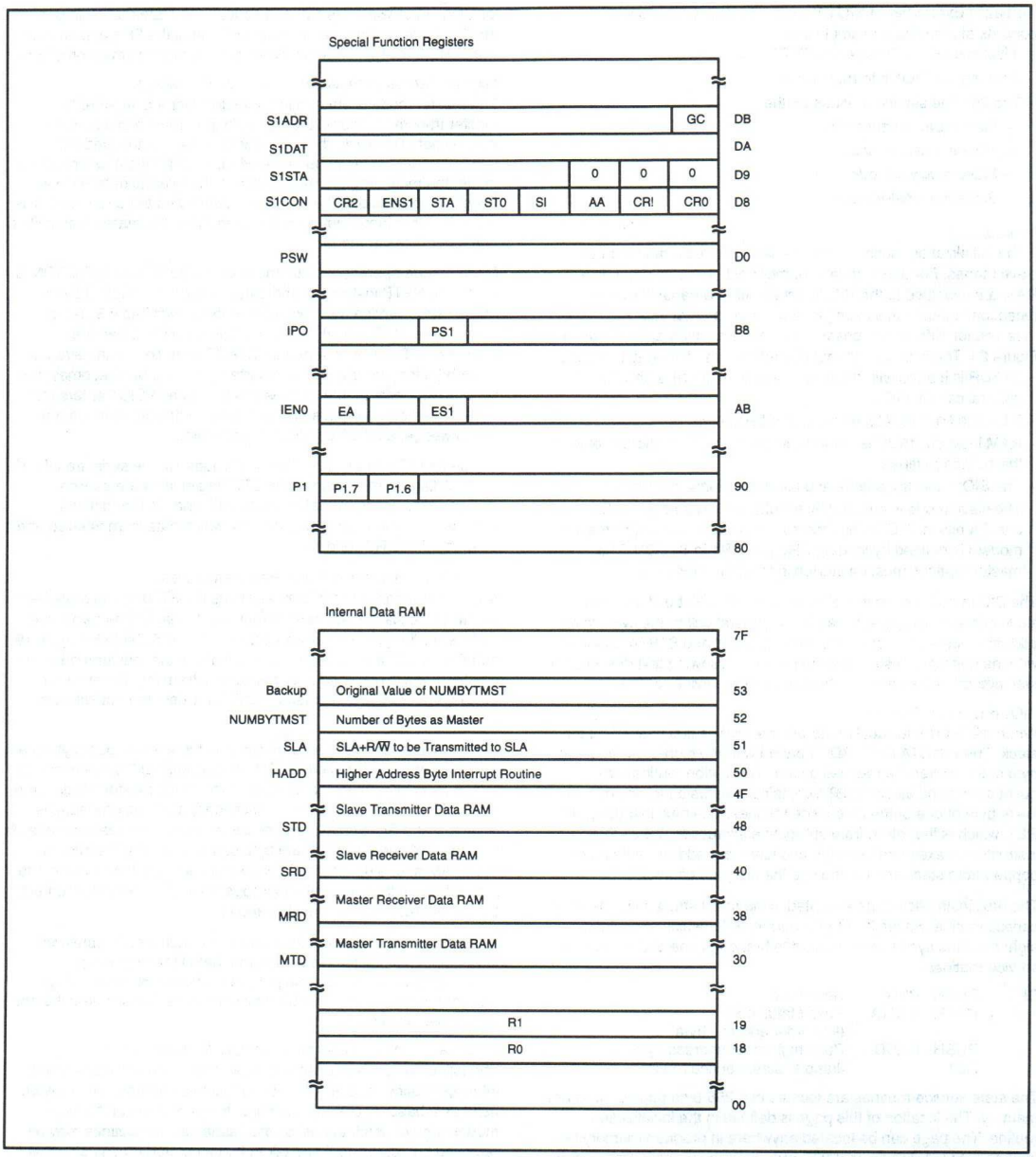



**Figure 24. SIO1 Data Memory Map**

```
                    !*****************************************************************************
                    ! SI01 EQUATE LIST
                    !*****************************************************************************
                    !*****************************************************************************
                    ! LOCATIONS OF THE SI01 SPECIAL FUNCTION REGISTERS
                    !*****************************************************************************
00D8                S1CON     –0xd8
00D9                S1STA     –0xd9
00DA                S1DAT     –0xda
00DB                S1ADR     –0xdb

00A8                IEN0      –0xa8
00B8                IP0       –02b8

                    !*****************************************************************************
                    ! BIT LOCATIONS
                    !*****************************************************************************

00DD                STA       –0xdd                    ! STA bit in S1CON
00BD                SI01HP    –0xbd                    ! IP0, SI01 Priority bit

                    !*****************************************************************************
                    ! IMMEDIATE DATA TO WRITE INTO REGISTER S1CON
                    !*****************************************************************************
00D5                ENS1_NOTSTA_STO_NOTSI_AA_CR0       –0xd5    ! Generates STOP
                                                                ! (CR0 = 100kHz)
00C5                ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0    –0xc5    ! Releases BUS and
                                                                ! ACK
00C1                ENS1_NOTSTA_NOTSTO_NOTSI_NOTAA_CR0 –0xc1    ! Releases BUS and
                                                                ! NOT ACK
00E5                ENS1_STA_NOTSTO_NOTSI_AA_CR0       –0xe5    ! Releases BUS and
                                                                ! set STA

                    !*****************************************************************************
                    ! GENERAL IMMEDIATE DATA
                    !*****************************************************************************
0031                OWNSLA    –0x31                    ! Own SLA+General Call
                                                       ! must be written into S1ADR
00A0                ENSI01    –0xa0                    ! EA+ES1, enable SIO1 interrupt
                                                       ! must be written into IEN0
0001                PAG1      –0x01                    ! select PAG1 as HADD
00C0                SLAW      –0xc0                    ! SLA+W to be transmitted
00C1                SLAR      –0xc1                    ! SLA+R to be transmitted
0018                SELRB3    –0x18                    ! Select Register Bank 3

                    !*****************************************************************************
                    ! LOCATIONS IN DATA RAM
                    !*****************************************************************************
0030                MTD       –0x30                    ! MST/TRX/DATA base address
0038                MRD       –0x38                    ! MST/REC/DATA base address
0040                SRD       –0x40                    ! SLV/REC/DATA base address
0048                STD       –0x48                    ! SLV/TRX/DATA base address

0053                BACKUP        –0x53                ! Backup from NUMBYTMST
                                                       ! To restore NUMBYTMST in case
                                                       ! of an Arbitration Loss.
0052                NUMBYTMST     –0x52                ! Number of bytes to transmit
                                                       ! or receive as MST.
0051                SLA           –0x51                ! Contains SLA+R/W to be
                                                       ! transmitted.
0050                HADD          –0x50                ! High Address byte for STATE 0
                                                       ! till STATE 25.
```

```
                        !*************************************************************************************
                        ! INITIALIZATION ROUTINE
                        ! Example to initialize IIC Interface as slave receiver or slave transmitter and
                        ! start a MASTER TRANSMIT or a MASTER RECEIVE function. 4 bytes will be transmitted or received.
                        !*************************************************************************************
                        .sect   strt
                        .base   0x00
0000    4100                            ajmp    INIT                    ! RESET

                        .sect   initial
                        .base   0x200
0200    75DB31          INIT:           mov     S1ADR,#OWNSLA           ! Load own SLA + enable
                                                                        ! general call recognition
0203    D296                            setb    P1(6)                   ! P1.6 High level.
0205    D297                            setb    P1(7)                   ! P1.7 High level.
0207    755001                          mov     HADD,#PAG1
020A    43A8A0                          orl     IEN0,#ENSI01            ! Enable SI01 interrupt
020D    C2BD                            clr     SI01HP                  ! SI01 interrupt low priority
020F    75D8C5                          mov     S1CON, #ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                        ! Initialize SLV funct.

                        !*************************************************************************************

                        !-------------------------------------------------------
                        ! START MASTER TRANSMIT FUNCTION
                        !-------------------------------------------------------

0212    755204                          mov     NUMBYTMST,#0x4          ! Transmit 4 bytes.
0215    7551C0                          mov     SLA,#SLAW               ! SLA+W, Transmit funct.
0218    D2DD                            setb    STA                     ! set STA in S1CON

                        !-------------------------------------------------------
                        ! START MASTER RECEIVE FUNCTION
                        !-------------------------------------------------------
021A    755204                          mov     NUMBYTMST,#0x4          ! Receive 4 bytes.
021D    7551C1                          mov     SLA,#SLAR               ! SLA+R, Receive funct.
0220    D2DD                            setb    STA                     ! set STA in S1CON


                        !*************************************************************************************
                        ! SI01 INTERRUPT ROUTINE
                        !*************************************************************************************
                        .sect   intvec                                  ! SI01 interrupt vector
                        .base   0x00

                        ! S1STA and HADD are pushed onto the stack.
                        ! They serve as return address for the RET instruction.
                        ! The RET instruction sets the Program Counter to address HADD,
                        ! S1STA and jumps to the right subroutine.

002B    C0D0                            push    psw                     ! save psw
002D    C0D9                            push    S1STA
002F    C050                            push    HADD
0031    22                              ret                             ! JMP to address HADD,S1STA.

                        !-------------------------------------------------------
                        ! STATE   : 00, Bus error.
                        ! ACTION  : Enter not addressed SLV mode and release bus. STO reset.
                        !-------------------------------------------------------
                        .sect   st0
                        .base   0x100

0100    75D8D5                          mov     S1CON,#ENS1_NOTSTA_STO_NOTSI_AA_CR0 ! clr SI
                                                                                    ! set STO,AA
0103    D0D0                            pop     psw
0105    32                              reti
```

```
                        !*************************************************************************
                        !*************************************************************************
                        ! MASTER STATE SERVICE ROUTINES
                        !*************************************************************************
                        ! State 08 and State 10 are both for MST/TRX and MST/REC.
                        ! The R/W bit decides whether the next state is within
                        ! MST/TRX mode or within MST/REC mode.
                        !*************************************************************************

                        !-------------------------------------------------------
                        ! STATE    : 08, A, START condition has been transmitted.
                        ! ACTION   : SLA+R/W are transmitted, ACK bit is received.
                        !-------------------------------------------------------
                        .sect      mts8
                        .base      0x108

0108    8551DA                          mov   S1DAT,SLA                    ! Load SLA+R/W
010B    75D8C5                          mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                           ! clr SI
010E    01A0                            ajmp  INITBASE1

                        !-------------------------------------------------------
                        ! STATE    : 10, A repeated START condition has been
                        !              transmitted.
                        ! ACTION   : SLA+R/W are transmitted, ACK bit is received.
                        !-------------------------------------------------------
                        .sect      mts10
                        .base      0x110

0110    8551DA                          mov   S1DAT,SLA                    ! Load SLA+R/W
0113    75D8C5                          mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                           ! clr SI
010E    01A0                            ajmp  INITBASE1

                        .sect      ibase1
                        .base      0xa0
00A0    75D018          INITBASE1:      mov   psw,#SELRB3
00A3    7930                            mov   r1,#MTD
00A5    7838                            mov   r0,#MRD
00A7    855253                          mov   BACKUP,NUMBYTMST             ! Save initial value
00AA    D0D0                            pop   psw
00AC    32                              reti


                        !*************************************************************************
                        !*************************************************************************
                        ! MASTER TRANSMITTER STATE SERVICE ROUTINES
                        !*************************************************************************
                        !*************************************************************************

                        !-------------------------------------------------------
                        ! STATE    : 18, Previous state was STATE 8 or STATE 10, SLA+W have been transmitted,
                        !              ACK has been received.
                        ! ACTION   : First DATA is transmitted, ACK bit is received.
                        !-------------------------------------------------------
                        .sect      mts18
                        .base      0x118

0118    75D018                          mov   psw,#SELRB3
011B    87DA                            mov   S1DAT,@r1
011D    01B5                            ajmp  CON
```

```
                        !-------------------------------------------------------
                        ! STATE   : 20, SLA+W have been transmitted, NOT ACK has been received
                        ! ACTION  : Transmit STOP condition.
                        !-------------------------------------------------------
                        .sect     mts20
                        .base     0x120

0120  75D8D5                              mov   S1CON,#ENS1_NOTSTA_STO_NOTSI_AA_CR0
                                                                  ! set STO, clr SI
0123  D0D0                                pop   psw
0125  32                                  reti

                        !-------------------------------------------------------
                        ! STATE   : 28, DATA of S1DAT have been transmitted, ACK received.
                        ! ACTION  : If Transmitted DATA is last DATA then transmit a STOP condition,
                        !           else transmit next DATA.
                        !-------------------------------------------------------
                        .sect     mts28
                        .base     0x128

0128  D55285                              djnz  NUMBYTMST,NOTLDAT1        ! JMP if NOT last DATA
012B  75D8D5                              mov   S1CON,#ENS1_NOTSTA_STO_NOTSI_AA_CR0
                                                                  ! clr SI, set AA
012E  01B9                                ajmp  RETmt

                        .sect     mts28sb
                        .base     0x0b0
00B0  75D018            NOTLDAT1:         mov   psw,#SELRB3
00B3  87DA                                mov   S1DAT,@r1
00B5  75D8C5            CON:              mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                  ! clr SI, set AA
00B8  09                                  inc   r1
00B9  D0D0              RETmt     :       pop   psw
00BB  32                                  reti
                        !-------------------------------------------------------
                        ! STATE   : 30, DATA of S1DAT have been transmitted, NOT ACK received.
                        ! ACTION  : Transmit a STOP condition.
                        !-------------------------------------------------------
                        .sect     mts30
                        .base     0x130

0130  75D8D5                              mov   S1CON,#ENS1_NOTSTA_STO_NOTSI_AA_CR0
                                                                  ! set STO, clr SI
0133  D0D0                                pop   psw
0135  32                                  reti

                        !-------------------------------------------------------
                        ! STATE   : 38, Arbitration lost in SLA+W or DATA.
                        ! ACTION  : Bus is released, not addressed SLV mode is entered.
                        !           A new START condition is transmitted when the IIC bus is free again.
                        !-------------------------------------------------------
                        .sect     mts38
                        .base     0x138

0138  75D8E5                              mov   S1CON,#ENS1_STA_NOTSTO_NOTSI_AA_CR0
013B  855352                              mov   NUMBYTMST,BACKUP
013E  01B9                                ajmp  RETmt
```

```
                        !**********************************************************************
                        !**********************************************************************
                        ! MASTER RECEIVER STATE SERVICE ROUTINES
                        !**********************************************************************
                        !**********************************************************************

                        !---------------------------------------------------
                        ! STATE    : 40, Previous state was STATE 08 or STATE 10,
                        !            SLA+R have been transmitted, ACK received.
                        ! ACTION   : DATA will be received, ACK returned.
                        !---------------------------------------------------
                        .sect      mts40
                        .base      0x140

0140  75D8C5                              mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                  ! clr STA, STO, SI set AA
0143  D0D0                                pop   psw
      32                                  reti

                        !---------------------------------------------------
                        ! STATE    : 48, SLA+R have been transmitted, NOT ACK received.
                        ! ACTION   : STOP condition will be generated.
                        !---------------------------------------------------
                        .sect      mts48
                        .base      0x148

0148  75D8D5            STOP:             mov   S1CON,#ENS1_NOTSTA_STO_NOTSI_AA_CR0
                                                                  ! set STO, clr SI
014B  D0D0                                pop   psw
014D  32                                  reti

                        !---------------------------------------------------
                        ! STATE    : 50, DATA have been received, ACK returned.
                        ! ACTION   : Read DATA of S1DAT.
                        !            DATA will be received, if it is last DATA
                        !            then NOT ACK will be returned else ACK will be returned.
                        !---------------------------------------------------
                        .sect      mrs50
                        .base      0x150

0150  75D018                              mov   psw,#SELRB3
0153  A6DA                                mov   @r0,S1DAT                 ! Read received DATA
0155  01C0                                ajmp  REC1

                        .sect      mrs50s
                        .base      0xc0

00C0  D55205            REC1:             djnz  NUMBYTMST,NOTLDAT2
00C3  75D8C1                              mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_NOTAA_CR0
                                                                  ! clr SI,AA
00C6  8003                                sjmp  RETmr
00C8  75D8C5            NOTLDAT2:         mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                  ! clr SI, set AA
00CB  08                RETmr:            inc   r0
00CC  D0D0                                pop   psw
00CE  32                                  reti

                        !---------------------------------------------------
                        ! STATE    : 58, DATA have been received, NOT ACK returned.
                        ! ACTION   : Read DATA of S1DAT and generate a STOP condition.
                        !---------------------------------------------------
                        .sect      mrs58
                        .base      0x158

0158  75D018                              mov   psw,#SELRB3
015B  A6DA                                mov   @R0,S1DAT
015D  80E9                                sjmp  STOP
```

```
                        !*****************************************************************************
                        !*****************************************************************************
                        ! SLAVE RECEIVER STATE SERVICE ROUTINES
                        !*****************************************************************************
                        !*****************************************************************************

                        !-----------------------------------------------------------
                        ! STATE   : 60, Own SLA+W have been received, ACK returned.
                        ! ACTION  : DATA will be received and ACK returned.
                        !-----------------------------------------------------------
                        .sect     srs60
                        .base     0x160
0160  75D8C5                      mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                  ! clr SI, set AA
0163  75D018                      mov   psw,#SELRB3
0166  01D0                        ajmp  INITSRD
                        .sect     insrd
                        .base     0xd0

00D0  7840              INITSRD:  mov   r0,#SRD
00D2  7908                        mov   r1,#8
00D4  D0D0                        pop   psw
00D6  32                          reti

                        !-----------------------------------------------------------
                        ! STATE   : 68, Arbitration lost in SLA and R/W as MST
                        !           Own SLA+W have been received, ACK returned
                        ! ACTION  : DATA will be received and ACK returned.
                        !           STA is set to restart MST mode after the bus is free again.
                        !-----------------------------------------------------------
                        .sect     srs68
                        .base     0x168
0168  75D8E5                      mov   S1CON,#ENS1_STA_NOTSTO_NOTSI_AA_CR0
016B  75D018                      mov   psw,#SELRB3
016E  01D0                        ajmp  INITSRD

                        !-----------------------------------------------------------
                        ! STATE   : 70, General call has been received, ACK returned.
                        ! ACTION  : DATA will be received and ACK returned.
                        !-----------------------------------------------------------
                        .sect     srs70
                        .base     0x170
0170  75D8C5                      mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                  ! clr SI, set AA
0173  75D018                      mov   psw,#SELRB3               ! Initialize SRD counter
0176  01D0                        ajmp  initsrd

                        !-----------------------------------------------------------
                        ! STATE   : 78, Arbitration lost in SLA+R/W as MST.
                        !           General call has been received, ACK returned.
                        ! ACTION  : DATA will be received and ACK returned.
                        !           STA is set to restart MST mode after the bus is free again.
                        !-----------------------------------------------------------
                        .sect     srs78
                        .base     0x178
0178  75D8E5                      mov   S1CON,#ENS1_STA_NOTSTO_NOTSI_AA_CR0
017B  75D018                      mov   psw,#SELRB3               ! Initialize SRD counter
017E  01D0                        ajmp  INITSRD
```

```
                        !-------------------------------------------------------
                        ! STATE   : 80, Previously addressed with own SLA. DATA received, ACK returned.
                        ! ACTION  : Read DATA.
                        !          IF received DATA was the last
                        !          THEN superfluous DATA will be received  and NOT ACK returned
                                   ELSE next DATA will be received and ACK returned.
                        !-------------------------------------------------------
                        .sect     srs80
                        .base     0x180

0180  75D018                           mov   psw,#SELRB3
0183  A6DA                             mov   @r0,S1DAT                  ! Read received DATA
0185  01D8                             ajmp  REC2

                        .sect     srs80s
                        .base     0xd8

00D8  D906              REC2:          djnz  r1,NOTLDAT3
00DA  75D8C1            LDAT:          mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_NOTAA_CR0
                                                                  ! clr  SI,AA
00DD  D0D0                             pop   psw
00DF  32                               reti
00E0  75D8C5            NOTLDAT3:      mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                  ! clr  SI, set  AA
00E3  08                               inc   r0
00E4  D0D0              RETsr:         pop   psw
00E6  32                               reti

                        !-------------------------------------------------------
                        ! STATE   : 88, Previously addressed with own SLA. DATA received NOT ACK returned.
                        ! ACTION  : No save of DATA, Enter NOT addressed SLV mode.
                        !          Recognition of own SLA. General call recognized, if S1ADR. 0–1.
                        !-------------------------------------------------------
                        .sect     srs88
                        .base     0x188

0188  75D8C5                           mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                  ! clr  SI, set  AA
018B  01E4                             ajmp  RETsr

                        !-------------------------------------------------------
                        ! STATE   : 90, Previously addressed with general call.
                        !          DATA has been received, ACK has been returned.
                        ! ACTION  : Read DATA.
                                   After General call only one byte will be received with ACK
                        !          the second DATA will be received with NOT ACK.
                        !          DATA will be received and NOT ACK returned.
                        !-------------------------------------------------------
                        .sect     srs90
                        .base     0x190

0190  75D018                           mov   psw,#SELRB3
0193  A6DA                             mov   @r0,S1DAT                  ! Read received DATA
0195  01DA                             ajmp  LDAT

                        !-------------------------------------------------------
                        ! STATE   : 98, Previously addressed with general call.
                        !          DATA has been received, NOT ACK has been returned.
                        ! ACTION  : No save of DATA, Enter NOT addressed SLV mode.
                                   Recognition of own SLA. General call recognized, if S1ADR. 0–1.
                        !-------------------------------------------------------
                        .sect     srs98
                        .base     0x198

0198  75D8C5                           mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                  ! clr  SI, set  AA
019B  D0D0                             pop   psw
019D  32                               reti
```

```
                        !-------------------------------------------------
                        ! STATE    : A0, A STOP condition or repeated START has been received,
                        !            while still addressed as SLV/REC or SLV/TRX.
                        ! ACTION   : No save of DATA, Enter NOT addressed SLV mode.
                        !            Recognition of own SLA. General call recognized, if S1ADR. 0–1.
                        !-------------------------------------------------
                        .sect     srsA0
                        .base     0x1a0

01A0  75D8C5                      mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                              ! clr SI, set AA
01A3  D0D0                        pop   psw
01A5  32                          reti


                        !*************************************************************************
                        !*************************************************************************
                        ! SLAVE TRANSMITTER STATE SERVICE ROUTINES
                        !*************************************************************************
                        !*************************************************************************

                        !-------------------------------------------------
                        ! STATE    : A8, Own SLA+R received, ACK returned.
                        ! ACTION   : DATA will be transmitted, A bit received.
                        !-------------------------------------------------
                        .sect     stsa8
                        .base     0x1a8

01A8  8548DA                      mov   S1DAT,STD                 ! load DATA in S1DAT
01AB  75D8C5                      mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                              ! clr SI, set AA
01AE  01E8                        ajmp  INITBASE2

                        .sect     ibase2
                        .base     0xe8
00E8  75D018            INITBASE2:mov   psw,#SELRB3
00EB  7948                        mov   r1, #STD
00ED  09                          inc   r1
00EE  D0D0                        pop   psw
00F0  32                          reti

                        !-------------------------------------------------
                        ! STATE    : B0, Arbitration lost in SLA and R/W as MST.  Own SLA+R received, ACK returned.
                        ! ACTION   : DATA will be transmitted, A bit received.
                        !            STA is set to restart MST mode after the bus is free again.
                        !-------------------------------------------------
                        .sect     stsb0
                        .base     0x1b0

01B0  8548DA                      mov   S1DAT,STD                 ! load DATA in S1DAT
01B3  75D8E5                      mov   S1CON,#ENS1_STA_NOTSTO_NOTSI_AA_CR0
01B6  01E8                        ajmp  INITBASE2
```

```
                        !-------------------------------------------------
                        ! STATE   : B8, DATA has been transmitted, ACK received.
                        ! ACTION  : DATA will be transmitted, ACK bit is received.
                        !-------------------------------------------------
                        .sect     stsb8
                        .base     0x1b8
01B8  75D018                              mov   psw,#SELRB3
01BB  87DA                                mov   S1DAT,@r1
01BD  01F8                                ajmp  SCON

                        .sect     scn
                        .base     0xf8

00F8  75D8C5            SCON:             mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                    ! clr SI, set AA
00FB  09                                  inc   r1
00FC  D0D0                                pop   psw
00FE  32                                  reti
                        !-------------------------------------------------
                        ! STATE   : C0, DATA has been transmitted, NOT ACK received.
                        ! ACTION  : Enter not addressed SLV mode.
                        !-------------------------------------------------
                        .sect     stsc0
                        .base     0x1c0
01C0  75D8C5                              mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                    ! clr SI, set AA
01C3  D0D0                                pop   psw
01C5  32                                  reti

                        !-------------------------------------------------
                        ! STATE   : C8, Last DATA has been transmitted (AA=0), ACK received.
                        ! ACTION  : Enter not addressed SLV mode.
                        !-------------------------------------------------
                        .sect     stsc8
                        .base     0x1c8
01C8  75D8C5                              mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                    ! clr SI, set AA
01CB  D0D0                                pop   psw
01CD  32                                  reti


                        !*************************************************
                        !*************************************************
                        ! END OF SI01 INTERRUPT ROUTINE
                        !*************************************************
                        !*************************************************
```

**Reset Circuitry**

The reset circuitry for the 8XC552 is connected to the reset pin RST. A Schmitt trigger is used at the input for noise rejection (see Figure 25). The output of the Schmitt trigger is sampled by the reset circuitry every machine cycle.

A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods) while the oscillator is running. The CPU responds by executing an internal reset. During reset, ALE and PSEN output a HIGH level. In order to perform a correct reset, this level must not be affected by external elements. The RST line can also be pulled HIGH internally by a pull-up transistor activated by the watchdog timer T3. The length of the output pulse from T3 is 3 machine cycles. A pulse of such short duration is necessary in order to recover from a processor or system fault as fast as possible.

Note that the short reset pulse from Timer T3 cannot discharge the power-on reset capacitor (see Figure 26). Consequently, when the watchdog timer is also used to set external devices, this capacitor arrangement should not be connected to the RST pin, and a different circuit should be used to perform the power-on reset operation. A timer T3 overflow, if enabled, will force a reset condition to the 8XC552 by an internal connection, whether the output RST is tied LOW or not.

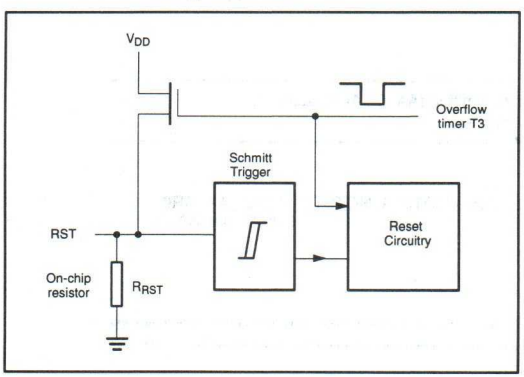



**Figure 25. On-Chip Reset Configuration**

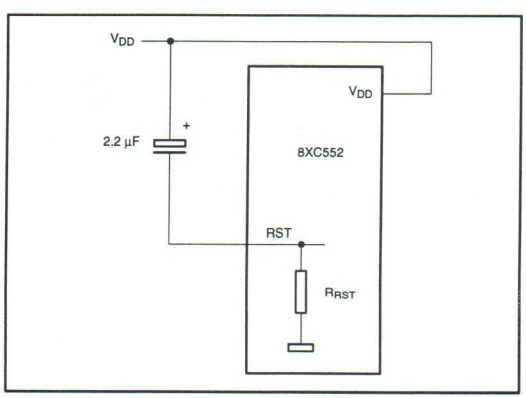



**Figure 26. Power-On Reset**

The internal reset is executed during the second cycle in which RST is HIGH and is repeated every cycle until RST goes low. It leaves the internal registers as follows:

| RESGISTER | CONTENT |
|---|---|
| ACC | 0000 0000 |
| ADCON | xx00 0000 |
| ADCH | xxxx xxxx |
| B | 0000 0000 |
| CML0-CML2 | 0000 0000 |
| CMH0-CMH2 | 0000 0000 |
| CTCON | 0000 0000 |
| CTL0-CTL3 | xxxx xxxx |
| CTH0-CTH3 | xxxx xxxx |
| DPL | 0000 0000 |
| DPH | 0000 0000 |
| IEN0 | 0000 0000 |
| IEN1 | 0000 0000 |
| IP0 | 0000 0000 |
| IP1 | 0000 0000 |
| PCH | 0000 0000 |
| PCL | 0000 0000 |
| PCON | 0xx0 0000 |
| PSW | 0000 0000 |
| PWM0 | 0000 0000 |
| PWM1 | 0000 0000 |
| PWMP | 0000 0000 |
| P0-P4 | 1111 1111 |
| PS | xxxx xxxx |
| RTE | 0000 0000 |
| S0BUF | xxxx xxxx |
| S0CON | 0000 0000 |
| S1ADR | 0000 0000 |
| S1CON | 0000 0000 |
| S1DAT | 0000 0000 |
| S1STA | 1111 1000 |
| SP | 0000 0111 |
| STE | 1100 0000 |
| TCON | 0000 0000 |
| TH0, TH1 | 0000 0000 |
| TMH2 | 0000 0000 |
| TL0, TL1 | 0000 0000 |
| TML2 | 0000 0000 |
| TMOD | 0000 0000 |
| TM2CON | 0000 0000 |
| TM2IR | 0000 0000 |
| T3 | 0000 0000 |

The internal RAM is not affected by reset. At power-on, the RAM content is indeterminate.

**Interrupts**

The 8XC552 has fifteen interrupt sources, each of which can be assigned one of two priority levels, as shown in Figure 27. The five interrupt sources common to the 80C51 are the external interrupts ($\overline{INT0}$ and $\overline{INT1}$), the timer 0 and timer 1 interrupts (IT0 and IT1), and the serial I/O interrupt (RI or TI). In the 8XC552, the standard serial interrupt is called SIO0. Since the subsystems which create these interrupts are identical on both parts, their functionality is likewise identical. The only differences are the locations of the enable and priority register configurations and the priority structure. This is detailed below along with the specifics of the interrupts unique to the 8XC552.

The eight Timer T2 interrupts are generated by flags CTI0-CT13, CMI0-CMI2, and by the logical OR of flags T2OV and T2BO. Flags CTI0 to CT13 are set by input signals CT0I to CT3i. Flags CMI0 to CMI2 are set when a match occurs between Timer T2 and the compare registers CM0, CM1, and CM2. When an 8-bit or 16-bit overflow occurs, flags T2BO and T2OV are set, respectively. These nine flags are not cleared by hardware and must be reset by software to avoid recurring interrupts.

The ADC interrupt is generated by the ADCI flag in the ADC control register (ADCON). This flag is set when an ADC conversion result is ready to be read. ADCI is not cleared by hardware and must be reset by software to avoid recurring interrupts.

The SIO1 ($I^2C$) interrupt is generated by the SI flag in the SIO1 control register (S1CON). This flag is set when S1STA is loaded with a valid status code.

The ADCI flag may be reset by software. It cannot be set by software. All other flags that generate interrupts may be set or cleared by software, and the effect is the same as setting or resetting the flags by hardware. Thus, interrupts may be generated by software and pending interrupts can be canceled by software.

**Interrupt Enable Registers:** Each interrupt source can be individually enabled or disabled by setting or clearing a bit in the interrupt enable special function registers IEN0 and IEN1. All interrupt sources can also be globally enabled or disabled by setting or clearing bit EA in IEN0. The interrupt enable registers are described in Figures 28 and 29.

**Interrupt Priority Structure:** Each interrupt source can be assigned one of two priority levels. Interrupt priority levels are defined by the interrupt priority special function registers IP0 and IP1. IP0 and IP1 are described in Figures 30 and 31.

Interrupt priority levels are as follows:
"0"—low priority
"1"—high priority

A low priority interrupt may be interrupted by a high priority interrupt. A high priority interrupt cannot be interrupted by any other interrupt source. If two requests of different priority occur simultaneously, the high priority level request is serviced. If requests of the same priority are received simultaneously, an internal polling sequence determines which request is serviced. Thus, within each priority level, there is a second priority structure determined by the polling sequence. This second priority structure is shown in Table 8.

The above Priority Within Level structure is only used when there are simultaneous requests of the same priority level.

**Interrupt Handling:** The interrupt sources are sampled at S5P2 of every machine cycle. The samples are polled during the following machine cycle. If one of the flags was in a set condition at S5P2 of the previous machine cycle, the polling cycle will find it and the interrupt system will generate an LCALL to the appropriate service routine, provided this hardware-generated LCALL is not blocked by any of the following conditions:

1. An interrupt of higher or equal priority level is already in progress.
2. The current machine cycle is not the final cycle in the execution of the instruction in progress. (No interrupt request will be serviced until the instruction in progress is completed.)
3. The instruction in progress is RETI or any access to the interrupt priority or interrupt enable registers. (No interrupt will be serviced after RETI or after a read or write to IP0, IP1, IE0, or IE1 until at least one other instruction has been subsequently executed.)

The polling cycle is repeated with every machine cycle, and the values polled are the values present at S5P2 of the previous machine cycle. Note that if an interrupt flag is active but is not being responded to because of one of the above conditions, and if the flag is inactive when the blocking condition is removed, then the blocked interrupt will not be serviced. Thus, the fact that the interrupt flag was once active but not serviced is not remembered. Every polling cycle is new.

The processor acknowledges an interrupt request by executing a hardware-generated LCALL to the appropriate service routine. In some cases it also clears the flag which generated the interrupt, and in others it does not. It clears the Timer 0, Timer 1, and external interrupt flags. An external interrupt flag (IE0 or IE1) is cleared only if it was transition-activated. All other interrupt flags are not cleared by hardware and must be cleared by the software. The LCALL pushes the contents of the program counter on to the stack (but it does not save the PSW) and reloads the PC with an address that depends on the source of the interrupt being vectored to as shown in Table 9.

Execution proceeds from the vector address until the RETI instruction is encountered. The RETI instruction clears the "priority level active" flip-flop that was set when this interrupt was acknowledged. It then pops the top two bytes from the stack and reloads the program counter. Execution of the interrupted program continues from where it was interrupted.

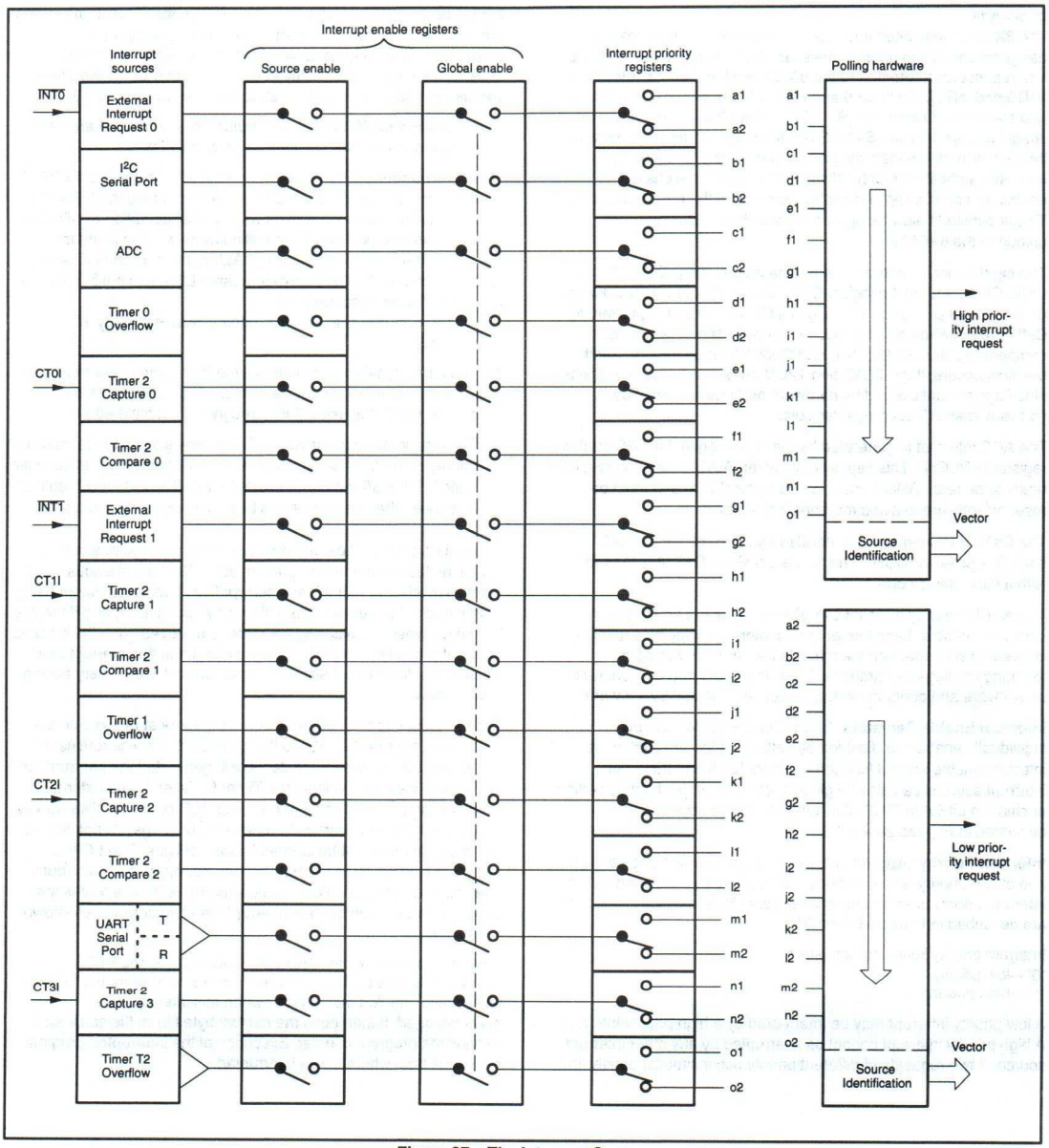



**Figure 27. The Interrupt System**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **IEN0 (A8H)** | EA | EAD | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 |
| | (MSB) | | | | | | | (LSB) |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IEN0.7 | EA | Global enable/disable control<br>0 = No interrupt is enabled<br>1 = Any individually enabled interrupt will be accepted |
| IEN0.6 | EAD | Eanble ADC interrupt |
| IEN0.5 | ES1 | Enable SIO1 ($I^2C$) interrupt |
| IEN0.4 | ES0 | Enable SIO0 (UART) interrupt |
| IEN0.3 | ET1 | Enable Timer 1 interrupt |
| IEN0.2 | EX1 | Enable External interrupt 1 |
| IEN0.1 | ET0 | Enable Timer 0 interrupt |
| IEN0.0 | EX0 | Enable External interrupt 0 |

SU00762

**Figure 28. Interrupt Enable Register (IEN0)**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **IEN1 (E8H)** | ET2 | ECM2 | ECM1 | ECM0 | ECT3 | ECT2 | ECT1 | ECT0 |
| | (MSB) | | | | | | | (LSB) |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IEN1.7 | ET2 | Enable Timer T2 overflow interrupt(s) |
| IEN1.6 | ECM2 | Enable T2 Comparator 2 interrupt |
| IEN1.5 | ECM1 | Enable T2 Comparator 1 interrupt |
| IEN1.4 | ECM0 | Enable T2 Comparator 0 interrupt |
| IEN1.3 | ECT3 | Enable T2 Capture register 3 interrupt |
| IEN1.2 | ECT2 | Enable T2 Capture register 2 interrupt |
| IEN1.1 | ECT1 | Enable T2 Capture register 1 interrupt |
| IEN1.0 | ECT0 | Enable T2 Capture register 0 interrupt |

SU00755

In all cases, if the enable bit is 0, then the interrupt is disabled, and if the enable bit is 1, then the interrupt is enabled.

**Figure 29. Interrupt Enable Register (IEN1)**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **IP0 (B8H)** | – | PAD | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 |
| | (MSB) | | | | | | | (LSB) |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IP0.7 | – | Unused |
| IP0.6 | PAD | ADC interrupt priority level |
| IP0.5 | PS1 | SIO1 ($I^2C$) interrupt priority level |
| IP0.4 | PS0 | SIO0 (UART) interrupt priority level |
| IP0.3 | PT1 | Timer 1 interrupt priority level |
| IP0.2 | PX1 | External interrupt 1 priority level |
| IP0.1 | PT0 | Timer 0 interrupt priority level |
| IP0.0 | PX0 | External interrupt 0 priority level |

SU00763

**Figure 30. Interrupt Priority Register (IP0)**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| IP1 (F8H) | PT2 | PCM2 | PCM1 | PCM0 | PCT3 | PCT2 | PCT1 | PCT0 |
| | (MSB) | | | | | | | (LSB) |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IP1.7 | PT2 | T2 overflow interrupt(s) priority level |
| IP1.6 | PCM2 | T2 comparator 2 interrupt priority level |
| IP1.5 | PCM1 | T2 comparator 1 interrupt priority level |
| IP1.4 | PCM0 | T2 comparator 0 interrupt priority level |
| IP1.3 | PCT3 | T2 capture register 3 interrupt priority level |
| IP1.2 | PCT2 | T2 capture register 2 interrupt priority level |
| IP1.1 | PCT1 | T2 capture register 1 interrupt priority level |
| IP1.0 | PCT0 | T2 capture register 0 interrupt priority level |

SU00764

**Figure 31. Interrupt Priority Register (IP1)**

**Table 8. Interrupt Priority Structure**

| SOURCE | NAME | PRIORITY WITHIN LEVEL |
|---|---|---|
| | | (highest) ↑ |
| External interrupt 0 | X0 | |
| SIO1 ($I^2C$) | S1 | |
| ADC completion | ADC | |
| Timer 0 overflow | T0 | |
| T2 capture 0 | CT0 | |
| T2 compare 0 | CM0 | |
| External interrupt 1 | X1 | |
| T2 capture 1 | CT1 | |
| T2 compare 1 | CM1 | |
| Timer 1 overflow | T1 | |
| T2 capture 2 | CT2 | |
| T2 compare 2 | CM2 | |
| SIO0 (UART) | S0 | |
| T2 capture 3 | CT3 | |
| Timer T2 overflow | T2 | ↓ (lowest) |

**Table 9. Interrupt Vector Addresses**

| SOURCE | NAME | VECTOR ADDRESS |
|---|---|---|
| External interrupt 0 | X0 | 0003H |
| Timer 0 overflow | T0 | 000BH |
| External interrupt 1 | X1 | 0013H |
| Timer 1 overflow | T1 | 001BH |
| SIO0 (UART) | S0 | 0023H |
| SIO1 ($I^2C$) | S1 | 002BH |
| T2 capture 0 | CT0 | 0033H |
| T2 capture 1 | CT1 | 003BH |
| T2 capture 2 | CT2 | 0043H |
| T2 capture 3 | CT3 | 004BH |
| ADC completion | ADC | 0053H |
| T2 compare 0 | CM0 | 005BH |
| T2 compare 1 | CM1 | 0063H |
| T2 compare 2 | CM2 | 006BH |
| T2 overflow | T2 | 0073H |

**I/O Port Structure**

The 8XC552 has six 8-bit ports. Each port consists of a latch (special function registers P0 to P5), an input buffer, and an output driver (port 0 to 4 only). Ports 0-3 are the same as in the 80C51, with the exception of the additional functions of port 1. The parallel I/O function of port 4 is equal to that of ports 1, 2, and 3. Port 5 may be used as an input port only.

Figure 32 shows the bit latch and I/O buffer functional diagrams of the unique 8XC552 ports. A bit latch corresponds to one bit in a port's SFR and is represented as a D type flip-flop. A "write to latch" signal from the CPU latches a bit from the internal bus and a "read latch" signal from the CPU places the Q output of the flip-flop on the internal bus. A "read pin" signal from the CPU places the actual port pin level on the internal bus. Some instructions that read a port read the actual port pin levels, and other instructions read the latch (SFR) contents.

**Port 1 Operation**

Port 1 operates the same as it does in the 8051 with the exception of port lines P1.6 and P1.7, which may be selected as the SCL and SDA lines of serial port SIO1 ($I^2C$). Because the $I^2C$ bus may be active while the device is disconnected from $V_{DD}$, these pins are provided with open drain drivers. Therefore pins P1.6 and P1.7 do not have internal pull-ups.

**Port 5 Operation**

Port 5 may be used to input up to 8 analog signals to the ADC. Unused ADC inputs may be used to input digital inputs. These inputs have an inherent hysteresis to prevent the input logic from drawing excessive current from the power lines when driven by analog signals. Channel to channel crosstalk (Ct) should be taken into consideration when both analog and digital signals are simultaneously input to Port 5 (see, D.C. characteristics in data sheet).

Port 5 is not bidirectional and may not be configured as an output port. All six ports are multifunctional, and their alternate functions are listed in Table 10. A more detailed description of these features can be found in the relevant parts of this section.

**Pulse Width Modulated Outputs**

The 8XC552 contains two pulse width modulated output channels (see Figure 33). These channels generate pulses of programmable length and interval. The repetition frequency is defined by an 8-bit prescaler PWMP, which supplies the clock for the counter. The prescaler and counter are common to both PWM channels. The 8-bit counter counts modulo 255, i.e., from 0 to 254 inclusive. The value of the 8-bit counter is compared to the contents of two registers: PWM0 and PWM1. Provided the contents of either of these registers is greater than the counter value, the corresponding $\overline{PWM0}$ or $\overline{PWM1}$ output is set LOW. If the contents of these registers are equal to, or less than the counter value, the output will be HIGH. The pulse-width-ratio is therefore defined by the contents of the registers PWM0 and PWM1. The pulse-width-ratio is in the range of 0 to 1 and may be programmed in increments of 1/255.

Buffered PWM outputs may be used to drive DC motors. The rotation speed of the motor would be proportional to the contents of PWMn. The PWM outputs may also be configured as a dual DAC. In this application, the PWM outputs must be integrated using conventional operational amplifier circuitry. If the resulting output voltages have to be accurate, external buffers with their own analog supply should be used to buffer the PWM outputs before they are integrated. The repetition frequency $f_{PWM}$, at the PWMn outputs is give by:

$$f_{PWM} = \frac{f_{OSC}}{2 \times (1 + PWMP) \times 255}$$

This gives a repetition frequency range of 123Hz to 31.4kHz ($f_{OSC}$ = 16MHz). At fosc = 24MHz, the frequency range is 184Hz to 47.1Hz. By loading the PWM registers with either 00H or FFH, the PWM channels will output a constant HIGH or LOW level, respectively. Since the 8-bit counter counts modulo 255, it can never actually reach the value of the PWM registers when they are loaded with FFH.

When a compare register (PWM0 or PWM1) is loaded with a new value, the associated output is updated immediately. It does not have to wait until the end of the current counter period. Both $\overline{PWMn}$ output pins are driven by push-pull drivers. These pins are not used for any other purpose.

Prescaler frequency control register PWMP

| PWMP (FEH) | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | MSB | | | | | | | LSB |

PWMP.0-7 Prescaler division factor = PWMP + 1.

Reading PWMP gives the current reload value. The actual count of the prescaler cannot be read.

| PWM0 (FCH)<br>PWM1 (FDH) | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | MSB | | | | | | | LSB |

PWM0/1.0-7} Low/high ratio of $\overline{PWMn} = \frac{(PWMn)}{255 - (PWMn)}$

**Analog-to-Digital Converter**

The analog input circuitry consists of an 8-input analog multiplexer and a 10-bit, straight binary, successive approximation ADC. The analog reference voltage and analog power supplies are connected via separate input pins. The conversion takes 50 machine cycles, i.e., 37.5µs at an oscillator frequency of 16MHz, 25µs at an oscillator frequency of 24MHz. Input voltage swing is from 0V to +5V. Because the internal DAC employs a ratiometric potentiometer, there are no discontinuities in the converter characteristic. Figure 34 shows a functional diagram of the analog input circuitry.

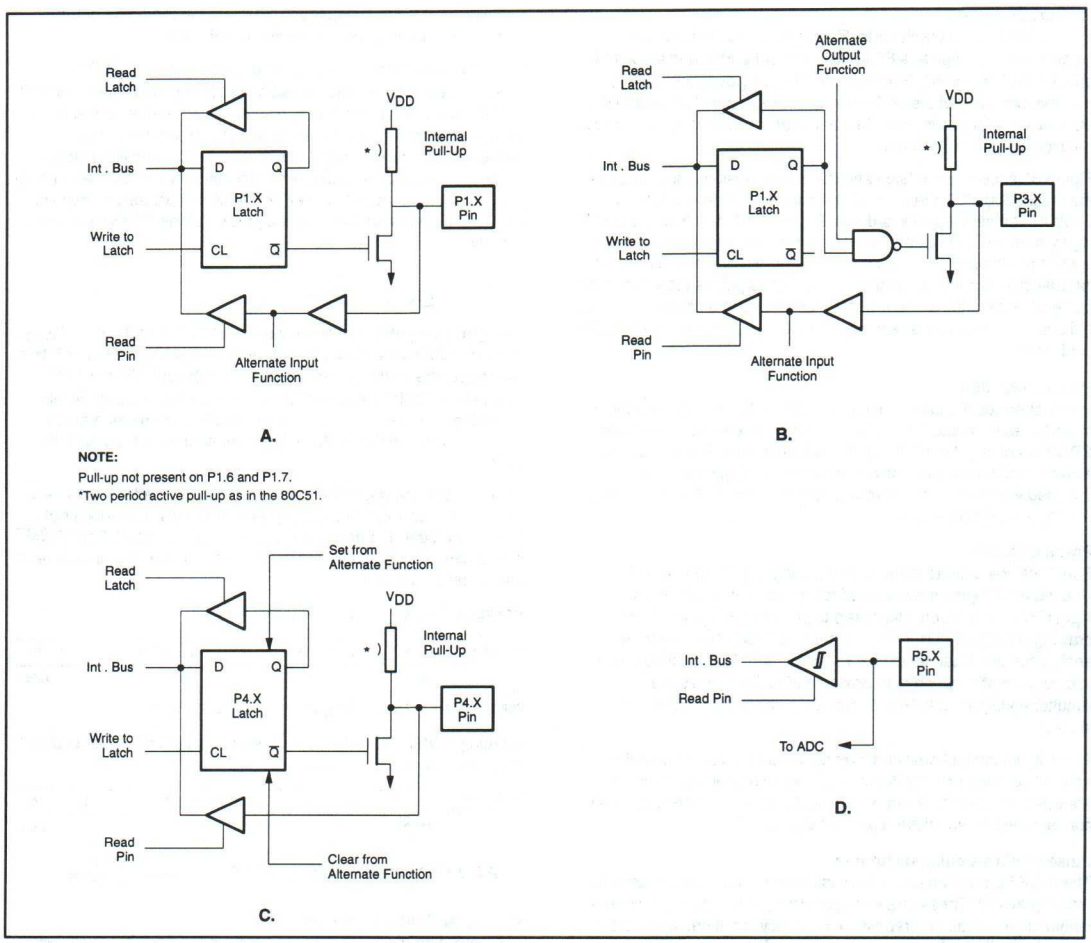



**Figure 32. Port Bit Latches and I/O Buffers**

**Table 10. Input/Output Ports**

| PORT PIN | ALTERNATE FUNCTION | |
|---|---|---|
| P0.0 | AD0 | Multiplexed lower order address/data bus used during external memory accesses |
| P0.1 | AD1 | |
| P0.2 | AD2 | |
| P0.3 | AD3 | |
| P0.4 | AD4 | |
| P0.5 | AD5 | |
| P0.6 | AD6 | |
| P0.7 | AD7 | |
| P1.0 | CT0I | Capture timiner input signals for timer T2 |
| P1.1 | CT1I | |
| P1.2 | CT2I | |
| P1.3 | CT3I | |
| P1.4 | T2 | T2 event input |
| P1.5 | RT2 | T2 timer reset signal. Rising edge triggered |
| P1.6 | SCL | Serial port clock line $I^2C$ bus |
| P1.7 | SDA | Serial port data line $I^2C$ bus |
| P2.0 | A8 | High order address byte used during external memory accesses |
| P2.1 | A9 | |
| P2.2 | A10 | |
| P2.3 | A11 | |
| P2.4 | A12 | |
| P2.5 | A13 | |
| P2.6 | A14 | |
| P2.7 | A15 | |
| P3.0 | RxD | Serial input port (UART) |
| P3.1 | TxD | Serial output port (UART) |
| P3.2 | $\overline{INT0}$ | External interrupt 0 |
| P3.3 | $\overline{INT1}$ | External interrupt 1 |
| P3.4 | T0 | Timer 0 external input |
| P3.5 | T1 | Timer 1 external input |
| P3.6 | $\overline{WR}$ | External data memory write strobe |
| P3.7 | $\overline{RD}$ | External data memory read strobe |
| P4.0 | CMSR0 | Timer T2: compare and set/reset outputs on a match with timer T2 |
| P4.1 | CMSR1 | |
| P4.2 | CMSR2 | |
| P4.3 | CMSR3 | |
| P4.4 | CMSR4 | |
| P4.5 | CMSR5 | Timer T2: compare and toggle outputs on a match with timer T2 |
| P4.6 | CMT0 | |
| P4.7 | CMT1 | |
| P5.0 | ADC0 | Eight analogue ADC inputs |
| P5.1 | ADC1 | |
| P5.2 | ADC2 | |
| P5.3 | ADC3 | |
| P5.4 | ADC4 | |
| P5.5 | ADC5 | |
| P5.6 | ADC6 | |
| P5.7 | ADC7 | |

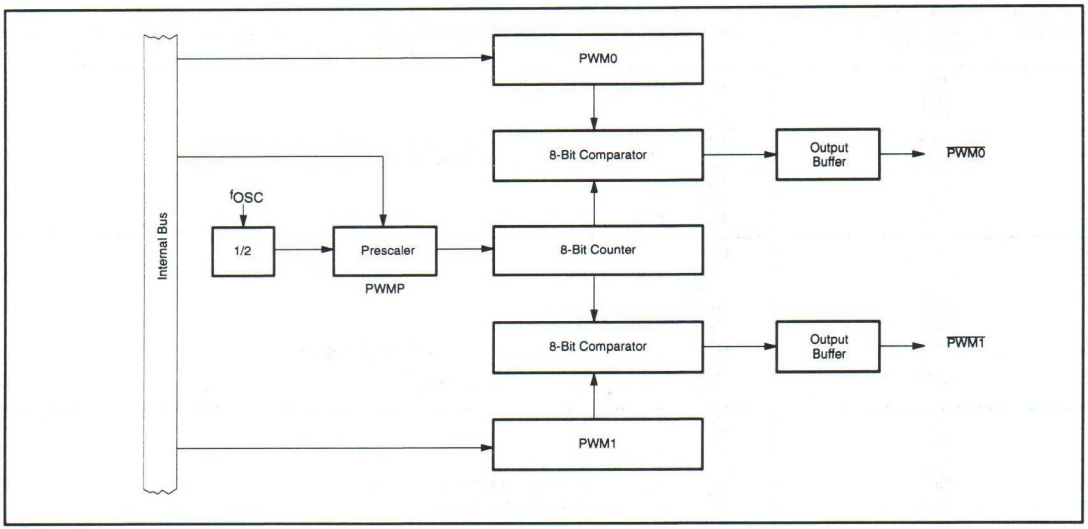



**Figure 33. Functional Diagram of Pulse Width Modulated Outputs**

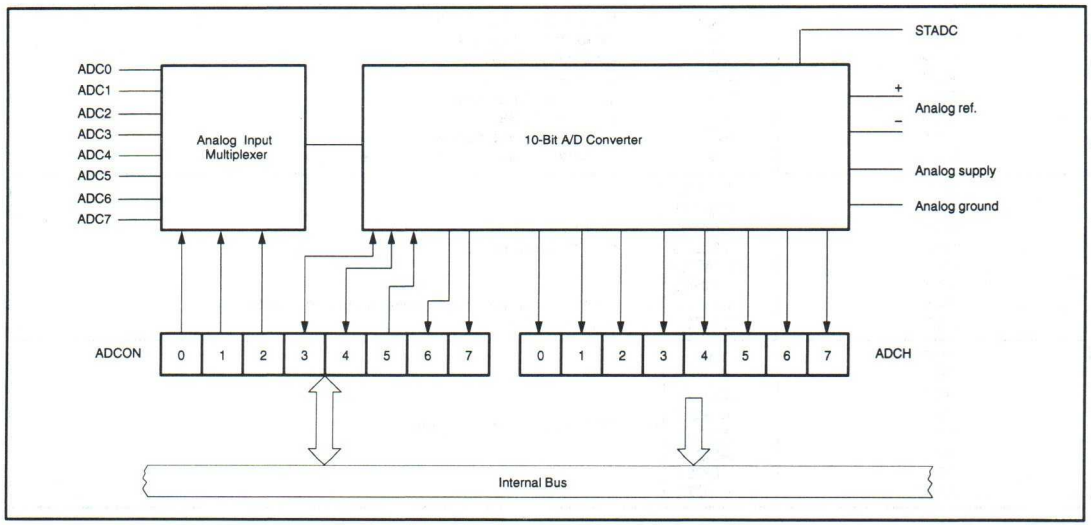



**Figure 34. Functional Diagram of Analog Input Circuitry**

**Analog-to-Digital Conversion:** Figure 35 shows the elements of a successive approximation (SA) ADC. The ADC contains a DAC which converts the contents of a successive approximation register to a voltage (VDAC) which is compared to the analog input voltage (Vin). The output of the comparator is fed to the successive approximation control logic which controls the successive approximation register. A conversion is initiated by setting ADCS in the ADCON register. ADCS can be set by software only or by either hardware or software.

The software only start mode is selected when control bit ADCON.5 (ADEX) = 0. A conversion is then started by setting control bit ADCON.3 (ADCS). The hardware or software start mode is selected when ADCON.5 = 1, and a conversion may be started by setting ADCON.3 as above or by applying a rising edge to external pin STADC. When a conversion is started by applying a rising edge, a low level must be applied to STADC for at least one machine cycle followed by a high level for at least one machine cycle.

The low-to-high transition of STADC is recognized at the end of a machine cycle, and the conversion commences at the beginning of the next cycle. When a conversion is initiated by software, the conversion starts at the beginning of the machine cycle which follows the instruction that sets ADCS. ADCS is actually implemented with two flip-flops: a command flip-flop which is affected by set operations, and a status flag which is accessed during read operations.

The next two machine cycles are used to initiate the converter. At the end of the first cycle, the ADCS status flag is set and a value of "1" will be returned if the ADCS flag is read while the conversion is in progress. Sampling of the analog input commences at the end of the second cycle.

During the next eight machine cycles, the voltage at the previously selected pin of port 5 is sampled, and this input voltage should be stable in order to obtain a useful sample. In any event, the input voltage slew rate must be less than 10V/ms in order to prevent an undefined result.

The successive approximation control logic first sets the most significant bit and clears all other bits in the successive approximation register (10 0000 0000B). The output of the DAC (50% full scale) is compared to the input voltage Vin. If the input voltage is greater than VDAC, then the bit remains set; otherwise it is cleared.

The successive approximation control logic now sets the next most significant bit (11 0000 0000B or 01 0000 0000B, depending on the previous result), and VDAC is compared to Vin again. If the input voltage is greater than VDAC, then the bit being tested remains set; otherwise the bit being tested is cleared. This process is repeated until all ten bits have been tested, at which stage the result of the conversion is held in the successive approximation register. Figure 36 shows a conversion flow chart. The bit pointer identifies the bit under test. The conversion takes four machine cycles per bit.

The end of the 10-bit conversion is flagged by control bit ADCON.4 (ADCI). The upper 8 bits of the result are held in special function register ADCH, and the two remaining bits are held in ADCON.7 (ADC.1) and ADCON.6 (ADC.0). The user may ignore the two least significant bits in ADCON and use the ADC as an 8-bit converter (8 upper bits in ADCH). In any event, the total actual conversion time is 50 machine cycles for the 8XC552 or 24 machine cycles for the 8XC562. ADCI will be set and the ADCS status flag will be reset 50 (or 24) cycles after the command flip-flop (ADCS) is set.

Control bits ADCON.0, ADCON.1, and ADCON.2 are used to control an analog multiplexer which selects one of eight analog channels (see Figure 37). An ADC conversion in progress is unaffected by an external or software ADC start. The result of a completed conversion remains unaffected provided ADCI = logic 1; a new ADC conversion already in progress is aborted when the idle or power-down mode is entered. The result of a completed conversion (ADCI = logic 1) remains unaffected when entering the idle mode.

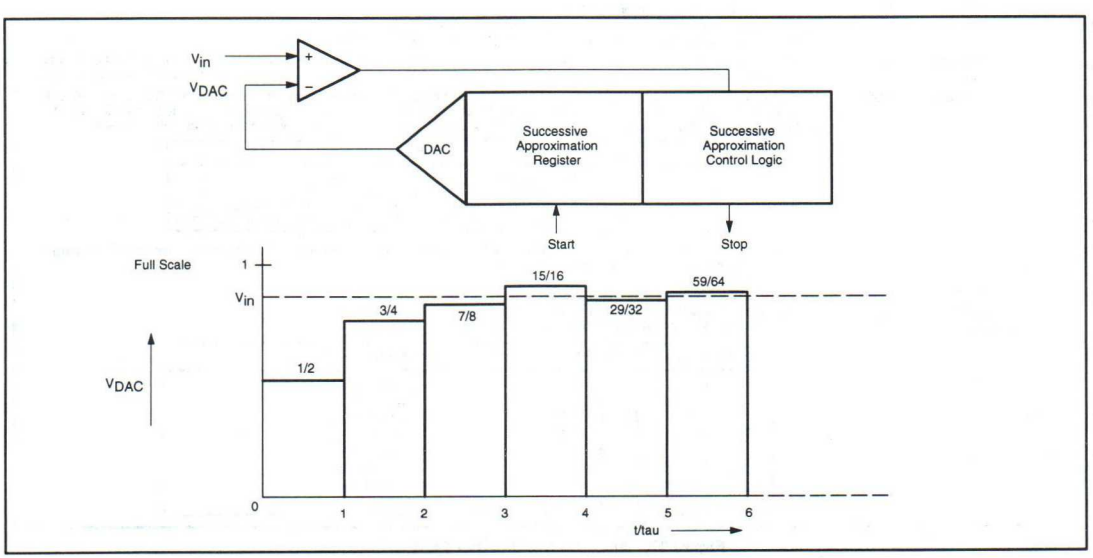



**Figure 35. Successive Approximation ADC**

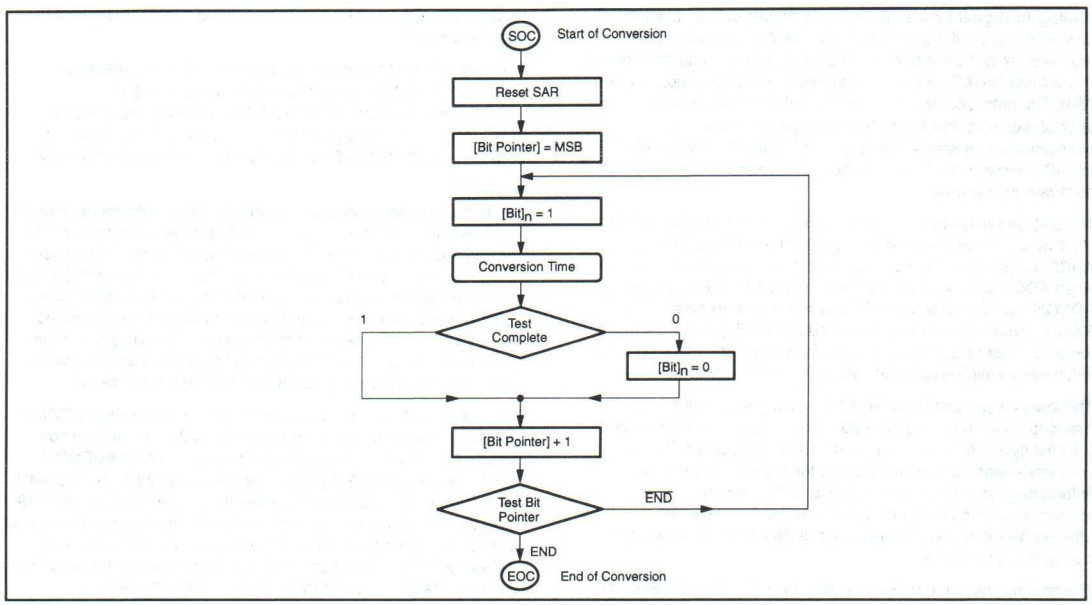



**Figure 36. A/D Conversion Flowchart**

ADCON (C5H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| ADC.1 | ADC.0 | ADEX | ADCI | ADCS | AADR2 | AADR1 | AADR0 |
| (MSB) | | | | | | | (LSB) |

| Bit | Symbol | Function |
|---|---|---|
| ADCON.7 | ADC.1 | Bit 1 of ADC result |
| ADCON.6 | ADC.0 | Bit 0 of ADC result |
| ADCON.5 | ADEX | Enable external start of conversion by STADC<br>0 = Conversion can be started by software only (by setting ADCS)<br>1 = Conversion can be started by software or externally (by a rising edge on STADC) |
| ADCON.4 | ADCI | ADC interrupt flag: this flag is set when an A/D conversion result is ready to be read. An interrupt is invoked if it is enabled. The flag may be cleared by the interrupt service routine. While this flag is set, the ADC cannot start a new conversion. ADCI cannot be set by software. |
| ADCON.3 | ADCS | ADC start and status: setting this bit starts an A/D conversion. It may be set by software or by the external signal STADC. The ADC logic ensures that this signal is HIGH while the ADC is busy. On completion of the conversion, ADCS is reset immediately after the interrupt flag has been set. ADCS cannot be reset by software. A new conversion may not be started while either ADCS or ADCI is high. |

| ADCI | ADCS | ADC Status |
|---|---|---|
| 0 | 0 | ADC not busy; a conversion can be started |
| 0 | 1 | ADC busy; start of a new conversion is blocked |
| 1 | 0 | Conversion completed; start of a new conversion requires ADCI=0 |
| 1 | 1 | Conversion completed; start of a new conversion requires ADCI=0 |

If ADCI is cleared by software while ADCS is set at the same time, a new A/D conversion with the same channel number may be started. But it is recommended to reset ADCI ***before*** ADCS is set.

| | | |
|---|---|---|
| ADCON.2<br>ADCON.1<br>ADCON.0 | AADR2<br>AADR1<br>AADR0 | Analogue input select: this binary coded address selects one of the eight analogue port bits of P5 to be input to the converter. It can only be changed when ADCI and ADCS are both LOW. |

| AADR2 | AADR1 | AADR0 | Selected Analog Channel |
|---|---|---|---|
| 0 | 0 | 0 | ADC0 (P5.0) |
| 0 | 0 | 1 | ADC1 (P5.1) |
| 0 | 1 | 0 | ADC2 (P5.2) |
| 0 | 1 | 1 | ADC3 (P5.3) |
| 1 | 0 | 0 | ADC4 (P5.4) |
| 1 | 0 | 1 | ADC5 (P5.5) |
| 1 | 1 | 0 | ADC6 (P5.6) |
| 1 | 1 | 1 | ADC7 (P5.7) |

**Figure 37. ADC Control Register (ADCON)**

**ADC Resolution and Analog Supply:** Figure 38 shows how the ADC is realized. The ADC has its own supply pins ($AV_{DD}$ and $AV_{SS}$) and two pins (Vref+ and Vref–) connected to each end of the DAC's resistance-ladder. The ladder has 1023 equally spaced taps, separated by a resistance of "R". The first tap is located 0.5 x R above Vref–, and the last tap is located 1.5 x R below Vref+. This gives a total ladder resistance of 1024 x R. This structure ensures that the DAC is monotonic and results in a symmetrical quantization error as shown in Figure 40.

For input voltages between Vref– and (Vref–) + 1/2 LSB, the 10-bit result of an A/D conversion will be 00 0000 0000B = 000H. For input voltages between (Vref+) – 3/2 LSB and Vref+, the result of a conversion will be 11 1111 1111B = 3FFH. AVref+ and AVref– may be between $AV_{DD}$ + 0.2V and $AV_{SS}$ – 0.2V. AVref+ should be positive with respect to AVref–, and the input voltage (Vin) should be between AVref+ and AVref–. If the analog input voltage range is from 2V to 4V, then 10-bit resolution can be obtained over this range if AVref+ = 4V and AVref– = 2V.

The result can always be calculated from the following formula:

$$\text{Result} = 1024 \times \frac{V_{IN} - AV_{ref-}}{AV_{ref+} - AV_{ref-}}$$

**Power Reduction Modes**

The 8XC552 has two reduced power modes of operation: the idle mode and the power-down mode. These modes are entered by setting bits in the PCON special function register. When the 8XC552 enters the idle mode, the following functions are disabled:

| | |
|---|---|
| CPU | (halted) |
| Timer T2 | (halted and reset) |
| PWM0, PWM1 | (reset; outputs are high) |
| ADC | (conversion aborted if in progress). |

In idle mode, the following functions remain active:

Timer 0
Timer 1
Timer T3
SIO0 SIO1
External interrupts

When the 8XC552 enters the power-down mode, the oscillator is stopped. The power-down mode is entered by setting the PD bit in the PCON register. The PD bit can only be set if the $\overline{EW}$ input is tied HIGH.

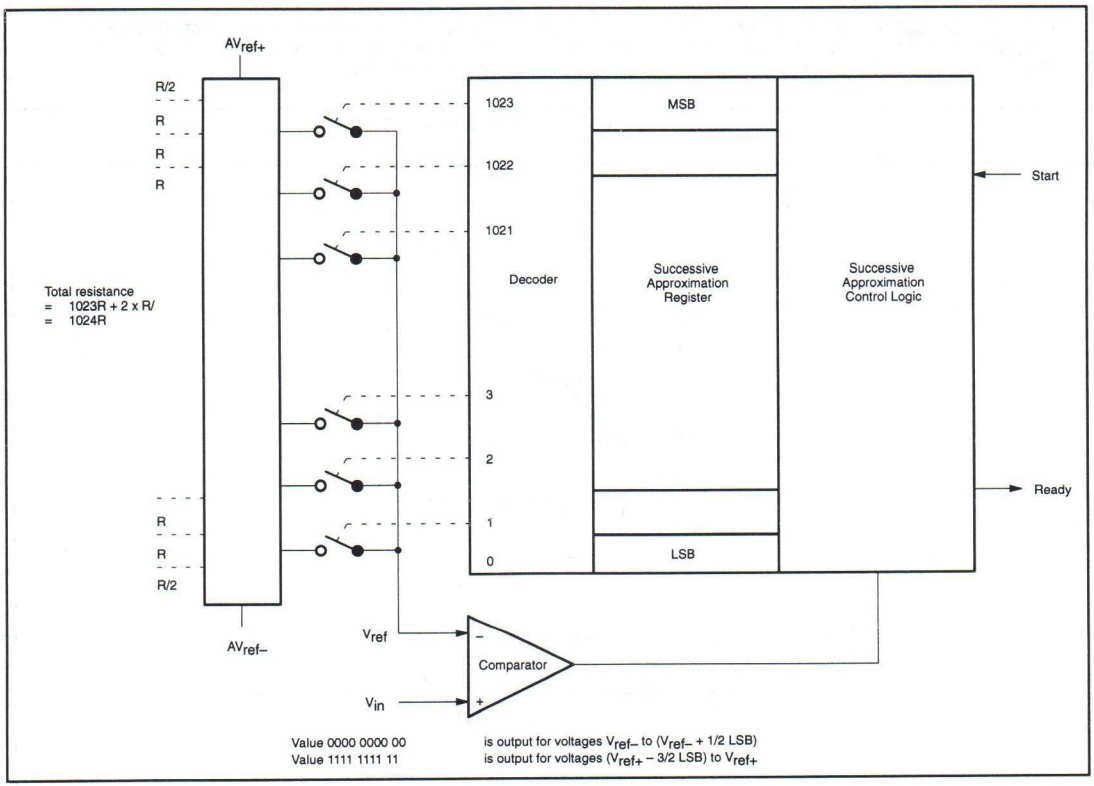
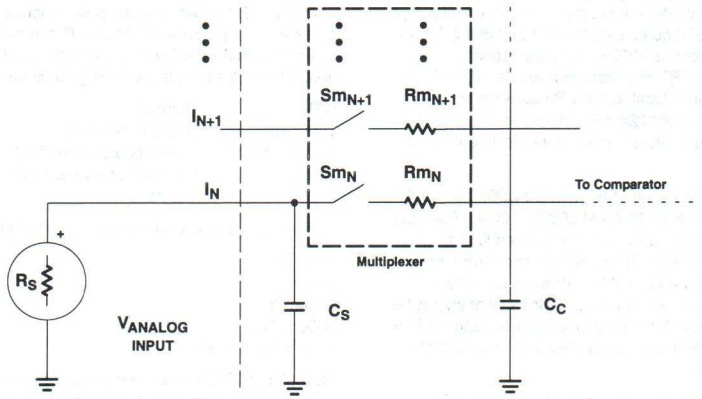



**Figure 38. ADC Realization**




Rm = 0.5 - 3 kohms
CS + CC = 15pF maximum
RS = Recommended < 9.6 kohms for 1 LSB @ 12MHz

**NOTE:**
Because the analog to digital converter has a sampled-data comparator, the input looks capacitive to a source. When a conversion is initiated, switch Sm closes for 8tcy (8µs @ 12MHz crystal frequency) during which time capacitance Cs + Cc is charged. It should be noted that the sampling causes the analog input to present a varying load to an analog source.

**Figure 39. A/D Input: Equivalent Circuit**

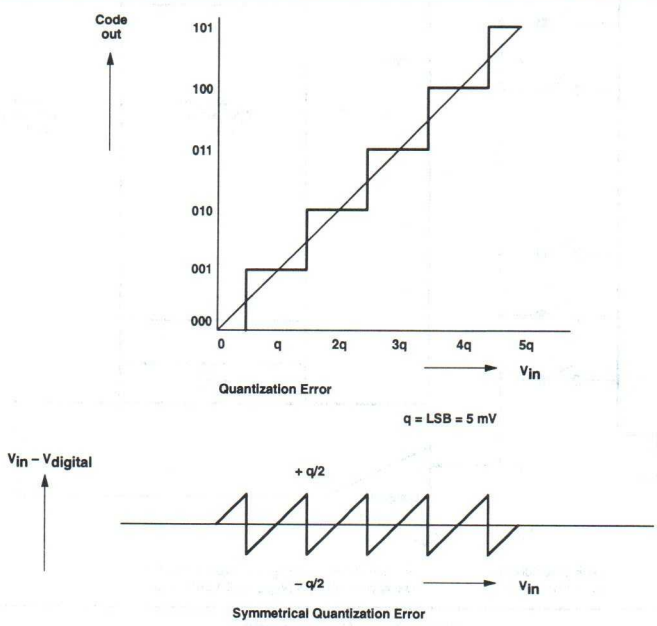



**Figure 40. Effective Conversion Characteristic**

**Power-Down Mode:** The instruction that sets PCON.1 will be the last instruction executed in the normal operating mode before the power-down mode is entered. In the power-down mode, the on-chip oscillator is stopped. This freezes all functions; only the on-chip RAM and special function registers are held. The port pins output the contents of their respective special function registers. A hardware reset is the only way to terminate the power-down mode. Reset re-defines all the special function registers, but does not change the on-chip RAM.

In the power-down mode, $V_{DD}$ and $AV_{DD}$ can be reduced to minimize power consumption. $V_{DD}$ and $AV_{DD}$ must not be reduced before the power-down mode is entered and must be restored to the normal operating voltage before the power-down mode is terminated. The reset that terminates the power-down mode also freezes the oscillator. The reset should not be activated before $V_{DD}$ and $AV_{DD}$ are restored to their normal operating level, and must be held active long enough to allow the oscillator to restart and stabilize (normally less than 10ms).

The status of the external pins during power-down is shown in Table 11. If the power-down mode is entered while the 8XC552 is executing out of external program memory, the port data that is held in the P2 special function register is restored to port 2. If a port latch contains a "1", the port pin is held HIGH during the power-down mode by the strong pull-up transistor.

**Power Control Register PCON:** The idle and power-down modes are entered by writing to bits in PCON. PCON is not bit addressable. See Figure 41.

**Memory Organization**

The memory organization of the 8XC552 is the same as in the 80C51, with the exception that the 8XC552 has 8k ROM, 256 bytes RAM, and additional SFRs. Addressing modes are the same in the 8XC552 and the 80C51. Details of the differences are given in the following paragraphs.

In the 8XC552, the lower 8k of the 64k program memory address space is filled by internal ROM. By tying the $\overline{EA}$ pin high, the processor fetches instructions from internal program ROM. Bus expansion for accessing program memory from 8k upwards is automatic since external instruction fetches occur automatically when the program counter exceeds 8191. If the $\overline{EA}$ pin is tied low, all program memory fetches are from external memory. The execution speed of the 8XC552 is the same regardless of whether fetches are from external or internal program memory. If all storage is on-chip, then byte location 8191 should be left vacant to prevent an undesired pre-fetch from external program memory address 8192.

Certain locations in program memory are reserved for specific programs. Locations 0000H to 0002H are reserved for the initialization program. Following reset, the CPU always begins execution at locations 0000H. Locations 0003H to 0075H are reserved for the fifteen interrupt request service routines.

Functionally, the internal data memory is the most flexible of the address spaces. The internal data memory space is subdivided into a 256-byte internal data RAM address space and a 128-byte special function register (SFR) address space, as shown in Figure 42.

The internal data RAM address space is 0 to 255. Four 8-bit register banks occupy locations 0 to 31. 128 bit locations of the internal data RAM are accessible through direct addressing. These bits reside in 16 bytes of internal data RAM at locations 20H to 2FH. The stack can be located anywhere in the internal data RAM address space by loading the 8-bit stack pointer. The stack depth may be 256 bytes maximum.

The SFR address space is 128 to 255. All registers except the program counter and the four 8-bit register banks reside in this address space. Memory mapping the SFRs allows them to be accessed as easily as internal RAM, and as such, they can be operated on by most instructions. The 56 SFRs are listed in Figure 43, and their mapping in the SFR address space is shown in Figures 44 and 45. RAM bit addresses are the same as in the 80C51 and are summarized in Figure 46. The special function bit addresses are summarized in Figure 47.

**Table 11. External Pin Status During Idle and Power-Down Modes**

| MODE | MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 | PORT 4 | $\overline{PWM0}$/$\overline{PWM1}$ |
|---|---|---|---|---|---|---|---|---|---|
| Idle (1) | Internal | 1 | 1 | Port data | Port data | Port data | Port data | Port data | HIGH |
| Idle (1) | External | 1 | 1 | Floating | Port data | Address | Port data | Port data | HIGH |
| Power-down | Internal | 0 | 0 | Port data | Port data | Port data | Port data | Port data | HIGH |
| Power-down | External | 0 | 0 | Floating | Port data | Port data | Port data | Port data | HIGH |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PCON (87H) | SMOD | – | – | WLE | GF1 | GF0 | PD | IDL |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| PCON.7 | SMOD | Double Baud rate bit. When set to logic 1 the baud rate is doubled when the serial port SIO0 is being used in modes 1, 2, or 3. |
| PCON.6 | – | (Reserved) |
| PCON.5 | – | (Reserved) |
| PCON.4 | WLE | Watchdog load enable. This flag must be set by software prior to loading timer T3 (watch-dog timer). It is cleared when timer T3 is loaded. |
| PCON.3 | GF1 | General-purpose flag bit |
| PCON.2 | GF0 | General-purpose flag bit |
| PCON.1 | PD | Power-down bit. Setting this bit activates the power-down mode. It can only be set if input $\overline{EW}$ is high. |
| PCON.0 | IDL | Idle mode bit. Setting this bit activates the idle mode. |

If logic 1s are written to PD and IDL at the same time, PD takes precedence. The reset value of PCON is (0XX00000).

**Figure 41. Power Control Register (PCON)**

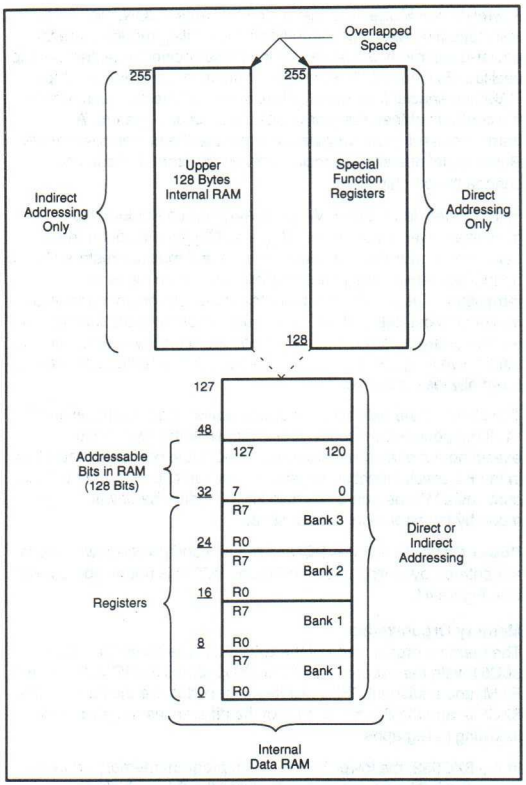



**Figure 42. Internal Data Memory Address Space**

ARITHMETIC REGISTERS:
ACCumulator,* B register,*
Program Status Word*

POINTERS:
Stack Pointer,
Data Pointer (High and Low)

PARALLEL I/O PORTS:
Port 5,* Port 4,*Port 3,*
Port 2,* Port 1,* Port 0*

INTERRUPT SYSTEM:
Interrupt Priority 0,*
Interrupt Priority 1,*
Interrupt Enable 0,*
Interrupt Enable 1*

PULSE WIDTH MODULATED O/Ps:
Pulse Width Modulation Prescaler
Pulse Width Modulation Register 0,
Pulse Width Modulation Register 1

SERIAL I/O PORTS:
Serial 0 CONtrol,* Serial 0 data BUFfer,
Serial 1 CONtrol,* Serial 1 DATa,
Serial 1 STAtus, Serial 1 ADDress, PCON

TIMERS:
Timer MODe, Timer CONtrol,*
Timer Low 0, Timer High 0,
Timer Low 1, Timer High 1,
TiMer T2 CONtrol, TiMer Low 2,
Timer High 2, Timer T3

CAPTURE AND COMPARE LOGIC:
CapTure CONtrol,
TiMer T2 Interrupt flag Register,
CapTure Low 0, CapTure High 0,
CapTure Low 1, CapTure High 1,
CapTure Low 2, CapTure High 2,
CapTure Low 3, CapTure High 3,
CoMpare Low 0, CoMpare High 0,
CoMpare Low 1, CoMpare High 1,
CoMpare Low 2, CoMpare High 2
SeT Enable, ReseT Enable

ADC
ADC cONtrol, ADC High byte

*NOTE: Bit and byte addressable

**Figure 43. Special Function Registers**

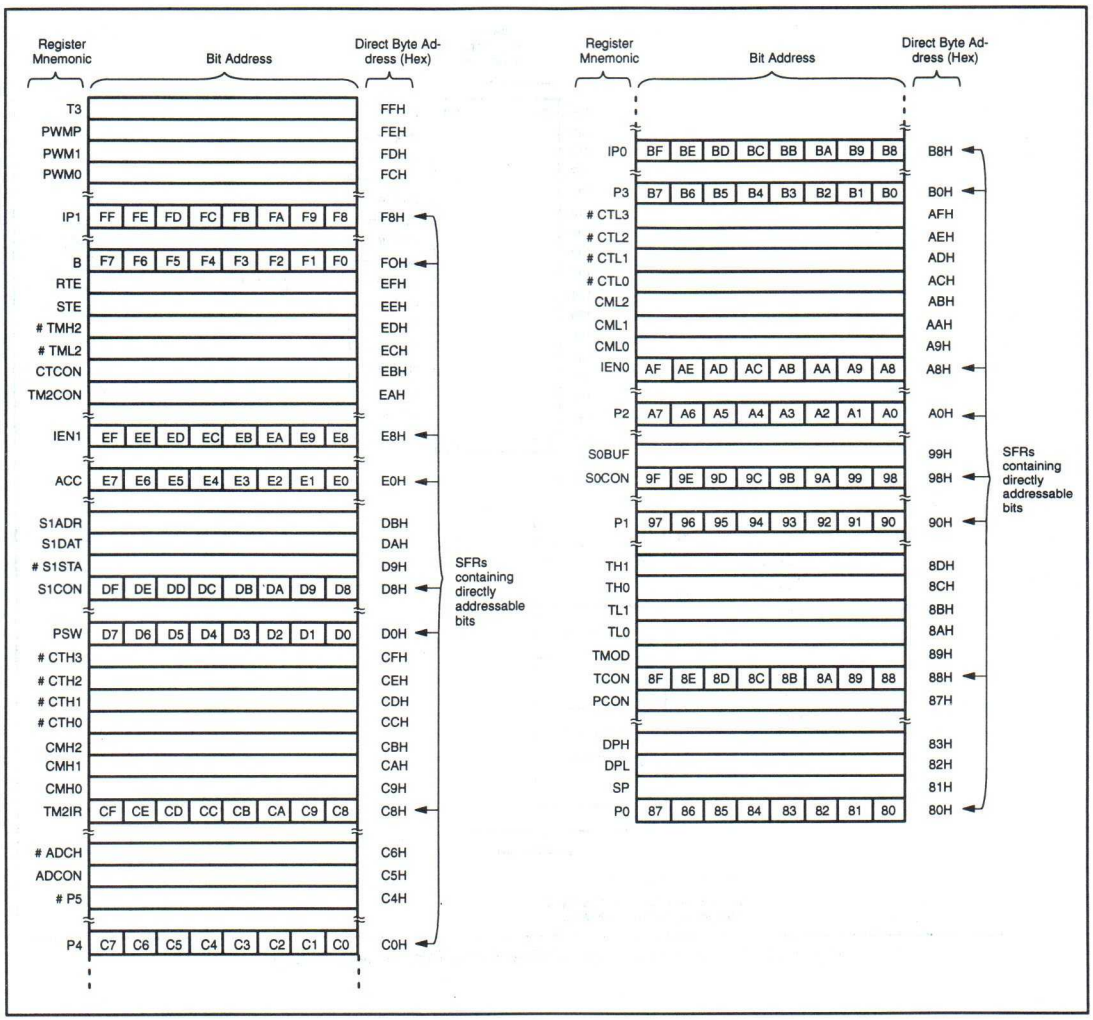



**Figure 44. Mapping of Special Function Registers**

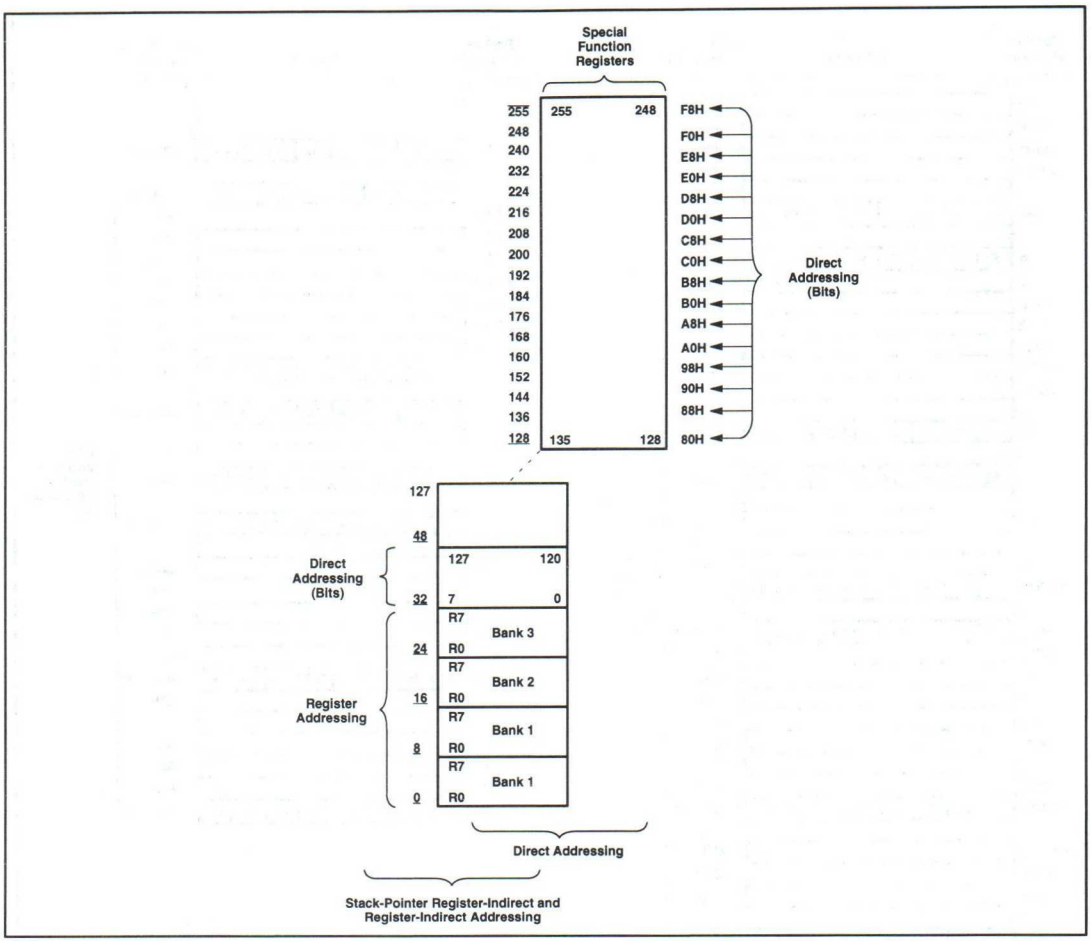



**Figure 45. Bit and Byte Addressing Overview of Internal Data Memory**

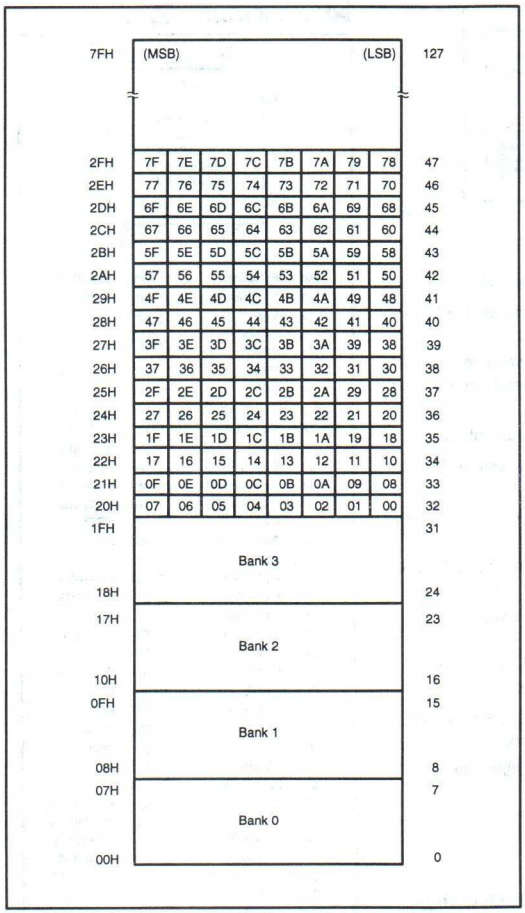



**Figure 46. RAM Bit Addresses**

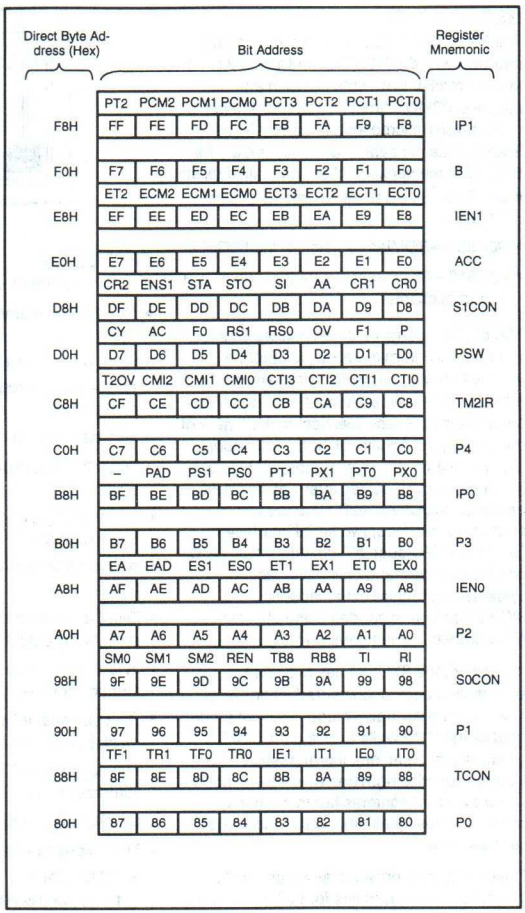



**Figure 47. Special Function Register Bit Address**

# Single-chip 8-bit microcontroller — 80C552/83C552

*Single-chip 8-bit microcontroller with 10-bit A/D, capture/compare timer, high-speed outputs, PWM*

## DESCRIPTION

The 80C552/83C552 (hereafter generically referred to as 8XC552) Single-Chip 8-Bit Microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 8XC552 has the same instruction set as the 80C51. Three versions of the derivative exist:

- 83C552—8k bytes mask programmable ROM
- 80C552—ROMless version of the 83C552
- 87C552—8k bytes EPROM (described in a separate chapter)

The 8XC552 contains a non-volatile 8k × 8 read-only program memory (83C552), a volatile 256 × 8 read/write data memory, five 8-bit I/O ports, one 8-bit input port, two 16-bit timer/event counters (identical to the timers of the 80C51), an additional 16-bit timer coupled to capture and compare latches, a 15-source, two-priority-level, nested interrupt structure, an 8-input ADC, a dual DAC pulse width modulated interface, two serial interfaces (UART and I²C-bus), a "watchdog" timer and on-chip oscillator and timing circuits. For systems that require extra capability, the 8XC552 can be expanded using standard TTL compatible memories and logic.

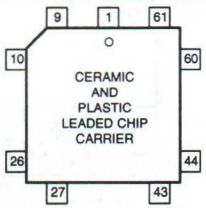
In addition, the 8XC552 has two software selectable modes of power reduction—idle mode and power-down mode. The idle mode freezes the CPU while allowing the RAM, timers, serial ports, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte, and 17 three-byte. With a 16MHz (24MHz) crystal, 58% of the instructions are executed in 0.75µs (0.5µs) and 40% in 1.5µs (1µs). Multiply and divide instructions require 3µs (2µs).

## FEATURES

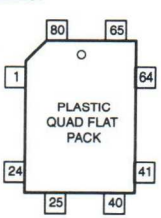
- 80C51 central processing unit
- 8k × 8 ROM expandable externally to 64k bytes
- An additional 16-bit timer/counter coupled to four capture registers and three compare registers
- Two standard 16-bit timer/counters
- 256 × 8 RAM, expandable externally to 64k bytes
- Capable of producing eight synchronized, timed outputs
- A 10-bit ADC with eight multiplexed analog inputs
- Two 8-bit resolution, pulse width modulation outputs
- Five 8-bit I/O ports plus one 8-bit input port shared with analog inputs
- I²C-bus serial I/O port with byte oriented master and slave functions
- Full-duplex UART compatible with the standard 80C51
- On-chip watchdog timer
- Three speed ranges:
  - 1.2 to 16MHz
  - 1.2 to 24MHz (ROM, ROMless only)
  - 1.2 to 30MHz (ROM, ROMless only)
- Three operating ambient temperature ranges:
  - PCB83C552–5: 0°C to +70°C
  - PCF83C552–5: –40°C to +85°C (XTAL frequency max. 24 MHz)
  - PCA83C552–5: –40°C to +125°C (XTAL frequency max. 16 MHz)

## PIN CONFIGURATIONS

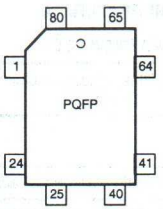



| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P5.0/ADC0 | 35 | XTAL1 |
| 2 | $V_{DD}$ | 36 | $V_{SS}$ |
| 3 | STADC | 37 | $V_{SS}$ |
| 4 | $\overline{PWM0}$ | 38 | NC* |
| 5 | $\overline{PWM1}$ | 39 | P2.0/A08 |
| 6 | $\overline{EW}$ | 40 | P2.1/A09 |
| 7 | P4.0/CMSR0 | 41 | P2.2/A10 |
| 8 | P4.1/CMSR1 | 42 | P2.3/A11 |
| 9 | P4.2/CMSR2 | 43 | P2.4/A12 |
| 10 | P4.3/CMSR3 | 44 | P2.5/A13 |
| 11 | P4.4/CMSR4 | 45 | P2.6/A14 |
| 12 | P4.5/CMSR5 | 46 | P2.7/A15 |
| 13 | P4.6/CMT0 | 47 | $\overline{PSEN}$ |
| 14 | P4.7/CMT1 | 48 | ALE |
| 15 | RST | 49 | $\overline{EA}$ |
| 16 | P1.0/CT0I | 50 | P0.7/AD7 |
| 17 | P1.1/CT1I | 51 | P0.6/AD6 |
| 18 | P1.2/CT2I | 52 | P0.5/AD5 |
| 19 | P1.3/CT3I | 53 | P0.4/AD4 |
| 20 | P1.4/T2 | 54 | P0.3/AD3 |
| 21 | P1.5/RT2 | 55 | P0.2/AD2 |
| 22 | P1.6/SCL | 56 | P0.1/AD1 |
| 23 | P1.7/SDA | 57 | P0.0/AD0 |
| 24 | P3.0/RxD | 58 | AVref– |
| 25 | P3.1/TxD | 59 | AVref+ |
| 26 | P3.2/$\overline{INT0}$ | 60 | $AV_{SS}$ |
| 27 | P3.3/$\overline{INT1}$ | 61 | $AV_{DD}$ |
| 28 | P3.4/T0 | 62 | P5.7/ADC7 |
| 29 | P3.5/T1 | 63 | P5.6/ADC6 |
| 30 | P3.6/$\overline{WR}$ | 64 | P5.5/ADC5 |
| 31 | P3.7/$\overline{RD}$ | 65 | P5.4/ADC4 |
| 32 | NC* | 66 | P5.3/ADC3 |
| 33 | NC* | 67 | P5.2/ADC2 |
| 34 | XTAL2 | 68 | P5.1/ADC1 |

*DO NOT CONNECT



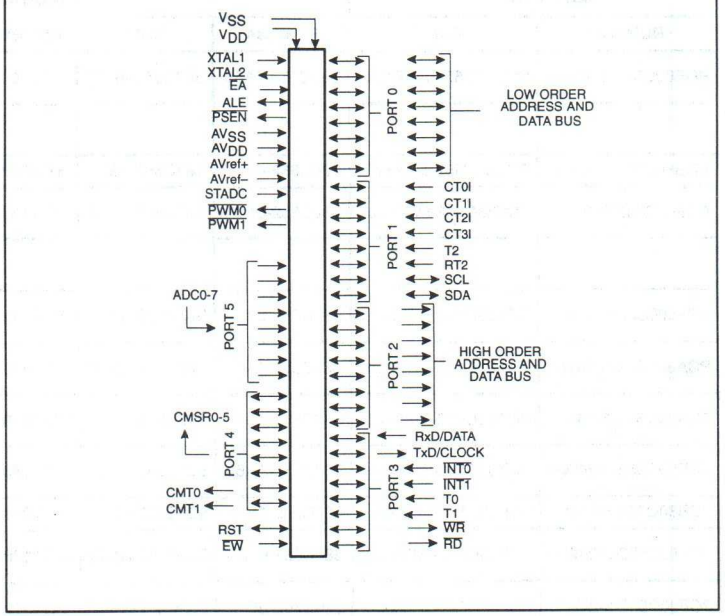
## PLASTIC QUAD FLAT PACK PIN FUNCTIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P4.1/CMSR1 | 41 | P2.3/A11 |
| 2 | P4.2/CMSR2 | 42 | P2.4/A12 |
| 3 | NC* | 43 | NC* |
| 4 | P4.3/CMSR3 | 44 | NC* |
| 5 | P4.4/CMSR4 | 45 | P2.5/A13 |
| 6 | P4.5/CMSR5 | 46 | P2.6/A14 |
| 7 | P4.6/CMT0 | 47 | P2.7/A15 |
| 8 | P4.7/CMT1 | 48 | $\overline{PSEN}$ |
| 9 | RST | 49 | ALE |
| 10 | P1.0/CT0I | 50 | $\overline{EA}$ |
| 11 | P1.1/CT1I | 51 | P0.7/AD7 |
| 12 | P1.2/CT2I | 52 | P0.6/AD6 |
| 13 | P1.3/CT3I | 53 | P0.5/AD5 |
| 14 | P1.4/T2 | 54 | P0.4/AD4 |
| 15 | P1.5/RT2 | 55 | P0.3/AD3 |
| 16 | P1.6/SCL | 56 | P0.2/AD2 |
| 17 | P1.7/SDA | 57 | P0.1/AD1 |
| 18 | P3.0/RxD | 58 | P0.0/AD0 |
| 19 | P3.1/TxD | 59 | AVref– |
| 20 | P3.2/$\overline{INT0}$ | 60 | AVref+ |
| 21 | NC* | 61 | $AV_{SS}$ |
| 22 | NC* | 62 | NC* |
| 23 | P3.3/$\overline{INT1}$ | 63 | $AV_{DD}$ |
| 24 | P3.4/T0 | 64 | P5.7/ADC7 |
| 25 | P3.5/T1 | 65 | P5.6/ADC6 |
| 26 | P3.6/$\overline{WR}$ | 66 | P5.5/ADC5 |
| 27 | P3.7/$\overline{RD}$ | 67 | P5.4/ADC4 |
| 28 | NC* | 68 | P5.3/ADC3 |
| 29 | NC* | 69 | P5.2/ADC2 |
| 30 | NC* | 70 | P5.1/ADC1 |
| 31 | XTAL2 | 71 | P5.0/ADC0 |
| 32 | XTAL1 | 72 | $V_{DD}$ |
| 33 | IC | 73 | IC |
| 34 | $V_{SS}$ | 74 | STADC |
| 35 | $V_{SS}$ | 75 | $\overline{PWM0}$ |
| 36 | $V_{SS}$ | 76 | $\overline{PWM1}$ |
| 37 | NC* | 77 | $\overline{EW}$ |
| 38 | P2.0/A08 | 78 | NC* |
| 39 | P2.1/A09 | 79 | NC* |
| 40 | P2.2/A10 | 80 | P4.0/CMSR0 |

* DO NOT CONNECT
IC = internally connected (do not use)

## LOGIC SYMBOL



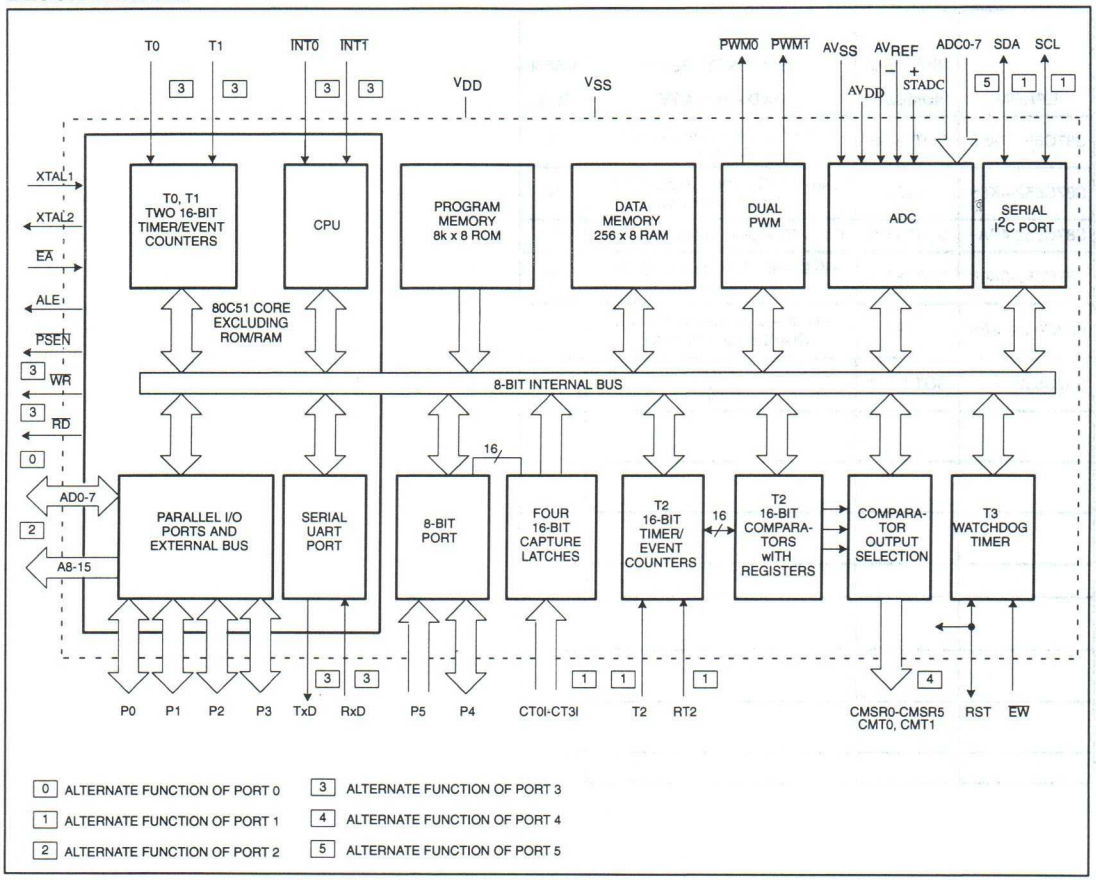
## ORDERING INFORMATION

| PHILIPS PART ORDER NUMBER PART MARKING | | NORTH AMERICA PHILIPS PART ORDER NUMBER | | DRAWING NUMBER | TEMPERATURE °C AND PACKAGE | FREQ MHz |
|---|---|---|---|---|---|---|
| ROMless | ROM | ROMless | ROM | | | |
| PCB80C552-5-16WP | PCB83C552-5WP/xxx | S80C552-4A68 | S83C552-4A68 | SOT188-3 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| | | | | | | |
| PCB80C552-5-16H | PCB83C552-5H/xxx | S80C552-4B | S83C552-4B | SOT318-2 | 0 to +70, Plastic Quad Flat Pack | 16 |
| PCF80C552-5-16WP | PCF83C552-5WP/xxx | S80C552-5A68 | S83C552-5A68 | SOT188-3 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| | | | | | | |
| PCF80C552-5-16H | PCF83C552-5H/xxx | S80C552-5B | S83C552-5B | SOT318-2 | –40 to +85, Plastic Quad Flat Pack | 16 |
| PCA80C552-5-16WP | PCA83C552-5WP/xxx | S80C552-6A68 | S83C552-6A68 | SOT188-3 | –40 to +125, Plastic Leaded Chip Carrier | 16 |
| PCA80C552-5-16H | PCA83C552-5H/xxx | S80C552-6B | S83C552-6B | SOT318-2 | –40 to +125, Plastic Quad Flat Pack | 16 |
| PCB80C552-5-24WP | PCB83C552-5WP/xxx | S80C552-AA68 | S83C552-AA68 | SOT188-3 | 0 to +70, Plastic Leaded Chip Carrier | 24 |
| PCB80C552-5-24H | PCB83C552-5H/xxx | S80C552-AB | S83C552-AB | SOT318-2 | 0 to +70, Plastic Quad Flat Pack | 24 |
| PCF80C552-5-24WP | PCF83C552-5WP/xxx | S80C552-BA68 | S83C552-BA68 | SOT188-3 | –40 to +85, Plastic Leaded Chip Carrier | 24 |
| PCF80C552-5-24H | PCF83C552-5H/xxx | S80C552-BB | S83C552-BB | SOT318-2 | –40 to +85, Plastic Quad Flat Pack | 24 |
| PCB80C552-5-30WP | PCB83C552-5WP/xxx | S80C552-CA68 | S83C552-CA68 | SOT188-3 | 0 to +70, Plastic Leaded Chip Carrier | 30 |
| PCB80C552-5-30H | PCB83C552-5H/xxx | S80C552-CB | S83C552-CB | SOT318-2 | 0 to +70, Plastic Quad Flat Pack | 30 |

**NOTE:**
1. xxx denotes the ROM code number.

| EPROM | DRAWING NUMBER | TEMPERATURE °C AND PACKAGE | FREQ MHz |
|---|---|---|---|
| S87C552-4A68 | SOT188-3 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| S87C552-4K68 | 1473A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 16 |
| S87C552-4BA | SOT318-2 | 0 to +70, Plastic Quad Flat Pack | 16 |
| S87C552-5A68 | SOT188-3 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| S87C552-5K68 | 1473A | –40 to +85, Ceramic Leaded Chip Carrier w/Window | 16 |
| S87C552-5BA | SOT318-2 | –40 to +85, Plastic Quad Flat Pack | 16 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

## BLOCK DIAGRAM



## PIN DESCRIPTION

| MNEMONIC | PIN NO. PLCC | PIN NO. QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|
| $V_{DD}$ | 2 | 72 | I | **Digital Power Supply:** +5V power supply pin during normal operation, idle and power-down mode. |
| STADC | 3 | 74 | I | **Start ADC Operation:** Input starting analog to digital conversion (ADC operation can also be started by software). This pin must not float. |
| $\overline{PWM0}$ | 4 | 75 | O | **Pulse Width Modulation:** Output 0. |
| $\overline{PWM1}$ | 5 | 76 | O | **Pulse Width Modulation:** Output 1. |
| $\overline{EW}$ | 6 | 77 | I | **Enable Watchdog Timer:** Enable for T3 watchdog timer and disable power-down mode. This pin must not float. |
| P0.0-P0.7 | 57-50 | 58-51 | I/O | **Port 0:** Port 0 is an 8-bit open-drain bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application it uses strong internal pull-ups when emitting 1s. |
| P1.0-P1.7 | 16-23 | 10-17 | I/O | **Port 1:** 8-bit I/O port. Alternate functions include: |
| | 16-21 | 10-15 | I/O | **(P1.0-P1.5):** Quasi-bidirectional port pins. |
| | 22-23 | 16-17 | I/O | **(P1.6, P1.7):** Open drain port pins. |
| | 16-19 | 10-13 | I | **CT0I-CT3I (P1.0-P1.3):** Capture timer input signals for timer T2. |
| | 20 | 14 | I | **T2 (P1.4):** T2 event input. |
| | 21 | 15 | I | **RT2 (P1.5):** T2 timer reset signal. Rising edge triggered. |
| | 22 | 16 | I/O | **SCL (P1.6):** Serial port clock line $I^2C$-bus. |
| | 23 | 17 | I/O | **SDA (P1.7):** Serial port data line $I^2C$-bus. |
| | | | | Port 1 is also used to input the lower order address byte during EPROM programming and verification. A0 is on P1.0, etc. |
| P2.0-P2.7 | 39-46 | 38-42, 45-47 | I/O | **Port 2:** 8-bit quasi-bidirectional I/O port. Alternate function: High-order address byte for external memory (A08-A15). |
| P3.0-P3.7 | 24-31 | 18-20, 23-27 | I/O | **Port 3:** 8-bit quasi-bidirectional I/O port. Alternate functions include: |
| | 24 | 18 | | **RxD(P3.0):** Serial input port. |
| | 25 | 19 | | **TxD (P3.1):** Serial output port. |
| | 26 | 20 | | **$\overline{INT0}$ (P3.2):** External interrupt. |
| | 27 | 23 | | **$\overline{INT1}$ (P3.3):** External interrupt. |
| | 28 | 24 | | **T0 (P3.4):** Timer 0 external input. |
| | 29 | 25 | | **T1 (P3.5):** Timer 1 external input. |
| | 30 | 26 | | **$\overline{WR}$ (P3.6):** External data memory write strobe. |
| | 31 | 27 | | **$\overline{RD}$ (P3.7):** External data memory read strobe. |
| P4.0-P4.7 | 7-14 | 80, 1-2 4-8 | I/O | **Port 4:** 8-bit quasi-bidirectional I/O port. Alternate functions include: |
| | 7-12 | 80, 1-2 4-6 | O | **CMSR0-CMSR5 (P4.0-P4.5):** Timer T2 compare and set/reset outputs on a match with timer T2. |
| | 13, 14 | 7, 8 | O | **CMT0, CMT1 (P4.6, P4.7):** Timer T2 compare and toggle outputs on a match with timer T2. |
| P5.0-P5.7 | 68-62, 1 | 71-64, | I | **Port 5:** 8-bit input port. **ADC0-ADC7 (P5.0-P5.7):** Alternate function: Eight input channels to ADC. |
| RST | 15 | 9 | I/O | **Reset:** Input to reset the 8XC552. It also provides a reset pulse as output when timer T3 overflows. |
| XTAL1 | 35 | 32 | I | **Crystal Input 1:** Input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external clock signal when an external oscillator is used. |
| XTAL2 | 34 | 31 | O | **Crystal Input 2:** Output of the inverting amplifier that forms the oscillator. Left open-circuit when an external clock is used. |

## PIN DESCRIPTION (Continued)

| MNEMONIC | PIN NO. PLCC | PIN NO. QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|
| $V_{SS}$ | 36, 37 | 34-36 | I | **Two Digital ground pins.** |
| $\overline{PSEN}$ | 47 | 48 | O | **Program Store Enable:** Active-low read strobe to external program memory. |
| ALE | 48 | 49 | O | **Address Latch Enable:** Latches the low byte of the address during accesses to external memory. It is activated every six oscillator periods. During an external data memory access, one ALE pulse is skipped. ALE can drive up to eight LS TTL inputs and handles CMOS inputs without an external pull-up. |
| $\overline{EA}$ | 49 | 50 | I | **External Access:** When $\overline{EA}$ is held at TTL level high, the CPU executes out of the internal program ROM provided the program counter is less than 8192. When $\overline{EA}$ is held at TTL low level, the CPU executes out of external program memory. $\overline{EA}$ is not allowed to float. |
| $AV_{REF-}$ | 58 | 59 | I | **Analog to Digital Conversion Reference Resistor:** Low-end. |
| $AV_{REF+}$ | 59 | 60 | I | **Analog to Digital Conversion Reference Resistor:** High-end. |
| $AV_{SS}$ | 60 | 61 | I | **Analog Ground** |
| $AV_{DD}$ | 61 | 63 | I | **Analog Power Supply** |

**NOTE:**
1. To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher or lower than $V_{DD}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the logic symbol, page 3-531.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{DD}$ and RST must come up at the same time for a proper start-up.

## IDLE MODE

In the idle mode, the CPU puts itself to sleep while some of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. The control bits for the reduced power modes are in the special function register PCON. Table 1 shows the state of the I/O ports during low current operating modes.

**Table 1. External Pin Status During Idle and Power-Down Modes**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 | PORT 4 | PWM0/ PWM1 |
|---|---|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data | Data | 1 |
| Idle | External | 1 | 1 | Float | Data | Address | Data | Data | 1 |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data | Data | 1 |
| Power-down | External | 0 | 0 | Float | Data | Data | Data | Data | 1 |

### Serial Control Register (S1CON) – See Table 2

| S1CON (D8H) | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |
|---|---|---|---|---|---|---|---|---|

**Bits CR0, CR1 and CR2 determine the serial clock frequency that is generated in the master mode of operation.**

### Table 2. Serial Clock Rates

| CR2 | CR1 | CR0 | BIT FREQUENCY (kHz) AT $f_{OSC}$ 6MHz | 12MHz | 16MHz | 24MHz[2] | 30MHz[2] | $f_{OSC}$ DIVIDED BY |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 23 | 47 | 62.5 | 94 | 117 [1] | 256 |
| 0 | 0 | 1 | 27 | 54 | 71 | 107 [1] | 134 [1] | 224 |
| 0 | 1 | 0 | 31 | 63 | 83.3 | 125 [1] | 156 [1] | 192 |
| 0 | 1 | 1 | 37 | 75 | 100 | 150 [1] | 188 [1] | 160 |
| 1 | 0 | 0 | 6.25 | 12.5 | 17 | 25 | 31 | 960 |
| 1 | 0 | 1 | 50 | 100 | 133 [1] | 200 [1] | 250 [1] | 120 |
| 1 | 1 | 0 | 100 | 200 | 267 [1] | 400 [1] | 500 [1] | 60 |
| 1 | 1 | 1 | 0.24 < 62.5<br>0 < 255 | 0.49 < 62.5<br>0 < 254 | 0.65 < 55.6<br>0 < 253 | 0.98 < 50.0<br>0 <251 | 1.22 < 52.1<br>0 < 250 | 96 × (256 – (reload value Timer 1))<br>reload value Timer 1 in Mode 2. |

**NOTES:**
1. These frequencies exceed the upper limit of 100kHz of the I²C-bus specification and cannot be used in an I²C-bus application.
2. At $f_{OSC}$ = 24MHz/ 30MHz the maximum I²C bus rate of 100kHz cannot be realized due to the fixed divider rates.

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Input, output DC current on any single I/O pin | 5.0 | mA |
| Power dissipation<br>(based on package heat transfer limitations, not device power consumption) | 1.0 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DEVICE SPECIFICATIONS

| TYPE | SUPPLY VOLTAGE (V) MIN | MAX | FREQUENCY (MHz) MIN | MAX | TEMPERATURE RANGE (°C) |
|---|---|---|---|---|---|
| PCB83(0)C552-5-16 | 4.0 | 6.0 | 1.2 | 16 | 0 to +70 |
| PCF83(0)C552-5-16 | 4.0 | 6.0 | 1.2 | 16 | –40 to +85 |
| PCA83(0)C552-5-16 | 4.5 | 5.5 | 1.2 | 16 | –40 to +125 |
| PCB83(0)C552-5-24 | 4.5 | 5.5 | 1.2 | 24 | 0 to +70 |
| PCF83(0)C552-5-24 | 4.5 | 5.5 | 1.2 | 24 | –40 to +85 |
| PCB83(0)C552-5-30 | 4.5 | 5.5 | 1.2 | 30 | 0 to +70 |

## DC ELECTRICAL CHARACTERISTICS

$V_{SS}$, $AV_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|
| $I_{DD}$ | Supply current operating: | See notes 1 and 2 | | | |
| | PCB8XC552-5-16 | $f_{OSC}$ = 16MHz | | 45 | mA |
| | PCF8XC552-5-16 | $f_{OSC}$ = 16MHz | | 45 | mA |
| | PCA8XC552-5-16 | $f_{OSC}$ = 16MHz | | 40 | mA |
| | PCB8XC552-5-24 | $f_{OSC}$ = 24MHz | | 55 | mA |
| | PCF8XC552-5-24 | $f_{OSC}$ = 24MHz | | 55 | mA |
| | PCB8XC552-5-30 | $f_{OSC}$ = 30MHz | | 68 | mA |
| $I_{ID}$ | Idle mode: | See notes 1 and 3 | | | |
| | PCB8XC552-5-16 | $f_{OSC}$ = 16MHz | | 10 | mA |
| | PCF8XC552-5-16 | $f_{OSC}$ = 16MHz | | 10 | mA |
| | PCA8XC552-5-16 | $f_{OSC}$ = 16MHz | | 9 | mA |
| | PCB8XC552-5-24 | $f_{OSC}$ = 24MHz | | 12.5 | mA |
| | PCF8XC552-5-24 | $f_{OSC}$ = 24MHz | | 12.5 | mA |
| | PCB8XC552-5-30 | $f_{OSC}$ = 30MHz | | 15 | mA |
| $I_{PD}$ | Power-down current: | See notes 1 and 4; $2V < V_{PD} < V_{DD}$ max | | | |
| | PCB8XC552 | | | 50 | µA |
| | PCF8XC552 | | | 50 | µA |
| | PCA8XC552 | | | 150 | µA |
| **Inputs** | | | | | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$, P1.6, P1.7 | | –0.5 | $0.2V_{DD}$–0.1 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | | –0.5 | $0.2V_{DD}$–0.3 | V |
| $V_{IL2}$ | Input low voltage to P1.6/SCL, P1.7/SDA[5] | | –0.5 | $0.3V_{DD}$ | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST, P1.6/SCL, P1.7/SDA | | $0.2V_{DD}$+0.9 | $V_{DD}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | | $0.7V_{DD}$ | $V_{DD}$+0.5 | V |
| $V_{IH2}$ | Input high voltage, P1.6/SCL, P1.7/SDA[5] | | $0.7V_{DD}$ | 6.0 | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3, 4, except P1.6, P1.7 | $V_{IN}$ = 0.45V | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3, 4, except P1.6, P1.7 | See note 6 | | –650 | µA |
| $\pm I_{IL1}$ | Input leakage current, port 0, $\overline{EA}$, STADC, $\overline{EW}$ | $0.45V < V_I < V_{DD}$ | | 10 | µA |
| $\pm I_{IL2}$ | Input leakage current, P1.6/SCL, P1.7/SDA | $0V < V_I < 6V$<br>$0V < V_{DD} < 5.5V$ | | 10 | µA |
| $\pm I_{IL3}$ | Input leakage current, port 5 | $0.45V < V_I < V_{DD}$ | | 1 | µA |
| **Outputs** | | | | | |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3, 4, except P1.6, P1.7 | $I_{OL}$ = 1.6mA[7] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$ | $I_{OL}$ = 3.2mA[7] | | 0.45 | V |
| $V_{OL2}$ | Output low voltage, P1.6/SCL, P1.7/SDA | $I_{OL}$ = 3.0mA[7] | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3, 4, except P1.6/SCL, P1.7/SDA | $V_{DD}$ = 5V +10%<br>$-I_{OH}$ = 60µA<br>$-I_{OH}$ = 25µA<br>$-I_{OH}$ = 10µA | <br>2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | <br>V<br>V<br>V |

## DC ELECTRICAL CHARACTERISTICS (Continued)

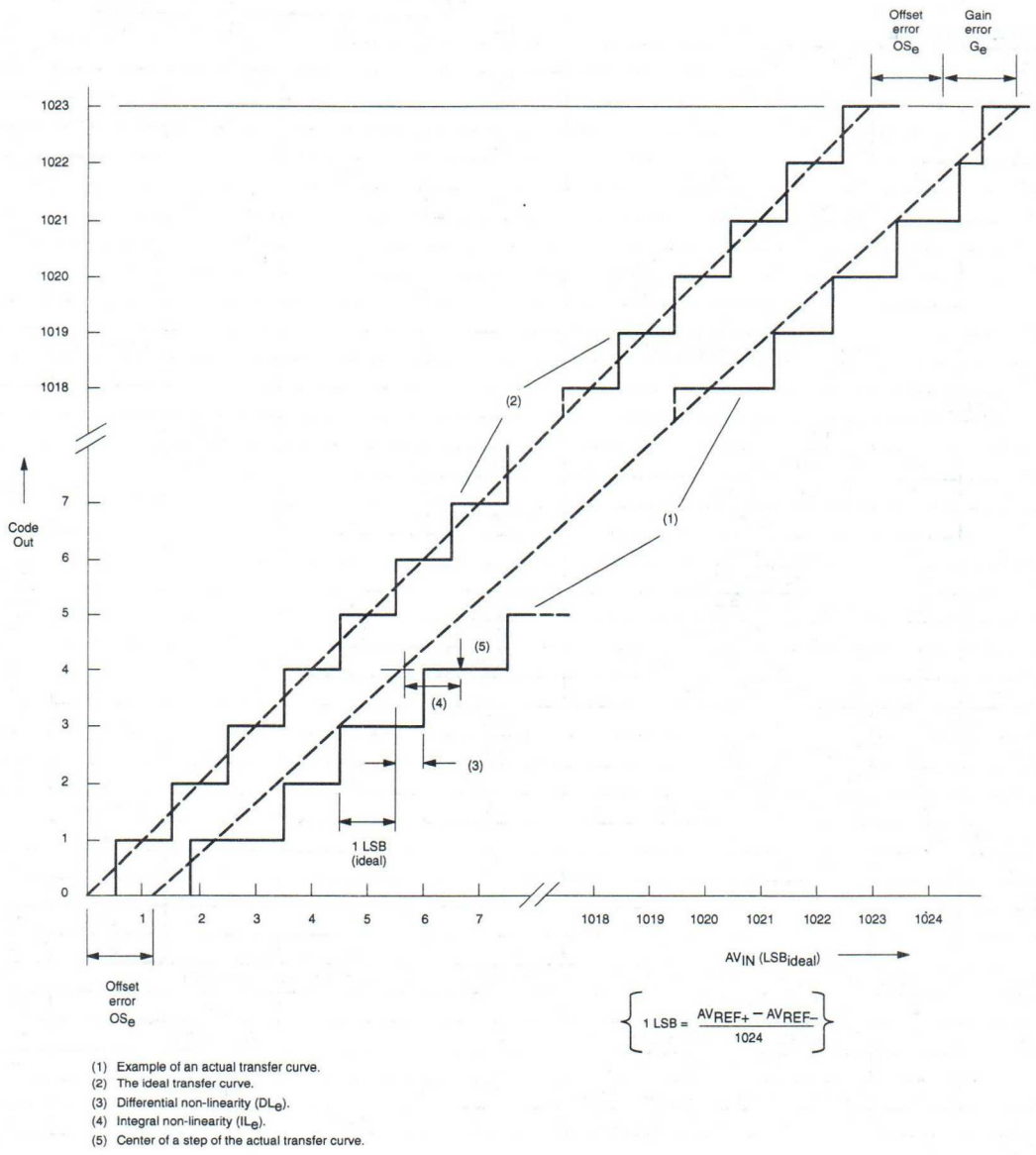
| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|
| **Outputs (Continued)** | | | | | |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$)[8] | $V_{DD}$ = 5V +10%<br>$-I_{OH}$ = 400µA<br>$-I_{OH}$ = 150µA<br>$-I_{OH}$ = 40µA | <br>2.4<br>0.75$V_{DD}$<br>0.9$V_{DD}$ | | <br>V<br>V<br>V |
| $V_{OH2}$ | Output high voltage (RST) | $-I_{OH}$ = 400µA<br>$-I_{OH}$ = 120µA | 2.4<br>0.8$V_{DD}$ | | V<br>V |
| $R_{RST}$ | Internal reset pull-down resistor | | 50 | 150 | kΩ |
| $C_{IO}$ | Pin capacitance | Test freq = 1MHz,<br>$T_{amb}$ = 25°C | | 10 | pF |
| **Analog Inputs** | | | | | |
| $AV_{DD}$ | Analog supply voltage:<br>PCB8XC552-5-16<br>PCF8XC552-5-16<br>PCA8XC552-5-16<br>PCB8XC552-5-24<br>PCF8XC552-5-24<br>PCB8XC552-5-30 | <br>$AV_{DD} = V_{DD} \pm 0.2V$<br>$AV_{DD} = V_{DD} \pm 0.2V$<br>$AV_{DD} = V_{DD} \pm 0.2V$<br>$AV_{DD} = V_{DD} \pm 0.2V$<br>$AV_{DD} = V_{DD} \pm 0.2V$<br>$AV_{DD} = V_{DD} \pm 0.2V$ | <br>4.0<br>4.0<br>4.5<br>4.5<br>4.5<br>4.5 | <br>6.0<br>6.0<br>5.5<br>5.5<br>5.5<br>5.5 | <br>V<br>V<br>V<br>V<br>V<br>V |
| $AI_{DD}$ | Analog supply current: operating: (16MHz)<br>Analog supply current: operating: (24MHz, 30MHz) | Port 5 = 0 to $AV_{DD}$<br>Port 5 = 0 to $AV_{DD}$ | | 1.2<br>1.0 | mA<br>mA |
| $AI_{ID}$ | Idle mode:<br>PCB8XC552-5-16<br>PCF8XC552-5-16<br>PCA8XC552-5-16<br>PCB8XC552-5-24<br>PCF8XC552-5-24<br>PCB8XC552-5-30 | | | <br>50<br>50<br>100<br>50<br>50<br>50 | <br>µA<br>µA<br>µA<br>µA<br>µA<br>µA |
| $AI_{PD}$ | Power-down mode:<br>PCB8XC552<br>PCF8XC552<br>PCA8XC552 | 2V < $AV_{PD}$ < $AV_{DD}$ max | | <br>50<br>50<br>100 | <br>µA<br>µA<br>µA |
| $AV_{IN}$ | Analog input voltage | | $AV_{SS}$–0.2 | $AV_{DD}$+0.2 | V |
| $AV_{REF}$ | Reference voltage:<br>$AV_{REF-}$<br>$AV_{REF+}$ | | <br>$AV_{SS}$–0.2 | <br>$AV_{DD}$+0.2 | <br>V<br>V |
| $R_{REF}$ | Resistance between $AV_{REF+}$ and $AV_{REF-}$ | | 10 | 50 | kΩ |
| $C_{IA}$ | Analog input capacitance | | | 15 | pF |
| $t_{ADS}$ | Sampling time | | | 8$t_{CY}$ | µs |
| $t_{ADC}$ | Conversion time (including sampling time) | | | 50$t_{CY}$ | µs |
| $DL_e$ | Differential non-linearity[10, 11, 12] | | | ±1 | LSB |
| $IL_e$ | Integral non-linearity[10, 13] | | | ±2 | LSB |
| $OS_e$ | Offset error[10, 14] | | | ±2 | LSB |

## DC ELECTRICAL CHARACTERISTICS (Continued)

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|
| **Analog Inputs (continued)** | | | | | |
| $G_e$ | Gain error[10, 15] | | | ±0.4 | % |
| $A_e$ | Absolute voltage error[10, 16] | | | ±3 | LSB |
| $M_{CTC}$ | Channel to channel matching | | | ±1 | LSB |
| $C_t$ | Crosstalk between inputs of port 5[17] | 0–100kHz | | –60 | dB |

**NOTES FOR DC ELECTRICAL CHARACTERISTICS:**

1. See Figures 10 through 15 for $I_{DD}$ test conditions.
2. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 10ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5V$; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = $\overline{EW}$ = $V_{DD}$; STADC = $V_{SS}$.
3. The idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 10ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5V$; XTAL2 not connected; Port 0 = $\overline{EW}$ = $V_{DD}$; $\overline{EA}$ = RST = STADC = $V_{SS}$.
4. The power-down current is measured with all output pins disconnected; XTAL2 not connected; Port 0 = $\overline{EW}$ = $V_{DD}$; $\overline{EA}$ = RST = STADC = XTAL1 = $V_{SS}$.
5. The input threshold voltage of P1.6 and P1.7 (SIO1) meets the $I^2C$ specification, so an input voltage below 1.5V will be recognized as a logic 0 while an input voltage above 3.0V will be recognized as a logic 1.
6. Pins of ports 1 (except P1.6, P1.7), 2, 3, and 4 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
7. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
8. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{DD}$ specification when the address bits are stabilizing.
9. The following condition must not be exceeded: $V_{DD} - 0.2V < AV_{DD} < V_{DD} + 0.2V$.
10. Conditions: $AV_{REF-} = 0V$; $AV_{DD} = 5.0V$, $AV_{REF+}$ (80C552, 83C552) = 5.12V. ADC is monotonic with no missing codes. Measurement by continuous conversion of $AV_{IN}$ = –20mV to 5.12V in steps of 0.5mV.
11. The differential non-linearity ($DL_e$) is the difference between the actual step width and the ideal step width. (See Figure 1.)
12. The ADC is monotonic; there are no missing codes.
13. The integral non-linearity ($IL_e$) is the peak difference between the center of the steps of the actual and the ideal transfer curve after appropriate adjustment of gain and offset error. (See Figure 1.)
14. The offset error ($OS_e$) is the absolute difference between the straight line which fits the actual transfer curve (after removing gain error), and a straight line which fits the ideal transfer curve. (See Figure 1.)
15. The gain error ($G_e$) is the relative difference in percent between the straight line fitting the actual transfer curve (after removing offset error), and the straight line which fits the ideal transfer curve. Gain error is constant at every point on the transfer curve. (See Figure 1.)
16. The absolute voltage error ($A_e$) is the maximum difference between the center of the steps of the actual transfer curve of the non-calibrated ADC and the ideal transfer curve.
17. This should be considered when both analog and digital signals are simultaneously input to port 5.




(1) Example of an actual transfer curve.
(2) The ideal transfer curve.
(3) Differential non-linearity ($DL_e$).
(4) Integral non-linearity ($IL_e$).
(5) Center of a step of the actual transfer curve.

**Figure 1. ADC Conversion Characteristic**

## AC ELECTRICAL CHARACTERISTICS[1, 2]

16 MHz version

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 2 | Oscillator frequency | | | 1.2 | 16 | MHz |
| $t_{LHLL}$ | 2 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 2 | Address valid to ALE low | 8 | | $t_{CLCL}-55$ | | ns |
| $t_{LLAX}$ | 2 | Address hold after ALE low | 28 | | $t_{CLCL}-35$ | | ns |
| $t_{LLIV}$ | 2 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 2 | ALE low to $\overline{PSEN}$ low | 23 | | $t_{CLCL}-40$ | | ns |
| $t_{PLPH}$ | 2 | $\overline{PSEN}$ pulse width | 143 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 2 | $\overline{PSEN}$ low to valid instruction in | | 83 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 2 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 2 | Input instruction float after $\overline{PSEN}$ | | 38 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 2 | Address to valid instruction in | | 208 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 2 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 3 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 4 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 3 | $\overline{RD}$ low to valid data in | | 148 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 3 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 3 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-70$ | ns |
| $t_{LLDV}$ | 3 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 3 | Address to valid data in | | 398 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 3, 4 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 138 | 238 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 3, 4 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 120 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 4 | Data valid to $\overline{WR}$ transition | 3 | | $t_{CLCL}-60$ | | ns |
| $t_{DW}$ | 4 | Data before $\overline{WR}$ | 288 | | $7t_{CLCL}-150$ | | ns |
| $t_{WHQX}$ | 4 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{RLAZ}$ | 3 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 3, 4 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 5 | High time[4] | 20 | | 20 | | ns |
| $t_{CLCX}$ | 5 | Low time[4] | 20 | | 20 | | ns |
| $t_{CLCH}$ | 5 | Rise time[4] | | 20 | | 20 | ns |
| $t_{CHCL}$ | 5 | Fall time[4] | | 20 | | 20 | ns |
| **Serial Timing – Shift Register Mode[4]** (Test Conditions: $T_{amb}$ = 0°C to +70°C; $V_{SS}$ = 0V; Load Capacitance = 80pF) | | | | | | | |
| $t_{XLXL}$ | 6 | Serial port clock cycle time | 0.75 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 6 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 6 | Output data hold after clock rising edge | 8 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 6 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 6 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}-133$ | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. $t_{CLCL} = 1/f_{OSC}$ = one oscillator clock period.
   $t_{CLCL}$ = 83.3ns at $f_{OSC}$ = 12MHz.
   $t_{CLCL}$ = 62.5ns at $f_{OSC}$ = 16MHz.
4. These values are characterized but not 100% production tested.

## AC ELECTRICAL CHARACTERISTICS (Continued)[1, 2]

24/30 MHz version

| SYMBOL | FIGURE | PARAMETER | 24MHz CLOCK | | 30MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 2 | Oscillator frequency | | | | | 1.2 | 24 | MHz |
| $t_{LHLL}$ | 2 | ALE pulse width | 43 | | 27 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 2 | Address valid to ALE low | 17 | | 8 | | $t_{CLCL}-25$ | | ns |
| $t_{LLAX}$ | 2 | Address hold after ALE low | 17 | | 8 | | $t_{CLCL}-25$ | | ns |
| $t_{LLIV}$ | 2 | ALE low to valid instruction in | | 102 | | 68 | | $4t_{CLCL}-65$ | ns |
| $t_{LLPL}$ | 2 | ALE low to $\overline{PSEN}$ low | 17 | | 8 | | $t_{CLCL}-25$ | | ns |
| $t_{PLPH}$ | 2 | $\overline{PSEN}$ pulse width | 80 | | 55 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 2 | $\overline{PSEN}$ low to valid instruction in | | 65 | | 40 | | $3t_{CLCL}-60$ | ns |
| $t_{PXIX}$ | 2 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 2 | Input instruction float after $\overline{PSEN}$ | | 17 | | 8 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 2 | Address to valid instruction in | | 128 | | 87 | | $5t_{CLCL}-80$ | ns |
| $t_{PLAZ}$ | 2 | $\overline{PSEN}$ low to address float | | 10 | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | | | |
| $t_{RLRH}$ | 3 | $\overline{RD}$ pulse width | 150 | | 100 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 4 | $\overline{WR}$ pulse width | 150 | | 100 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 3 | $\overline{RD}$ low to valid data in | | 118 | | 77 | | $5t_{CLCL}-90$ | ns |
| $t_{RHDX}$ | 3 | Data hold after $\overline{RD}$ | 0 | | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 3 | Data float after $\overline{RD}$ | | 55 | | 39 | | $2t_{CLCL}-28$ | ns |
| $t_{LLDV}$ | 3 | ALE low to valid data in | | 183 | | 117 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 3 | Address to valid data in | | 210 | | 135 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 3, 4 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 75 | 175 | 50 | 150 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 3, 4 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 92 | | 58 | | $4t_{CLCL}-75$ | | ns |
| $t_{QVWX}$ | 4 | Data valid to $\overline{WR}$ transition | 12 | | 3 | | $t_{CLCL}-30$ | | ns |
| $t_{DW}$ | 4 | Data before $\overline{WR}$ | 162 | | 103 | | $7t_{CLCL}-130$ | | ns |
| $t_{WHQX}$ | 4 | Data hold after $\overline{WR}$ | 17 | | 8 | | $t_{CLCL}-25$ | | ns |
| $t_{RLAZ}$ | 3 | $\overline{RD}$ low to address float | | 0 | | 0 | | 0 | ns |
| $t_{WHLH}$ | 3, 4 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 17 | 67 | 8 | 58 | $t_{CLCL}-25$ | $t_{CLCL}+25$ | ns |
| **External Clock** | | | | | | | | | |
| $t_{CHCX}$ | 5 | High time[3] | 17 | | 15 | | 17 | | ns |
| $t_{CLCX}$ | 5 | Low time[3] | 17 | | 15 | | 17 | | ns |
| $t_{CLCH}$ | 5 | Rise time[3] | | 5 | | 3 | | 20 | ns |
| $t_{CHCL}$ | 5 | Fall time[3] | | 5 | | 3 | | 20 | ns |
| **Serial Timing – Shift Register Mode[3]** (Test Conditions: $T_{amb}$ = 0°C to +70°C; $V_{SS}$ = 0V; Load Capacitance = 80pF) | | | | | | | | | |
| $t_{XLXL}$ | 6 | Serial port clock cycle time | 0.5 | | 0.4 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 6 | Output data setup to clock rising edge | 283 | | 200 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 6 | Output data hold after clock rising edge | 23 | | 6.6 | | $2t_{CLCL}-60$ | | ns |
| $t_{XHDX}$ | 6 | Input data hold after clock rising edge | 0 | | 0 | | 0 | | ns |
| $t_{XHDV}$ | 6 | Clock rising edge to input data valid | | 283 | | 200 | | $10t_{CLCL}-133$ | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. These values are characterized but not 100% production tested.
4. $t_{CLCL} = 1/f_{OSC}$ = one oscillator clock period.
   $t_{CLCL}$ = 41.7ns at $f_{OSC}$ = 24MHz.

## AC ELECTRICAL CHARACTERISTICS (Continued)

| SYMBOL | PARAMETER | INPUT | OUTPUT |
|---|---|---|---|
| **I²C Interface (Refer to Figure 9)** | | | |
| $t_{HD;STA}$ | START condition hold time | ≥ 14 $t_{CLCL}$ | > 4.0µs [1] |
| $t_{LOW}$ | SCL low time | ≥ 16 $t_{CLCL}$ | > 4.7µs [1] |
| $t_{HIGH}$ | SCL high time | ≥ 14 $t_{CLCL}$ | > 4.0µs [1] |
| $t_{RC}$ | SCL rise time | ≤ 1µs | – [2] |
| $t_{FC}$ | SCL fall time | ≤ 0.3µs | < 0.3µs [3] |
| $t_{SU;DAT1}$ | Data set-up time | ≥ 250ns | > 20 $t_{CLCL}$ – $t_{RD}$ |
| $t_{SU;DAT2}$ | SDA set-up time (before rep. START cond.) | ≥ 250ns | > 1µs [1] |
| $t_{SU;DAT3}$ | SDA set-up time (before STOP cond.) | ≥ 250ns | > 8 $t_{CLCL}$ |
| $t_{HD;DAT}$ | Data hold time | ≥ 0ns | > 8 $t_{CLCL}$ – $t_{FC}$ |
| $t_{SU;STA}$ | Repeated START set-up time | ≥ 14 $t_{CLCL}$ | > 4.7µs [1] |
| $t_{SU;STO}$ | STOP condition set-up time | ≥ 14 $t_{CLCL}$ | > 4.0µs [1] |
| $t_{BUF}$ | Bus free time | ≥ 14 $t_{CLCL}$ | > 4.7µs [1] |
| $t_{RD}$ | SDA rise time | ≤ 1µs | – [2] |
| $t_{FD}$ | SDA fall time | ≤ 0.3µs | < 0.3µs [3] |

**NOTES:**
1. At 100 kbit/s. At other bit rates this value is inversely proportional to the bit-rate of 100 kbit/s.
2. Determined by the external bus-line capacitance and the external bus-line pull-resistor, this must be < 1µs.
3. Spikes on the SDA and SCL lines with a duration of less than 3 $t_{CLCL}$ will be filtered out. Maximum capacitance on bus-lines SDA and SCL = 400pF.
4. $t_{CLCL} = 1/f_{OSC}$ = one oscillator clock period at pin XTAL1. For 62ns, 42ns, 33.3ns < $t_{CLCL}$ < 285ns (16MHz, 24MHz, 30MHz > $f_{OSC}$ > 1.2MHz) the SI01 interface meets the I²C-bus specification for bit-rates up to 100 kbit/s.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{\text{PSEN}}$
Q – Output data
R – $\overline{\text{RD}}$ signal
t – Time
V – Valid
W – $\overline{\text{WR}}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{\text{PSEN}}$ low.

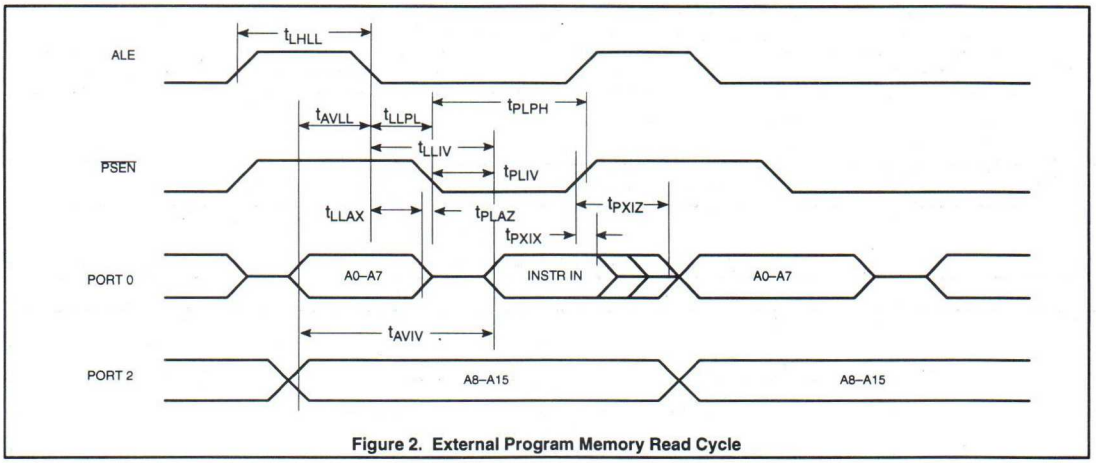



**Figure 2. External Program Memory Read Cycle**

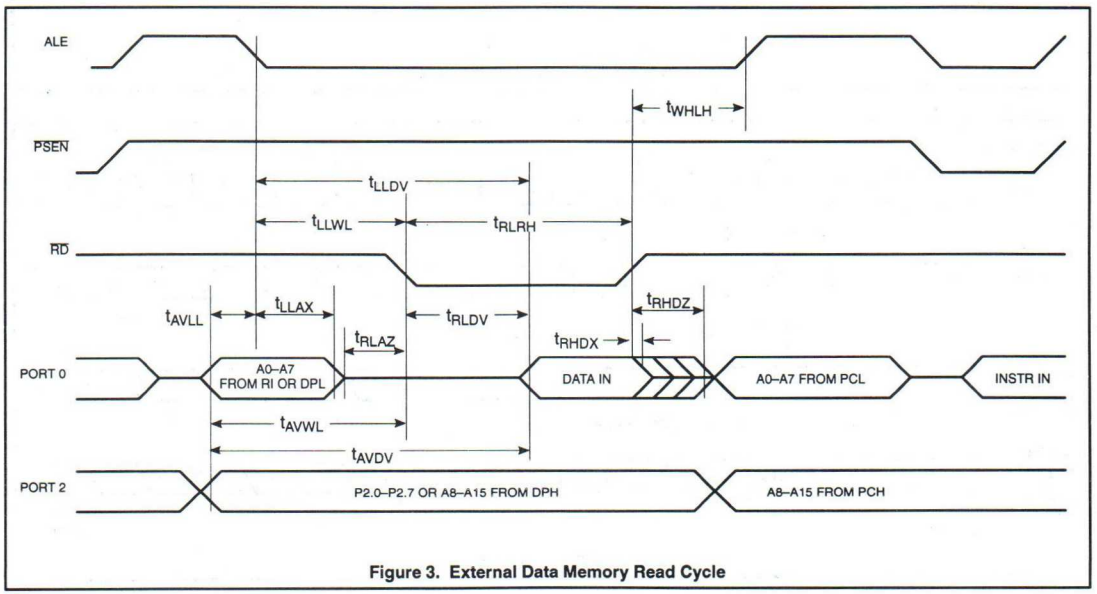



**Figure 3. External Data Memory Read Cycle**

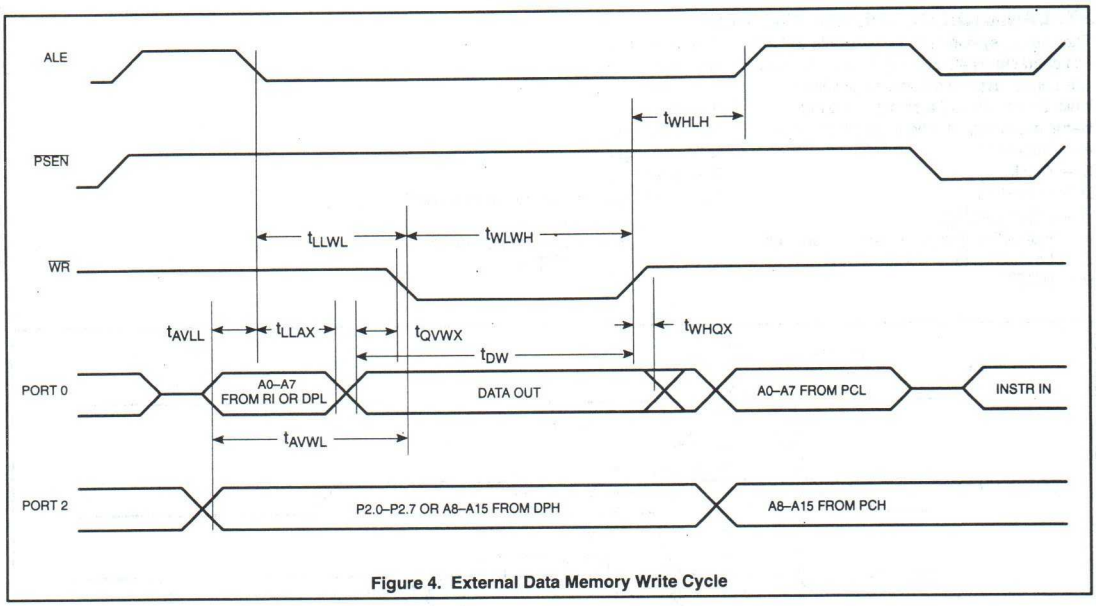



**Figure 4. External Data Memory Write Cycle**

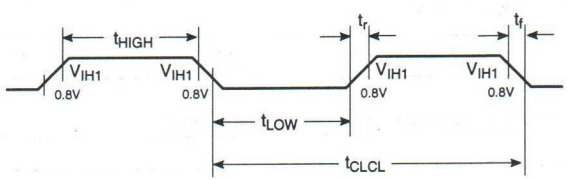



**Figure 5. External Clock Drive XTAL1**

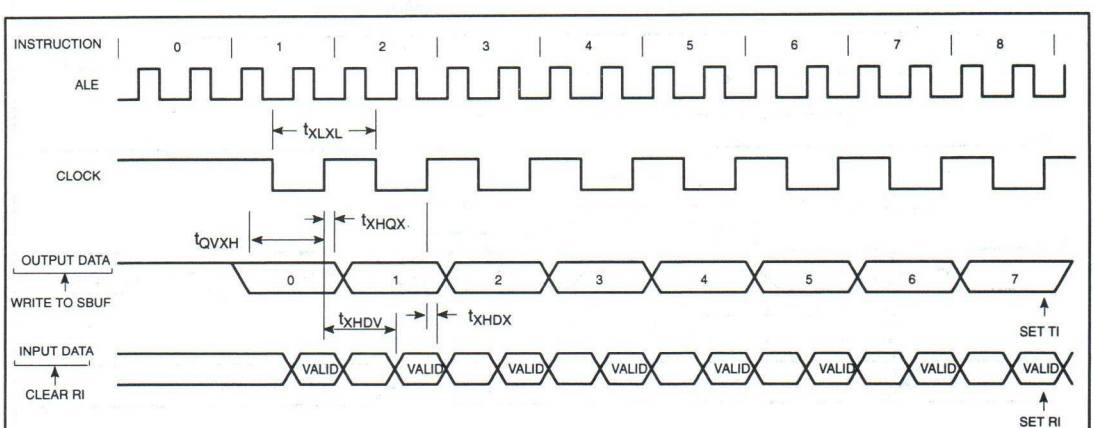
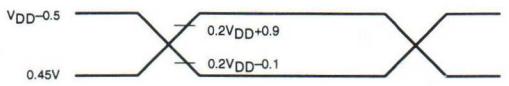
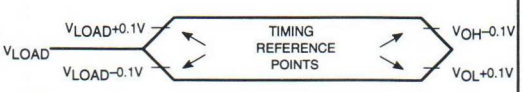



**Figure 6. Shift Register Mode Timing**




NOTE:
AC INPUTS DURING TESTING ARE DRIVEN AT $V_{DD}-0.5$ FOR A LOGIC '1' AND 0.45V FOR A LOGIC '0'. TIMING MEASUREMENTS ARE MADE AT $V_{IH}$ MIN FOR A LOGIC '1' AND $V_{IL}$ MAX FOR A LOGIC '0'.

**Figure 7. AC Testing Input/Output**




NOTE:
FOR TIMING PURPOSES, A PORT IS NO LONGER FLOATING WHEN A 100MV CHANGE FROM LOAD VOLTAGE OCCURS, AND BEGINS TO FLOAT WHEN A 100mV CHANGE FROM THE LOADED $V_{OH}/V_{OL}$ LEVEL OCCURS. $I_{OH}/I_{OL} \geq \pm$ 20mA.

**Figure 8. Float Waveform**

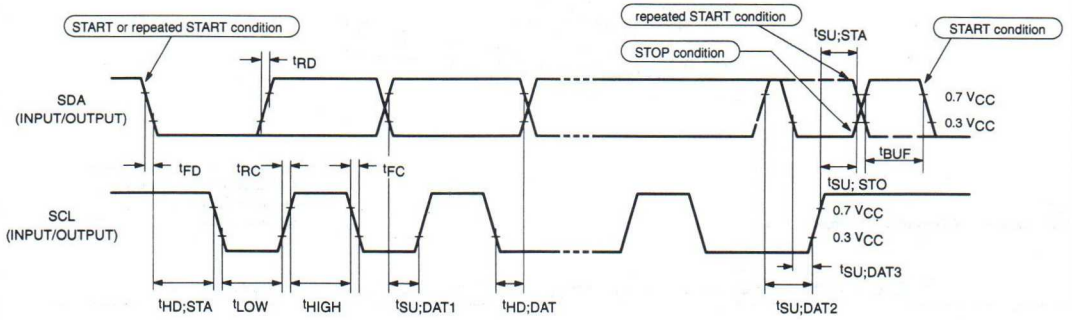
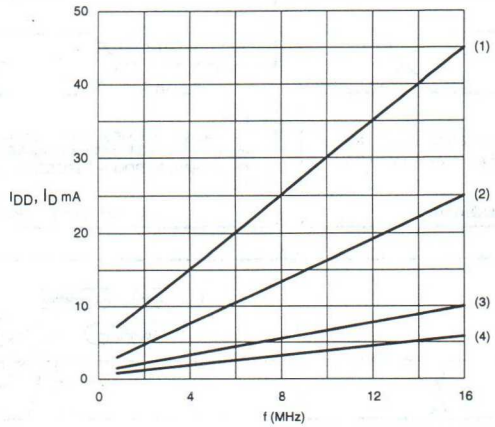


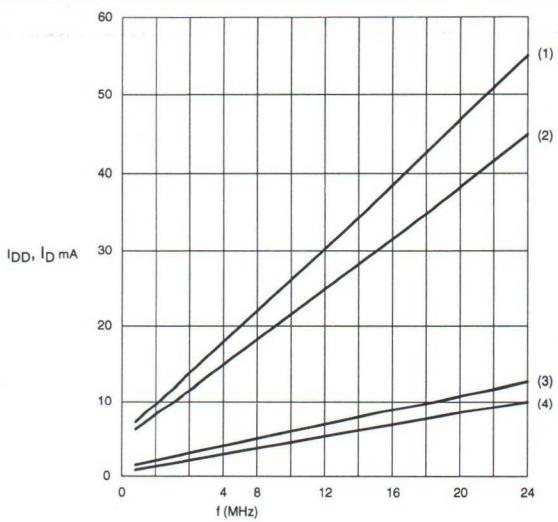

**Figure 9. Timing SIO1 ($I^2C$) Interface**




(1) Maximum operating mode; $V_{DD}$ = 6V
(2) Maximum operating mode; $V_{DD}$ = 4V
(3) Maximum idle mode; $V_{DD}$ = 6V
(4) Maximum idle mode; $V_{DD}$ = 4V

NOTE:
These values are valid only within the frequency specifications of the device under test.

**Figure 10. 16MHz Version Supply Current ($I_{DD}$) as a Function of Frequency at XTAL1 ($f_{OSC}$)**




(1) Maximum operating mode; $V_{DD}$ = 5.5V
(2) Maximum operating mode; $V_{DD}$ = 4.5V
(3) Maximum idle mode; $V_{DD}$ = 5.5V
(4) Maximum idle mode; $V_{DD}$ = 4.5V

NOTE:
These values are valid only within the frequency specifications of the device under test.

**Figure 11. 24MHz Version Supply Current ($I_{DD}$) as a Function of Frequency at XTAL1 ($f_{OSC}$)**

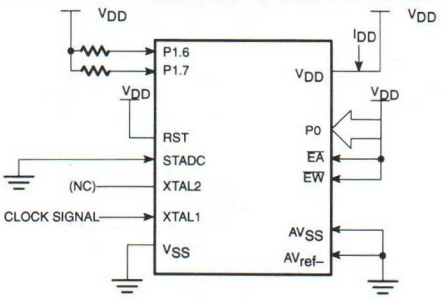



**Figure 12. $I_{DD}$ Test Condition, Active Mode**
**All other pins are disconnected[1]**

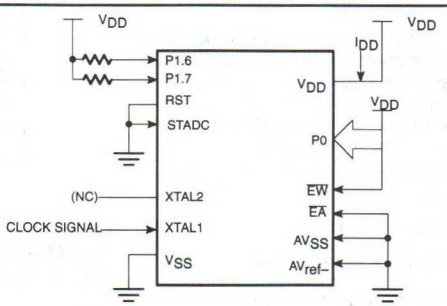



**Figure 13. $I_{DD}$ Test Condition, Idle Mode**
**All other pins are disconnected[2]**

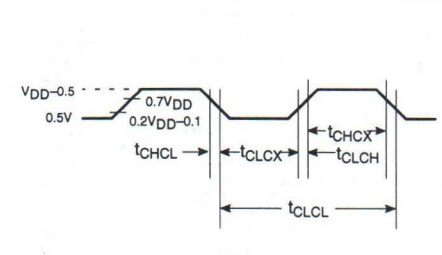



**Figure 14. Clock Signal Waveform for $I_{DD}$ Tests in Active and Idle Modes $t_{CLCH} = t_{CHCL} = 5ns$**

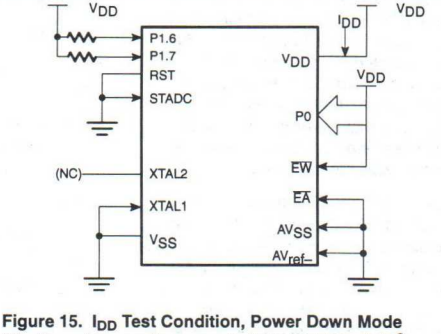



**Figure 15. $I_{DD}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{DD}$ = 2V to 5.5V[3]**

**NOTES:**

1. Active Mode:
   a. The following pins must be forced to $V_{DD}$: $\overline{EA}$, RST, Port 0, and $\overline{EW}$.
   b. The following pins must be forced to $V_{SS}$: STADC, $AV_{ss}$, and $AV_{ref-}$.
   c. Ports 1.6 and 1.7 should be connected to $V_{DD}$ through resistors of sufficiently high value such that the sink current into these pins cannot exceed the $I_{OL1}$ spec of these pins.
   d. The following pins must be disconnected: XTAL2 and all pins not specified above.
2. Idle Mode:
   a. The following pins must be forced to $V_{DD}$: Port 0 and $\overline{EW}$.
   b. The following pins must be forced to $V_{SS}$: RST, STADC, $AV_{ss}$,, $AV_{ref-}$, and $\overline{EA}$.
   c. Ports 1.6 and 1.7 should be connected to $V_{DD}$ through resistors of sufficiently high value such that the sink current into these pins cannot exceed the $I_{OL1}$ spec of these pins. These pins must not have logic 0 written to them prior to this measurement.
   d. The following pins must be disconnected: XTAL2 and all pins not specified above.
3. Power Down Mode:
   a. The following pins must be forced to $V_{DD}$: Port 0 and $\overline{EW}$.
   b. The following pins must be forced to $V_{SS}$: RST, STADC, XTAL1, $AV_{ss}$,, $AV_{ref-}$, and $\overline{EA}$.
   c. Ports 1.6 and 1.7 should be connected to $V_{DD}$ through resistors of sufficiently high value such that the sink current into these pins cannot exceed the $I_{OL1}$ spec of these pins. These pins must not have logic 0 written to them prior to this measurement.
   d. The following pins must be disconnected: XTAL2 and all pins not specified above.

I²C BUS

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# Single-chip 8-bit microcontroller — 87C552

*Single-chip 8-bit microcontroller with 10-bit A/D, capture/compare timer, high-speed outputs, PWM*

## DESCRIPTION

The 87C552 Single-Chip 8-Bit Microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 87C552 has the same instruction set as the 80C51. Three versions of the derivative exist:

- 83C552—8k bytes mask programmable ROM
- 80C552—ROMless version of the 83C552
- 87C552—8k bytes EPROM

The 87C552 contains a 8k × 8 a volatile 256 × 8 read/write data memory, five 8-bit I/O ports, one 8-bit input port, two 16-bit timer/event counters (identical to the timers of the 80C51), an additional 16-bit timer coupled to capture and compare latches, a 15-source, two-priority-level, nested interrupt structure, an 8-input ADC, a dual DAC pulse width modulated interface, two serial interfaces (UART and I²C-bus), a "watchdog" timer and on-chip oscillator and timing circuits. For systems that require extra capability, the 87C552 can be expanded using standard TTL compatible memories and logic.

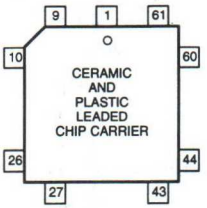
In addition, the 87C552 has two software selectable modes of power reduction—idle mode and power-down mode. The idle mode freezes the CPU while allowing the RAM, timers, serial ports, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte, and 17 three-byte. With a 16MHz (24MHz) crystal, 58% of the instructions are executed in 0.75μs (0.5μs) and 40% in 1.5μs (1μs). Multiply and divide instructions require 3μs (2μs).

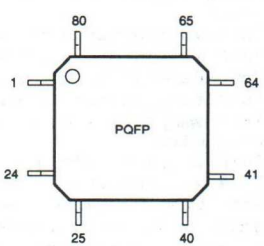
## FEATURES

- 80C51 central processing unit
- 8k × 8 EPROM expandable externally to 64k bytes
- An additional 16-bit timer/counter coupled to four capture registers and three compare registers
- Two standard 16-bit timer/counters
- 256 × 8 RAM, expandable externally to 64k bytes
- Capable of producing eight synchronized, timed outputs
- A 10-bit ADC with eight multiplexed analog inputs
- Two 8-bit resolution, pulse width modulation outputs
- Five 8-bit I/O ports plus one 8-bit input port shared with analog inputs
- I²C-bus serial I/O port with byte oriented master and slave functions
- Full-duplex UART compatible with the standard 80C51
- On-chip watchdog timer
- Three speed ranges:
  - 16MHz
- Extended temperature ranges
- OTP package available

## PIN CONFIGURATIONS

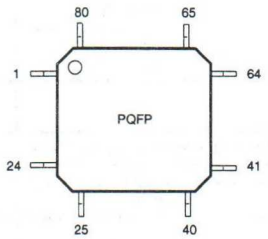



| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P5.0/ADC0 | 35 | XTAL1 |
| 2 | $V_{DD}$ | 36 | $V_{SS}$ |
| 3 | STADC | 37 | $V_{SS}$ |
| 4 | $\overline{PWM0}$ | 38 | NC |
| 5 | $\overline{PWM1}$ | 39 | P2.0/A08 |
| 6 | $\overline{EW}$ | 40 | P2.1/A09 |
| 7 | P4.0/CMSR0 | 41 | P2.2/A10 |
| 8 | P4.1/CMSR1 | 42 | P2.3/A11 |
| 9 | P4.2/CMSR2 | 43 | P2.4/A12 |
| 10 | P4.3/CMSR3 | 44 | P2.5/A13 |
| 11 | P4.4/CMSR4 | 45 | P2.6/A14 |
| 12 | P4.5/CMSR5 | 46 | P2.7/A15 |
| 13 | P4.6/CMT0 | 47 | $\overline{PSEN}$ |
| 14 | P4.7/CMT1 | 48 | ALE/$\overline{PROG}$ |
| 15 | RST | 49 | $\overline{EA}$/$V_{PP}$ |
| 16 | P1.0/CT0I | 50 | P0.7/AD7 |
| 17 | P1.1/CT1I | 51 | P0.6/AD6 |
| 18 | P1.2/CT2I | 52 | P0.5/AD5 |
| 19 | P1.3/CT3I | 53 | P0.4/AD4 |
| 20 | P1.4/T2 | 54 | P0.3/AD3 |
| 21 | P1.5/RT2 | 55 | P0.2/AD2 |
| 22 | P1.6/SCL | 56 | P0.1/AD1 |
| 23 | P1.7/SDA | 57 | P0.0/AD0 |
| 24 | P3.0/RxD | 58 | AVref– |
| 25 | P3.1/TxD | 59 | AVref+ |
| 26 | P3.2/$\overline{INT0}$ | 60 | $AV_{SS}$ |
| 27 | P3.3/$\overline{INT1}$ | 61 | $AV_{DD}$ |
| 28 | P3.4/T0 | 62 | P5.7/ADC7 |
| 29 | P3.5/T1 | 63 | P5.6/ADC6 |
| 30 | P3.6/$\overline{WR}$ | 64 | P5.5/ADC5 |
| 31 | P3.7/$\overline{RD}$ | 65 | P5.4/ADC4 |
| 32 | NC | 66 | P5.3/ADC3 |
| 33 | NC | 67 | P5.2/ADC2 |
| 34 | XTAL2 | 68 | P5.1/ADC1 |




SU00208

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS

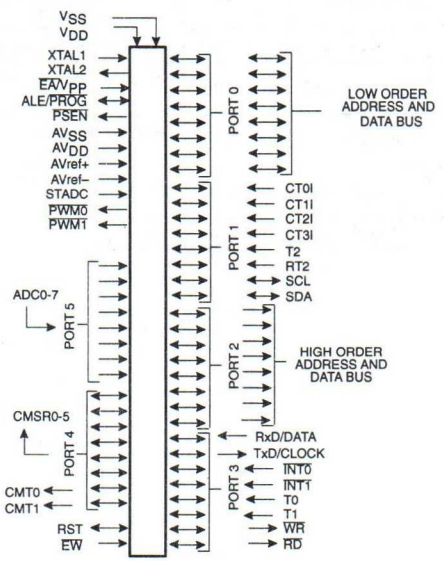



| Pin | Function | Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|---|---|
| 1 | P4.1/CMSR1 | 21 | NC | 41 | P2.3/A11 | 61 | $AV_{SS}$ |
| 2 | P4.2/CMSR2 | 22 | NC | 42 | P2.4/A12 | 62 | NC |
| 3 | NC | 23 | P3.3/$\overline{INT1}$ | 43 | NC | 63 | $AV_{DD}$ |
| 4 | P4.3/CMSR3 | 24 | P3.4/T0 | 44 | NC | 64 | P5.7/ADC7 |
| 5 | P4.4/CMSR4 | 25 | P3.5/T1 | 45 | P2.5/A13 | 65 | P5.6/ADC6 |
| 6 | P4.5/CMSR5 | 26 | P3.6/$\overline{WR}$ | 46 | P2.6/A14 | 66 | P5.5/ADC5 |
| 7 | P4.6/CMT0 | 27 | P3.7/$\overline{RD}$ | 47 | P2.7/A15 | 67 | P5.4/ADC4 |
| 8 | P4.7/CMT1 | 28 | NC | 48 | $\overline{PSEN}$ | 68 | P5.3/ADC3 |
| 9 | RST | 29 | NC | 49 | ALE/$\overline{PROG}$ | 69 | P5.2/ADC2 |
| 10 | P1.0/CT0I | 30 | NC | 50 | $\overline{EA}/V_{PP}$ | 70 | P5.1/ADC1 |
| 11 | P1.1/CT1I | 31 | XTAL2 | 51 | P0.7/AD7 | 71 | P5.0/ADC0 |
| 12 | P1.2/CT2I | 32 | XTAL1 | 52 | P0.6/AD6 | 72 | $V_{DD}$ |
| 13 | P1.3/CT3I | 33 | IC | 53 | P0.5/AD5 | 73 | IC |
| 14 | P1.4/T2 | 34 | $V_{SS}$ | 54 | P0.4/AD4 | 74 | STADC |
| 15 | P1.5/RT2 | 35 | $V_{SS}$ | 55 | P0.3/AD3 | 75 | $\overline{PWM0}$ |
| 16 | P1.6/SCL | 36 | $V_{SS}$ | 56 | P0.2/AD2 | 76 | $\overline{PWM1}$ |
| 17 | P1.7/SDA | 37 | NC | 57 | P0.1/AD1 | 77 | $\overline{EW}$ |
| 18 | P3.0/RxD | 38 | P2.0/A08 | 58 | P0.0/AD0 | 78 | NC |
| 19 | P3.1/TxD | 39 | P2.1/A09 | 59 | AVref– | 79 | NC |
| 20 | P3.2/$\overline{INT0}$ | 40 | P2.2/A10 | 60 | AVref+ | 80 | P4.0/CMSR0 |

NC = Not Connected
IC = Internally Connected (do not use)

SU00209

## LOGIC SYMBOL




SU00210

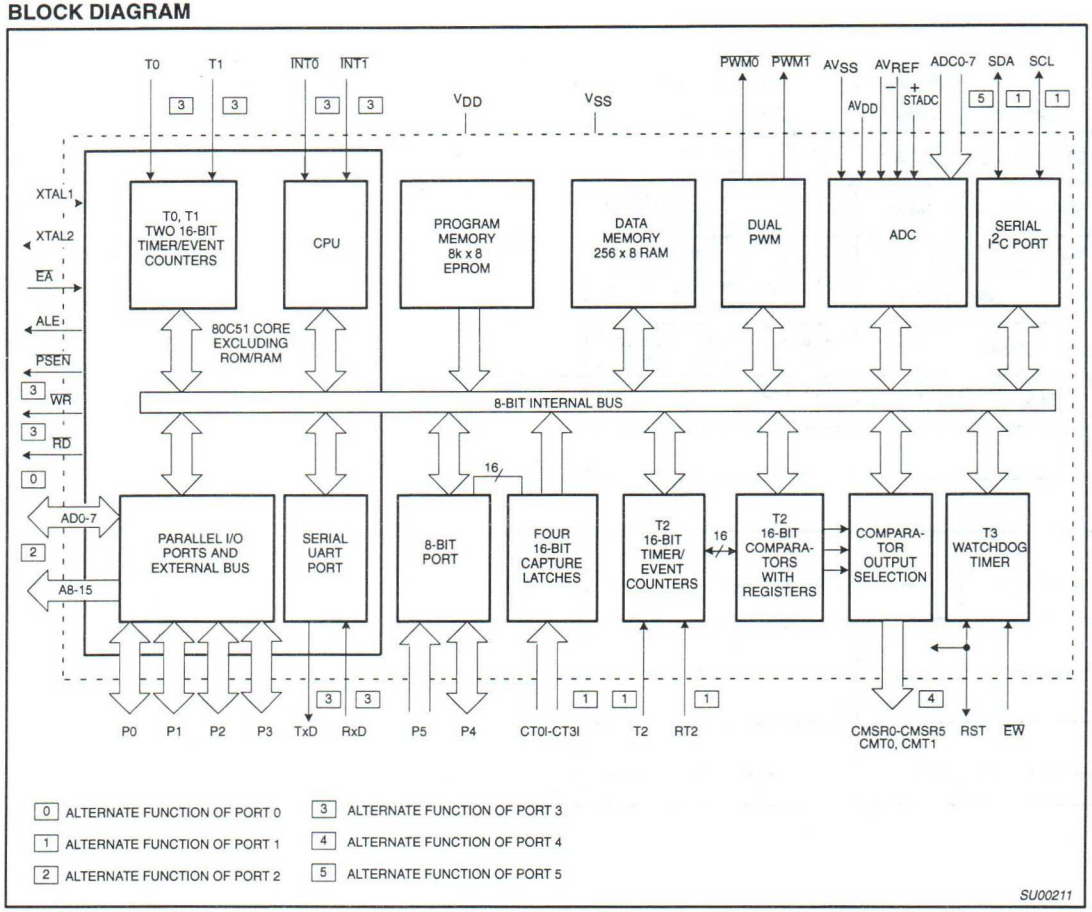
## ORDERING INFORMATION

| PHILIPS PART ORDER NUMBER PART MARKING | | NORTH AMERICA PHILIPS PART ORDER NUMBER | | DRAWING NUMBER | TEMPERATURE °C AND PACKAGE | FREQ MHz |
|---|---|---|---|---|---|---|
| ROMless | ROM | ROMless | ROM | | | |
| PCB80C552-5-16WP | PCB83C552-5WP/xxx | S80C552-4A68 | S83C552-4A68 | SOT188-3 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| | | | | | | |
| PCB80C552-5-16H | PCB83C552-5H/xxx | S80C552-4B | S83C552-4B | SOT318-2 | 0 to +70, Plastic Quad Flat Pack | 16 |
| PCF80C552-5-16WP | PCF83C552-5WP/xxx | S80C552-5A68 | S83C552-5A68 | SOT188-3 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| | | | | | | |
| PCF80C552-5-16H | PCF83C552-5H/xxx | S80C552-5B | S83C552-5B | SOT318-2 | –40 to +85, Plastic Quad Flat Pack | 16 |
| PCA80C552-5-16WP | PCA83C552-5WP/xxx | S80C552-6A68 | S83C552-6A68 | SOT188-3 | –40 to +125, Plastic Leaded Chip Carrier | 16 |
| PCA80C552-5-16H | PCA83C552-5H/xxx | S80C552-6B | S83C552-6B | SOT318-2 | –40 to +125, Plastic Quad Flat Pack | 16 |
| PCB80C552-5-24WP | PCB83C552-5WP/xxx | S80C552-AA68 | S83C552-AA68 | SOT188-3 | 0 to +70, Plastic Leaded Chip Carrier | 24 |
| PCB80C552-5-24H | PCB83C552-5H/xxx | S80C552-AB | S83C552-AB | SOT318-2 | 0 to +70, Plastic Quad Flat Pack | 24 |
| PCF80C552-5-24WP | PCF83C552-5WP/xxx | S80C552-BA68 | S83C552-BA68 | SOT188-3 | –40 to +85, Plastic Leaded Chip Carrier | 24 |
| PCF80C552-5-24H | PCF83C552-5H/xxx | S80C552-BB | S83C552-BB | SOT318-2 | –40 to +85, Plastic Quad Flat Pack | 24 |
| PCB80C552-5-30WP | PCB83C552-5WP/xxx | S80C552-CA68 | S83C552-CA68 | SOT188-3 | 0 to +70, Plastic Leaded Chip Carrier | 30 |
| PCB80C552-5-30H | PCB83C552-5H/xxx | S80C552-CB | S83C552-CB | SOT318-2 | 0 to +70, Plastic Quad Flat Pack | 30 |

**NOTE:**
1. xxx denotes the ROM code number.

| EPROM | DRAWING NUMBER | TEMPERATURE °C AND PACKAGE | FREQ MHz |
|---|---|---|---|
| S87C552-4A68 | SOT188-3 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| S87C552-4K68 | 1473A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 16 |
| S87C552-4BA | SOT318-2 | 0 to +70, Plastic Quad Flat Pack | 16 |
| S87C552-5A68 | SOT188-3 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| S87C552-5K68 | 1473A | –40 to +85, Ceramic Leaded Chip Carrier w/Window | 16 |
| S87C552-5BA | SOT318-2 | –40 to +85, Plastic Quad Flat Pack | 16 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

## BLOCK DIAGRAM



## PIN DESCRIPTION

| MNEMONIC | PIN NO. PLCC | PIN NO. QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|
| $V_{DD}$ | 2 | 72 | I | **Digital Power Supply:** +5V power supply pin during normal operation, idle and power-down mode. |
| STADC | 3 | 74 | I | **Start ADC Operation:** Input starting analog to digital conversion (ADC operation can also be started by software). |
| $\overline{PWM0}$ | 4 | 75 | O | **Pulse Width Modulation:** Output 0. |
| $\overline{PWM1}$ | 5 | 76 | O | **Pulse Width Modulation:** Output 1. |
| $\overline{EW}$ | 6 | 77 | I | **Enable Watchdog Timer:** Enable for T3 watchdog timer and disable power-down mode. |
| P0.0-P0.7 | 57-50 | 58-51 | I/O | **Port 0:** Port 0 is an 8-bit open-drain bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application it uses strong internal pull-ups when emitting 1s. Port 0 is also used to input the code byte during programming and to output the code byte during verification. |
| P1.0-P1.7 | 16-23 | 10-17 | I/O | **Port 1:** 8-bit I/O port. Alternate functions include: |
| | 16-21 | 10-15 | I/O | **(P1.0-P1.5):** Quasi-bidirectional port pins. |
| | 22-23 | 16-17 | I/O | **(P1.6, P1.7):** Open drain port pins. |
| | 16-19 | 10-13 | I | **CT0I-CT3I (P1.0-P1.3):** Capture timer input signals for timer T2. |
| | 20 | 14 | I | **T2 (P1.4):** T2 event input. |
| | 21 | 15 | I | **RT2 (P1.5):** T2 timer reset signal. Rising edge triggered. |
| | 22 | 16 | I/O | **SCL (P1.6):** Serial port clock line $I^2C$-bus. |
| | 23 | 17 | I/O | **SDA (P1.7):** Serial port data line $I^2C$-bus. |
| | | | | Port 1 is also used to input the lower order address byte during EPROM programming and verification. A0 is on P1.0, etc. |
| P2.0-P2.7 | 39-46 | 38-42, 45-47 | I/O | **Port 2:** 8-bit quasi-bidirectional I/O port. Alternate function: High-order address byte for external memory (A08-A15). Port 2 is also used to input the upper order address during EPROM programming and verification. A8 is on P2.0, A9 on P2.1, through A12 on P2.4. |
| P3.0-P3.7 | 24-31 | 18-20, 23-27 | I/O | **Port 3:** 8-bit quasi-bidirectional I/O port. Alternate functions include: |
| | 24 | 18 | | **RxD(P3.0):** Serial input port. |
| | 25 | 19 | | **TxD (P3.1):** Serial output port. |
| | 26 | 20 | | **$\overline{INT0}$ (P3.2):** External interrupt. |
| | 27 | 23 | | **$\overline{INT1}$ (P3.3):** External interrupt. |
| | 28 | 24 | | **T0 (P3.4):** Timer 0 external input. |
| | 29 | 25 | | **T1 (P3.5):** Timer 1 external input. |
| | 30 | 26 | | **$\overline{WR}$ (P3.6):** External data memory write strobe. |
| | 31 | 27 | | **$\overline{RD}$ (P3.7):** External data memory read strobe. |
| P4.0-P4.7 | 7-14 | 80, 1-2 4-8 | I/O | **Port 4:** 8-bit quasi-bidirectional I/O port. Alternate functions include: |
| | 7-12 | 80, 1-2 4-6 | O | **CMSR0-CMSR5 (P4.0-P4.5):** Timer T2 compare and set/reset outputs on a match with timer T2. |
| | 13, 14 | 7, 8 | O | **CMT0, CMT1 (P4.6, P4.7):** Timer T2 compare and toggle outputs on a match with timer T2. |
| P5.0-P5.7 | 68-62, 1 | 71-64, | I | **Port 5:** 8-bit input port. **ADC0-ADC7 (P5.0-P5.7):** Alternate function: Eight input channels to ADC. |
| RST | 15 | 9 | I/O | **Reset:** Input to reset the 87C552. It also provides a reset pulse as output when timer T3 overflows. |
| XTAL1 | 35 | 32 | I | **Crystal Input 1:** Input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external clock signal when an external oscillator is used. |
| XTAL2 | 34 | 31 | O | **Crystal Input 2:** Output of the inverting amplifier that forms the oscillator. Left open-circuit when an external clock is used. |
| $V_{SS}$ | 36, 37 | 34-36 | I | **Digital ground.** |
| $\overline{PSEN}$ | 47 | 48 | O | **Program Store Enable:** Active-low read strobe to external program memory. |

## PIN DESCRIPTION (Continued)

| MNEMONIC | PIN NO. PLCC | PIN NO. QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|
| ALE/$\overline{PROG}$ | 48 | 49 | O | **Address Latch Enable:** Latches the low byte of the address during accesses to external memory. It is activated every six oscillator periods. During an external data memory access, one ALE pulse is skipped. ALE can drive up to eight LS TTL inputs and handles CMOS inputs without an external pull-up. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. |
| $\overline{EA}$/$V_{PP}$ | 49 | 50 | I | **External Access:** When $\overline{EA}$ is held at TTL level high, the CPU executes out of the internal program ROM provided the program counter is less than 8192. When $\overline{EA}$ is held at TTL low level, the CPU executes out of external program memory. $\overline{EA}$ is not allowed to float. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. |
| $AV_{REF-}$ | 58 | 59 | I | **Analog to Digital Conversion Reference Resistor:** Low-end. |
| $AV_{REF+}$ | 59 | 60 | I | **Analog to Digital Conversion Reference Resistor:** High-end. |
| $AV_{SS}$ | 60 | 61 | I | **Analog Ground** |
| $AV_{DD}$ | 61 | 63 | I | **Analog Power Supply** |

**NOTE:**
1. To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher or lower than $V_{DD}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the logic symbol.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{DD}$ and RST must come up at the same time for a proper start-up.

## IDLE MODE

In the idle mode, the CPU puts itself to sleep while some of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. The control bits for the reduced power modes are in the special function register PCON. Table 1 shows the state of the I/O ports during low current operating modes.

**Table 1. External Pin Status During Idle and Power-Down Modes**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 | PORT 4 | PWM0/ PWM1 |
|---|---|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data | Data | High |
| Idle | External | 1 | 1 | Float | Data | Address | Data | Data | High |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data | Data | High |
| Power-down | External | 0 | 0 | Float | Data | Data | Data | Data | High |

### Serial Control Register (S1CON) – See Table 2

**S1CON (D8H)**

| CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |
|---|---|---|---|---|---|---|---|

**Bits CR0, CR1 and CR2 determine the serial clock frequency that is generated in the master mode of operation.**

### Table 2. Serial Clock Rates

| CR2 | CR1 | CR0 | BIT FREQUENCY (kHz) AT $f_{OSC}$ | | | $f_{OSC}$ DIVIDED BY |
|---|---|---|---|---|---|---|
| | | | 6MHz | 12MHz | 16MHz | |
| 0 | 0 | 0 | 23 | 47 | 62.5 | 256 |
| 0 | 0 | 1 | 27 | 54 | 71 | 224 |
| 0 | 1 | 0 | 31.25 | 62.5 | 83.3 | 192 |
| 0 | 1 | 1 | 37 | 75 | 100 | 160 |
| 1 | 0 | 0 | 6.25 | 12.5 | 17 | 960 |
| 1 | 0 | 1 | 50 | 100 | 133 [1] | 120 |
| 1 | 1 | 0 | 100 | 200 | 267 [1] | 60 |
| 1 | 1 | 1 | 0.25 < 62.5<br>0 to 225 | 0.5 < 62.5<br>0 to 224 | 0.67 < 56<br>0 to 223 | 96 × (256 – (reload value Timer 1))<br>Timer 1 in Mode 2. |

**NOTE:**
1. These frequencies exceed the upper limit of 100kHz of the $I^2C$-bus specification and cannot be used in an $I^2C$-bus application.

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ to $V_{SS}$ | –0.5 to +13 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Input, output DC current on any single I/O pin | 5.0 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.0 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DEVICE SPECIFICATIONS

| TYPE | SUPPLY VOLTAGE (V) | | FREQUENCY (MHz) | | TEMPERATURE RANGE (°C) |
|---|---|---|---|---|---|
| | MIN | MAX | MIN | MAX | |
| P87C552-4 | 4.5 | 5.5 | 3.5 | 16 | 0 to +70 |
| P87C552-5 | 4.5 | 5.5 | 3.5 | 16 | –40 to +85 |

## DC ELECTRICAL CHARACTERISTICS

$V_{SS}$, $AV_{SS}$ = 0V

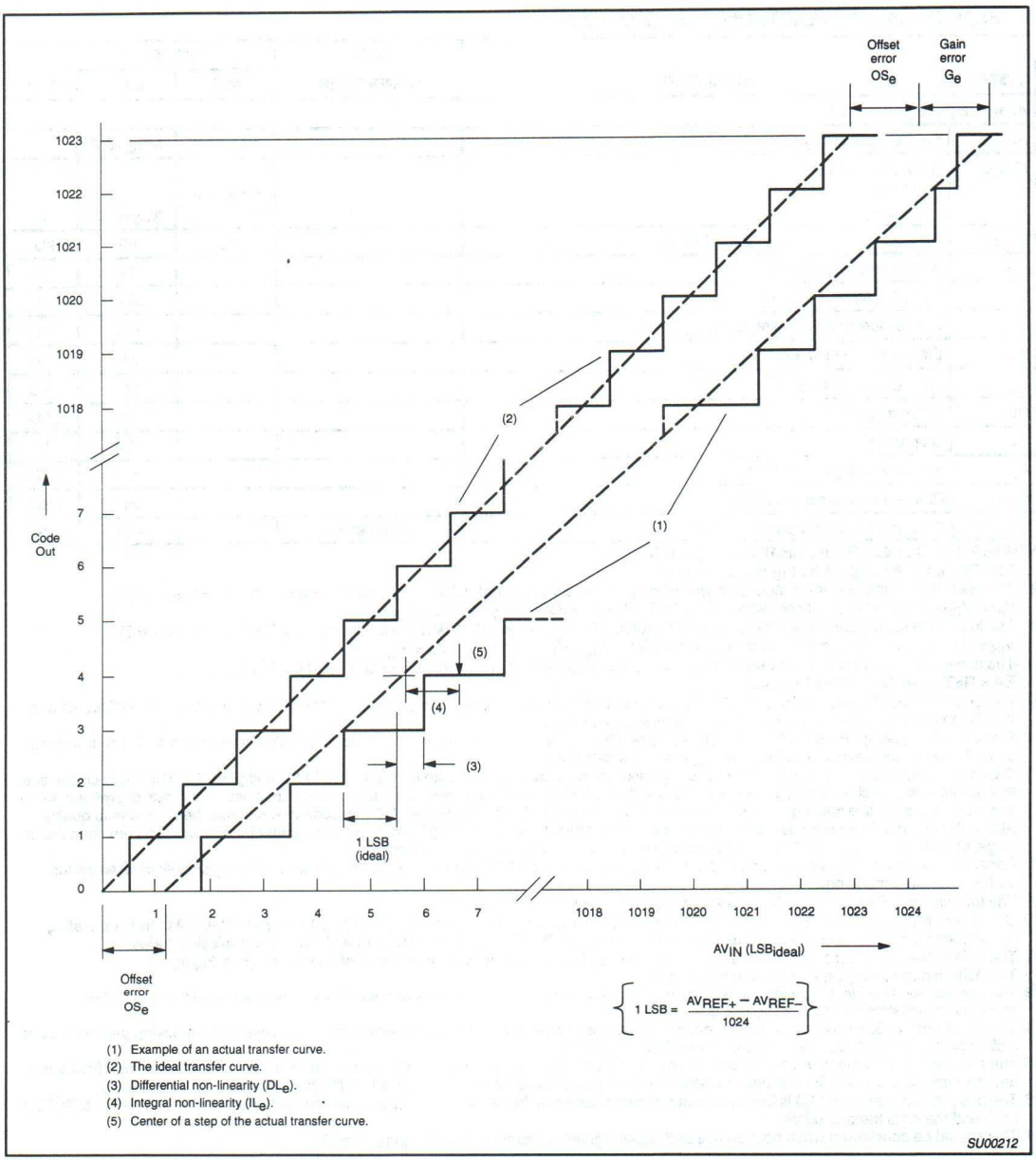
| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|
| $I_{DD}$ | Supply current operating: PCA8XC552-5-16 | See notes 1 and 2 $f_{OSC}$ = 16MHz | | 40 | mA |
| $I_{ID}$ | Idle mode: 87C552 | See notes 1 and 3 $f_{OSC}$ = 16MHz | | 7 | mA |
| $I_{PD}$ | Power-down current: 87C552 | See notes 1 and 4; 2V < $V_{PD}$ < $V_{DD}$ max | | 50 | µA |
| **Inputs** | | | | | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$, P1.6, P1.7 | | –0.5 | $0.2V_{DD}$–0.1 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | | –0.5 | $0.2V_{DD}$–0.3 | V |
| $V_{IL2}$ | Input low voltage to P1.6/SCL, P1.7/SDA[5] | | –0.5 | $0.3V_{DD}$ | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST | | $0.2V_{DD}$+0.9 | $V_{DD}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | | $0.7V_{DD}$ | $V_{DD}$+0.5 | V |
| $V_{IH2}$ | Input high voltage, P1.6/SCL, P1.7/SDA[5] | | $0.7V_{DD}$ | 6.0 | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3, 4, except P1.6, P1.7 | $V_{IN}$ = 0.45V | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3, 4, except P1.6, P1.7 | See note 6 | | –650 | µA |
| $\pm I_{IL1}$ | Input leakage current, port 0, $\overline{EA}$, STADC, $\overline{EW}$ | 0.45V < $V_I$ < $V_{DD}$ | | 10 | µA |
| $\pm I_{IL2}$ | Input leakage current, P1.6/SCL, P1.7/SDA | 0V < $V_I$ < 6V 0V < $V_{DD}$ < 5.5V | | 10 | µA |
| $\pm I_{IL3}$ | Input leakage current, port 5 | 0.45V < $V_I$ < $V_{DD}$ | | 1 | µA |
| **Outputs** | | | | | |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3, 4, except P1.6, P1.7 | $I_{OL}$ = 1.6mA[7] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$ | $I_{OL}$ = 3.2mA[7] | | 0.45 | V |
| $V_{OL2}$ | Output low voltage, P1.6/SCL, P1.7/SDA | $I_{OL}$ = 3.0mA[7] | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3, 4, except P1.6/SCL, P1.7/SDA | $-I_{OH}$ = 60µA<br>$-I_{OH}$ = 25µA<br>$-I_{OH}$ = 10µA | 2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | V<br>V<br>V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$)[8] | $-I_{OH}$ = 400µA<br>$-I_{OH}$ = 150µA<br>$-I_{OH}$ = 40µA | 2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | V<br>V<br>V |
| $V_{OH2}$ | Output high voltage (RST) | $-I_{OH}$ = 400µA<br>$-I_{OH}$ = 120µA | 2.4<br>$0.8V_{DD}$ | | V<br>V |
| $R_{RST}$ | Internal reset pull-down resistor | | 50 | 150 | kΩ |
| $C_{IO}$ | Pin capacitance | Test freq = 1MHz, $T_{amb}$ = 25°C | | 10 | pF |
| **Analog Inputs** | | | | | |
| $AV_{DD}$ | Analog supply voltage: 87C552[9] | $AV_{DD}$ = $V_{DD}$±0.2V | 4.5 | 5.5 | V |
| $AI_{DD}$ | Analog supply current: operating: | Port 5 = 0 to $AV_{DD}$ | | 1.2 | mA |
| $AI_{ID}$ | Idle mode: 87C552 | | | 50 | µA |
| $AI_{PD}$ | Power-down mode: 87C552 | 2V < $AV_{PD}$ < $AV_{DD}$ max | | 50 | µA |

## DC ELECTRICAL CHARACTERISTICS (Continued)

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|
| **Analog Inputs (Continued)** | | | | | |
| $AV_{IN}$ | Analog input voltage | | $AV_{SS}$–0.2 | $AV_{DD}$+0.2 | V |
| $AV_{REF}$ | Reference voltage:<br>$AV_{REF-}$<br>$AV_{REF+}$ | | $AV_{SS}$–0.2 | <br>$AV_{DD}$+0.2 | <br>V<br>V |
| $R_{REF}$ | Resistance between $AV_{REF+}$ and $AV_{REF-}$ | | 10 | 50 | kΩ |
| $C_{IA}$ | Analog input capacitance | | | 15 | pF |
| $t_{ADS}$ | Sampling time | | | $8t_{CY}$ | µs |
| $t_{ADC}$ | Conversion time (including sampling time) | | | $50t_{CY}$ | µs |
| $DL_e$ | Differential non-linearity[10, 11, 12] | | | ±1 | LSB |
| $IL_e$ | Integral non-linearity[10, 13] | | | ±2 | LSB |
| $OS_e$ | Offset error[10, 14] | | | ±2 | LSB |
| $G_e$ | Gain error[10, 15] | | | ±0.4 | % |
| $A_e$ | Absolute voltage error[10, 16] | | | ±3 | LSB |
| $M_{CTC}$ | Channel to channel matching | | | ±1 | LSB |
| $C_t$ | Crosstalk between inputs of port 5[17] | 0–100kHz | | –60 | dB |

**NOTES FOR DC ELECTRICAL CHARACTERISTICS:**

1. See Figures 10 through 15 for $I_{DD}$ test conditions.
2. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 10ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5V$; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = $\overline{EW}$ = $V_{DD}$; STADC = $V_{SS}$.
3. The idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 10ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5V$; XTAL2 not connected; Port 0 = $\overline{EW}$ = $V_{DD}$; $\overline{EA}$ = RST = STADC = $V_{SS}$.
4. The power-down current is measured with all output pins disconnected; XTAL2 not connected; Port 0 = $\overline{EW}$ = $V_{DD}$; $\overline{EA}$ = RST = STADC = XTAL1 = $V_{SS}$.
5. The input threshold voltage of P1.6 and P1.7 (SIO1) meets the $I^2C$ specification, so an input voltage below 1.5V will be recognized as a logic 0 while an input voltage above 3.0V will be recognized as a logic 1.
6. Pins of ports 1 (except P1.6, P1.7), 2, 3, and 4 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
7. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
8. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{DD}$ specification when the address bits are stabilizing.
9. The following condition must not be exceeded: $V_{DD} - 0.2V < AV_{DD} < V_{DD} + 0.2V$.
10. Conditions: $AV_{REF-}$ = 0V; $AV_{DD}$ = 5.0V. Measurement by continuous conversion of $AV_{IN}$ = –20mV to 5.12V in steps of 0.5mV, derivating parameters from collected conversion results of ADC. $AV_{REF+}$ (87C552) = 4.977V. ADC is monotonic with no missing codes.
11. The differential non-linearity ($DL_e$) is the difference between the actual step width and the ideal step width. (See Figure 1.)
12. The ADC is monotonic; there are no missing codes.
13. The integral non-linearity ($IL_e$) is the peak difference between the center of the steps of the actual and the ideal transfer curve after appropriate adjustment of gain and offset error. (See Figure 1.)
14. The offset error ($OS_e$) is the absolute difference between the straight line which fits the actual transfer curve (after removing gain error), and a straight line which fits the ideal transfer curve. (See Figure 1.)
15. The gain error ($G_e$) is the relative difference in percent between the straight line fitting the actual transfer curve (after removing offset error), and the straight line which fits the ideal transfer curve. Gain error is constant at every point on the transfer curve. (See Figure 1.)
16. The absolute voltage error ($A_e$) is the maximum difference between the center of the steps of the actual transfer curve of the non-calibrated ADC and the ideal transfer curve.
17. This should be considered when both analog and digital signals are simultaneously input to port 5.




(1) Example of an actual transfer curve.
(2) The ideal transfer curve.
(3) Differential non-linearity ($DL_e$).
(4) Integral non-linearity ($IL_e$).
(5) Center of a step of the actual transfer curve.

**Figure 1. ADC Conversion Characteristic**

## AC ELECTRICAL CHARACTERISTICS[1, 2]

| SYMBOL | FIGURE | PARAMETER | 12MHz CLOCK | | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 2 | Oscillator frequency | | | | | 3.5 | 16 | MHz |
| $t_{LHLL}$ | 2 | ALE pulse width | 127 | | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 2 | Address valid to ALE low | 28 | | 8 | | $t_{CLCL}-55$ | | ns |
| $t_{LLAX}$ | 2 | Address hold after ALE low | 48 | | 28 | | $t_{CLCL}-35$ | | ns |
| $t_{LLIV}$ | 2 | ALE low to valid instruction in | | 234 | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 2 | ALE low to $\overline{PSEN}$ low | 43 | | 23 | | $t_{CLCL}-40$ | | ns |
| $t_{PLPH}$ | 2 | $\overline{PSEN}$ pulse width | 205 | | 143 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 2 | $\overline{PSEN}$ low to valid instruction in | | 145 | | 83 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 2 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 2 | Input instruction float after $\overline{PSEN}$ | | 59 | | 38 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 2 | Address to valid instruction in | | 312 | | 208 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 2 | $\overline{PSEN}$ low to address float | | 10 | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | | | |
| $t_{AVLL}$ | 3, 4 | Address valid to ALE low | 43 | | 23 | | $t_{CLCL}-40$ | | ns |
| $t_{RLRH}$ | 3 | $\overline{RD}$ pulse width | 400 | | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 3 | $\overline{WR}$ pulse width | 400 | | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 3 | $\overline{RD}$ low to valid data in | | 252 | | 148 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 3 | Data hold after $\overline{RD}$ | 0 | | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 3 | Data float after $\overline{RD}$ | | 97 | | 55 | | $2t_{CLCL}-70$ | ns |
| $t_{LLDV}$ | 3 | ALE low to valid data in | | 517 | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 3 | Address to valid data in | | 585 | | 398 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 3, 4 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 200 | 300 | 138 | 238 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 3, 4 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 203 | | 120 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 4 | Data valid to $\overline{WR}$ transition | 23 | | 3 | | $t_{CLCL}-60$ | | ns |
| $t_{DW}$ | 4 | Data before $\overline{WR}$ | 433 | | 288 | | $7t_{CLCL}-150$ | | ns |
| $t_{WHQX}$ | 4 | Data hold after $\overline{WR}$ | 33 | | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{RLAZ}$ | 4 | $\overline{RD}$ low to address float | | 0 | | 0 | | 0 | ns |
| $t_{WHLH}$ | 3, 4 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 43 | 123 | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **External Clock** | | | | | | | | | |
| $t_{CHCX}$ | 5 | High time[3] | 20 | | 20 | | 20 | | ns |
| $t_{CLCX}$ | 5 | Low time[3] | 20 | | 20 | | 20 | | ns |
| $t_{CLCH}$ | 5 | Rise time[3] | | 20 | | 20 | | 20 | ns |
| $t_{CHCL}$ | 5 | Fall time[3] | | 20 | | 20 | | 20 | ns |
| **Serial Timing – Shift Register Mode[4]** (Test Conditions: $T_{amb}$ = 0°C to +70°C; $V_{SS}$ = 0V; Load Capaciatnce = 80pF) | | | | | | | | | |
| $t_{XLXL}$ | 6 | Serial port clock cycle time | 1.0 | | 0.75 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 6 | Output data setup to clock rising edge | 700 | | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 6 | Output data hold after clock rising edge | 50 | | 8 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 6 | Input data hold after clock rising edge | 0 | | 0 | | 0 | | ns |
| $t_{XHDV}$ | 6 | Clock rising edge to input data valid | | 700 | | 492 | | $10t_{CLCL}-133$ | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. $t_{CLCL} = 1/f_{OSC}$ = one oscillator clock period.
   $t_{CLCL}$ = 83.3ns at $f_{OSC}$ = 12MHz.
   $t_{CLCL}$ = 62.5ns at $f_{OSC}$ = 16MHz.
4. These values are characterized but not 100% production tested.

## AC ELECTRICAL CHARACTERISTICS (Continued)

| SYMBOL | PARAMETER | INPUT | OUTPUT |
|---|---|---|---|
| **I²C Interface (Refer to Figure 9)[5]** | | | |
| $t_{HD;STA}$ | START condition hold time | ≥ 14 $t_{CLCL}$ | > 4.0µs [1] |
| $t_{LOW}$ | SCL low time | ≥ 16 $t_{CLCL}$ | > 4.7µs [1] |
| $t_{HIGH}$ | SCL high time | ≥ 14 $t_{CLCL}$ | > 4.0µs [1] |
| $t_{RC}$ | SCL rise time | ≤ 1µs | – [2] |
| $t_{FC}$ | SCL fall time | ≤ 0.3µs | < 0.3µs [3] |
| $t_{SU;DAT1}$ | Data set-up time | ≥ 250ns | > 20 $t_{CLCL}$ – $t_{RD}$ |
| $t_{SU;DAT2}$ | SDA set-up time (before rep. START cond.) | ≥ 250ns | > 1µs [1] |
| $t_{SU;DAT3}$ | SDA set-up time (before STOP cond.) | ≥ 250ns | > 8 $t_{CLCL}$ |
| $t_{HD;DAT}$ | Data hold time | ≥ 0ns | > 8 $t_{CLCL}$ – $t_{FC}$ |
| $t_{SU;STA}$ | Repeated START set-up time | ≥ 14 $t_{CLCL}$ | > 4.7µs [1] |
| $t_{SU;STO}$ | STOP condition set-up time | ≥ 14 $t_{CLCL}$ | > 4.0µs [1] |
| $t_{BUF}$ | Bus free time | ≥ 14 $t_{CLCL}$ | > 4.7µs [1] |
| $t_{RD}$ | SDA rise time | ≤ 1µs | – [2] |
| $t_{FD}$ | SDA fall time | ≤ 0.3µs | < 0.3µs [3] |

**NOTES:**
1. At 100 kbit/s. At other bit rates this value is inversely proportional to the bit-rate of 100 kbit/s.
2. Determined by the external bus-line capacitance and the external bus-line pull-resistor, this must be < 1µs.
3. Spikes on the SDA and SCL lines with a duration of less than 3 $t_{CLCL}$ will be filtered out. Maximum capacitance on bus-lines SDA and SCL = 400pF.
4. $t_{CLCL} = 1/f_{OSC}$ = one oscillator clock period at pin XTAL1. For 62ns (42s) < $t_{CLCL}$ < 285ns (16MHz (24Hz) > $f_{OSC}$ > 3.5MHz) the SI01 interface meets the I²C-bus specification for bit-rates up to 100 kbit/s.
5. These values are guaranteed but not 100% production tested.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{\text{PSEN}}$
Q – Output data
R – $\overline{\text{RD}}$ signal
t – Time
V – Valid
W – $\overline{\text{WR}}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{\text{PSEN}}$ low.

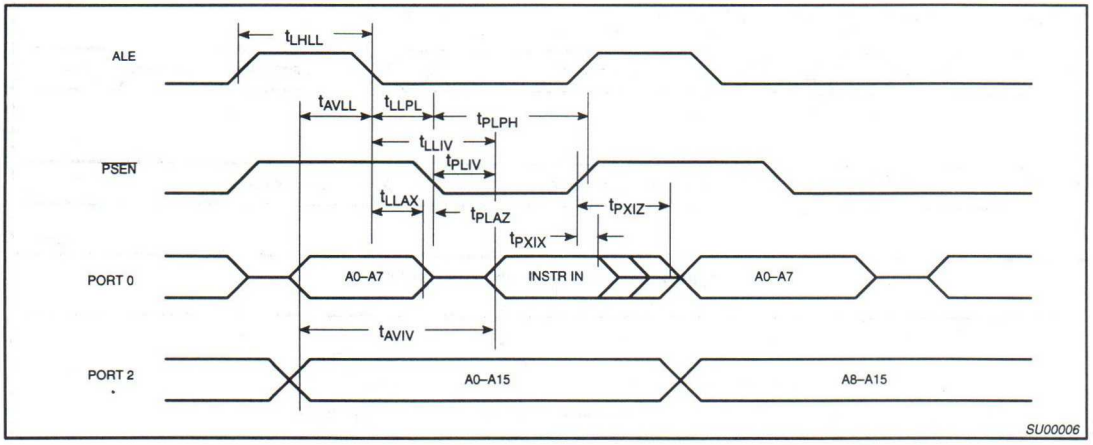



**Figure 2. External Program Memory Read Cycle**

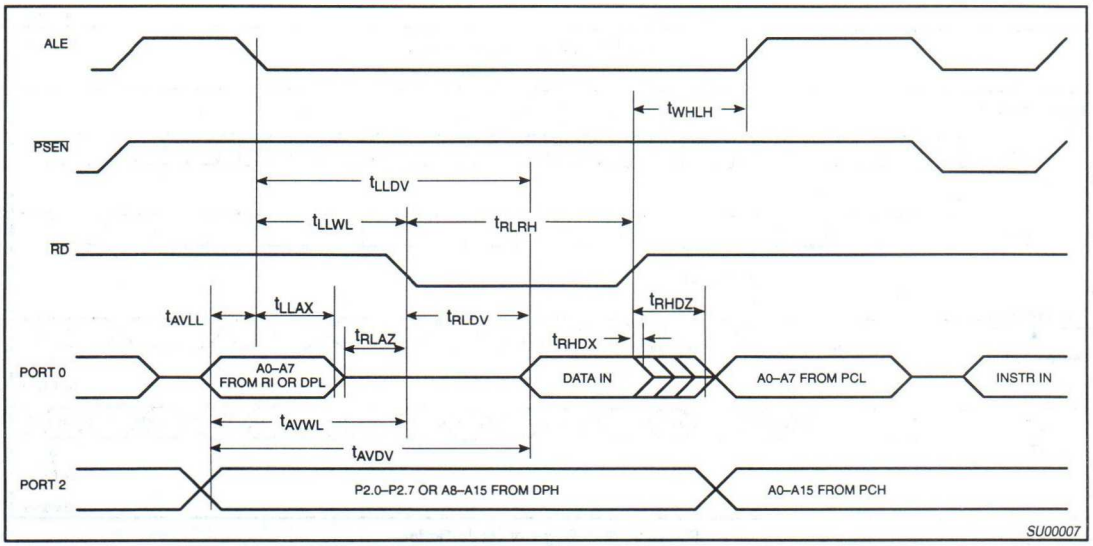



**Figure 3. External Data Memory Read Cycle**

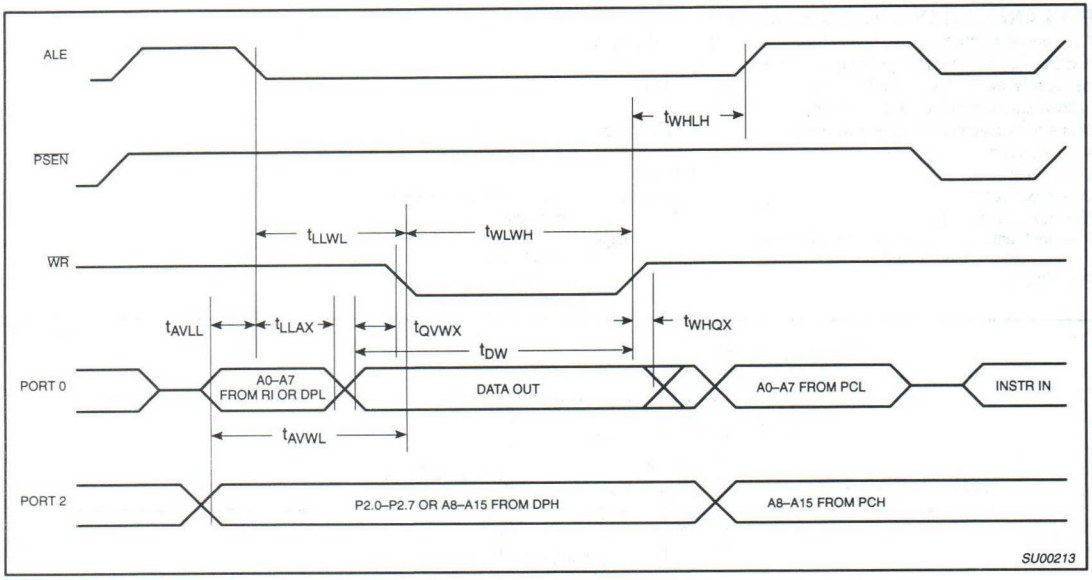



**Figure 4. External Data Memory Write Cycle**

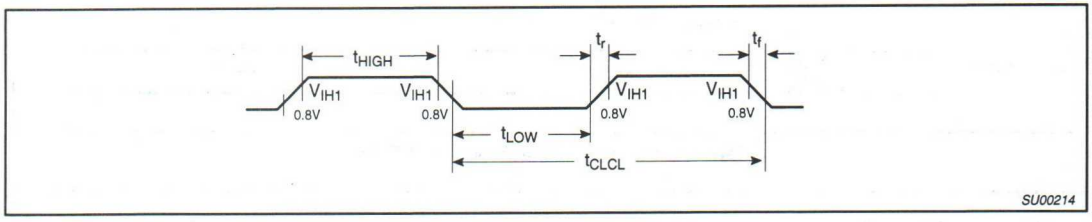



**Figure 5. External Clock Drive XTAL1**

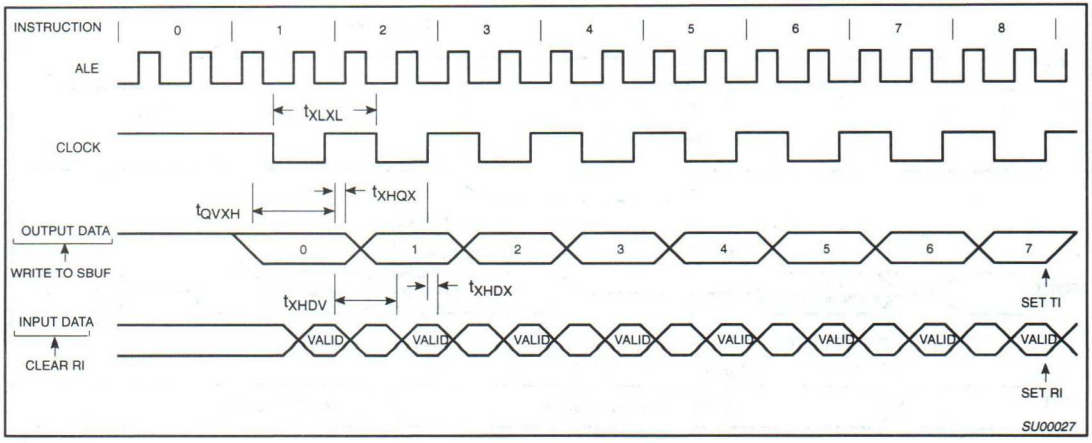
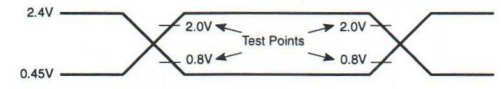


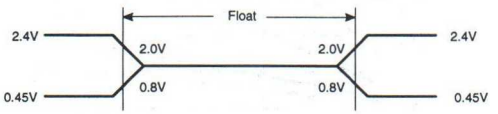

**Figure 6. Shift Register Mode Timing**




NOTE:
AC inputs during testing are driven at 2.4V for a logic '1' and 0.45V for a logic '0'.
Timing measurements are made at 2.0V for a logic '1' and 0.8V for a logic '0'.

SU00215

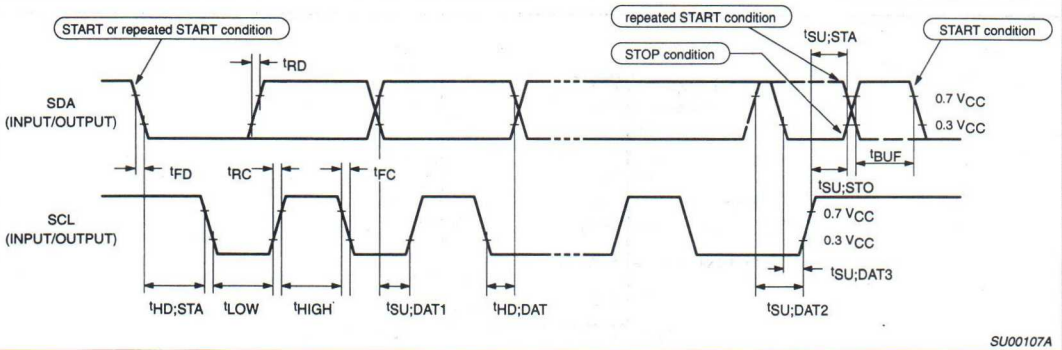
**Figure 7. AC Testing Input/Output**




NOTE:
The float state is defined as the point at which a port 0 pin sinks 3.2mA or sources 400μA at the voltage test levels.

SU00216

**Figure 8. AC Testing Input, Float Waveform**




SU00107A

**Figure 9. Timing SIO1 ($I^2C$) Interface**

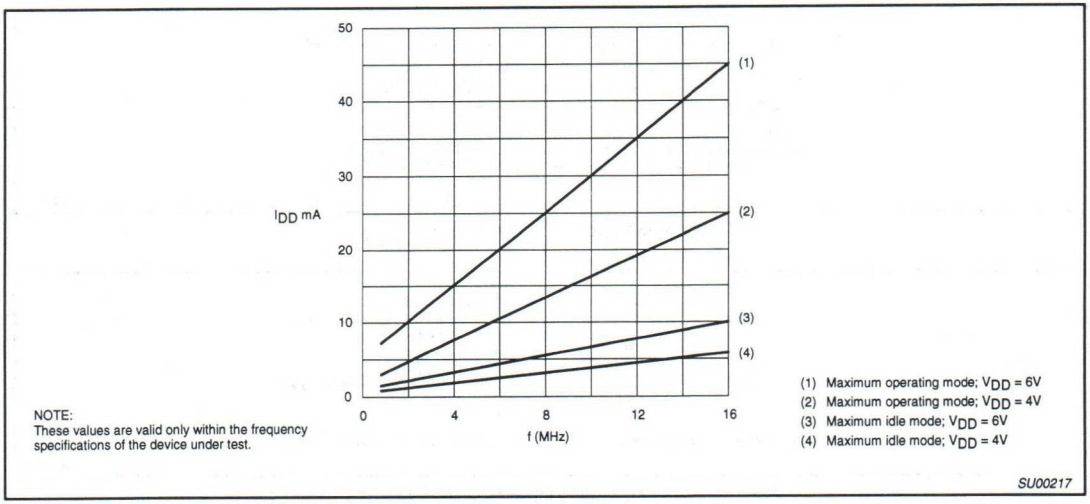



**Figure 10. 16MHz Version Supply Current ($I_{DD}$) as a Function of Frequency at XTAL1 ($f_{OSC}$)**

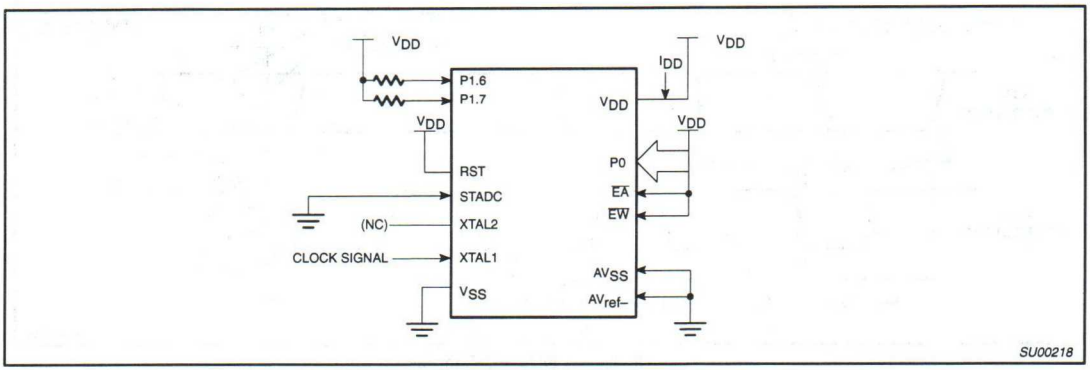



**Figure 11. $I_{DD}$ Test Condition, Active Mode**
**All other pins are disconnected[1]**

1. Active Mode:
   a. The following pins must be forced to $V_{DD}$: $\overline{EA}$, RST, Port 0, and $\overline{EW}$.
   b. The following pins must be forced to $V_{SS}$: STADC, $AV_{ss}$, and $AV_{ref-}$.
   c. Ports 1.6 and 1.7 should be connected to $V_{DD}$ through resistors of sufficiently high value such that the sink current into these pins cannot exceed the $I_{OL1}$ spec of these pins.
   d. The following pins must be disconnected: XTAL2 and all pins not specified above.

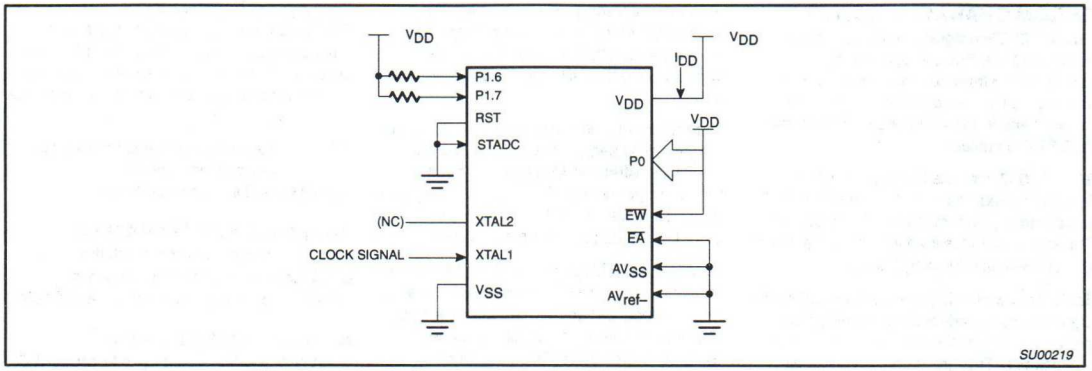



**Figure 12. $I_{DD}$ Test Condition, Idle Mode**
**All other pins are disconnected[2]**

2. Idle Mode:
   a. The following pins must be forced to $V_{DD}$: Port 0 and $\overline{EW}$.
   b. The following pins must be forced to $V_{SS}$: RST, STADC, $AV_{ss}$,, $AV_{ref-}$, and $\overline{EA}$.
   c. Ports 1.6 and 1.7 should be connected to $V_{DD}$ through resistors of sufficiently high value such that the sink current into these pins cannot exceed the $I_{OL1}$ spec of these pins. These pins must not have logic 0 written to them prior to this measurement.
   d. The following pins must be disconnected: XTAL2 and all pins not specified above.

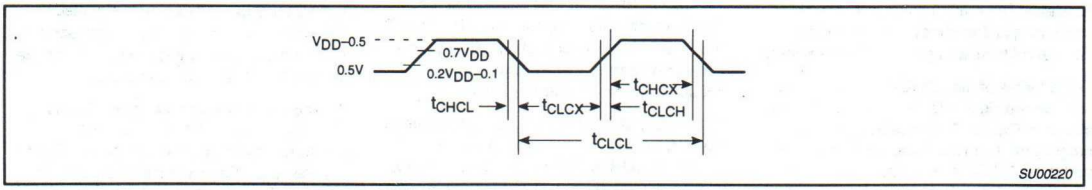



**Figure 13. Clock Signal Waveform for $I_{DD}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

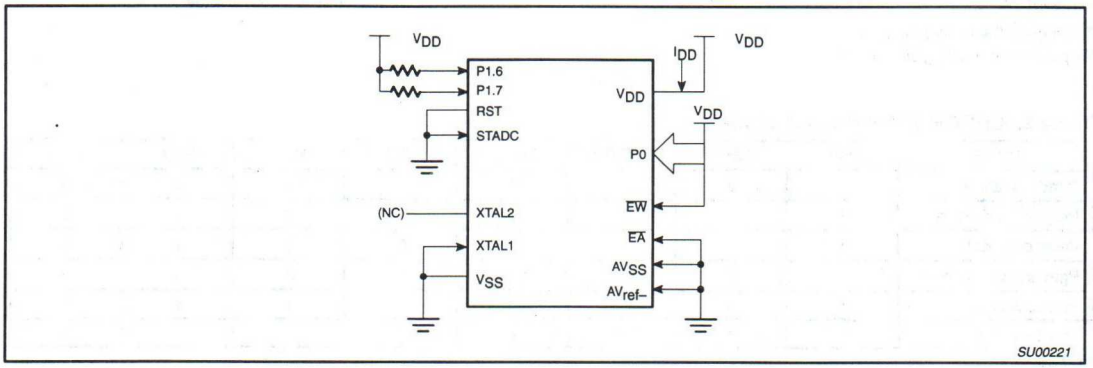



**Figure 14. $I_{DD}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{DD}$ = 2V to 5.5V[3]**

3. Power Down Mode:
   a. The following pins must be forced to $V_{DD}$: Port 0 and $\overline{EW}$.
   b. The following pins must be forced to $V_{SS}$: RST, STADC, XTAL1, $AV_{ss}$,, $AV_{ref-}$, and $\overline{EA}$.
   c. Ports 1.6 and 1.7 should be connected to $V_{DD}$ through resistors of sufficiently high value such that the sink current into these pins cannot exceed the $I_{OL1}$ spec of these pins. These pins must not have logic 0 written to them prior to this measurement.
   d. The following pins must be disconnected: XTAL2 and all pins not specified above.

## EPROM CHARACTERISTICS

The 87C552 is programmed by using a modified Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C552 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C552 manufactured by Philips.

Table 3 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the lock bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 15 and 16. Figure 17 shows the circuit configuration for normal program memory verification.

### Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 15. Note that the 87C552 is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1 and 2, as shown in Figure 15. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$, and pins of ports 2 and 3 specified in Table 3 are held at the "Program Code Data" levels indicated in Table 3. The ALE/$\overline{PROG}$ is pulsed low 25 times as shown in Figure 16.

To program the encryption table, repeat the 25-pulse programming sequence for addresses 0 through 1FH, using the "Pgm Encryption Table" levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the lock bits, repeat the 25-pulse programming sequence using the "Pgm Lock Bit" levels. After one lock bit is programmed, further programming of the code memory and encryption table is disabled. However, the other lock bit can still be programmed.

Note that the $\overline{EA}$/$V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

### Program Verification

If lock bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be red is applied to ports 1 and 2 as shown in Figure 17. The other pins are held at the "Verify Code Data" levels indicated in Table 3. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

**Reading the Signature Bytes**

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:

(030H) = 15H indicates manufactured by Philips Components
(031H) = 94H indicates 87C552

### Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 3, and which satisfies the timing specifications, is suitable.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to the light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345-5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-sec/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient. Erasure leaves the array in an all 1s state.

**Table 3. EPROM Programming Modes**

| MODE | RST | $\overline{PSEN}$ | ALE/$\overline{PROG}$ | $\overline{EA}$/$V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm lock bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm lock bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |

**NOTES:**
1. 0 = Valid low for that pin; 1 = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{DD}$ = 5V ±10% during programming and verification.

* ALE/$\overline{PROG}$ receives 25 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

™Trademark phrase of Intel Corporation.

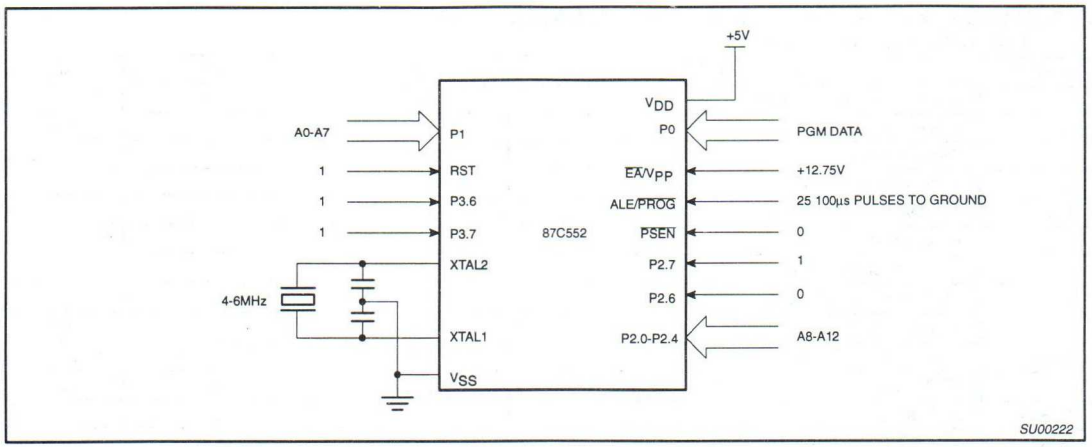



**Figure 15. Programming Configuration**

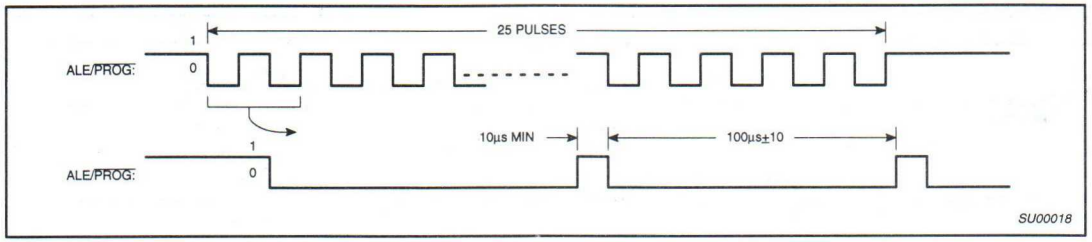



**Figure 16. PROG Waveform**

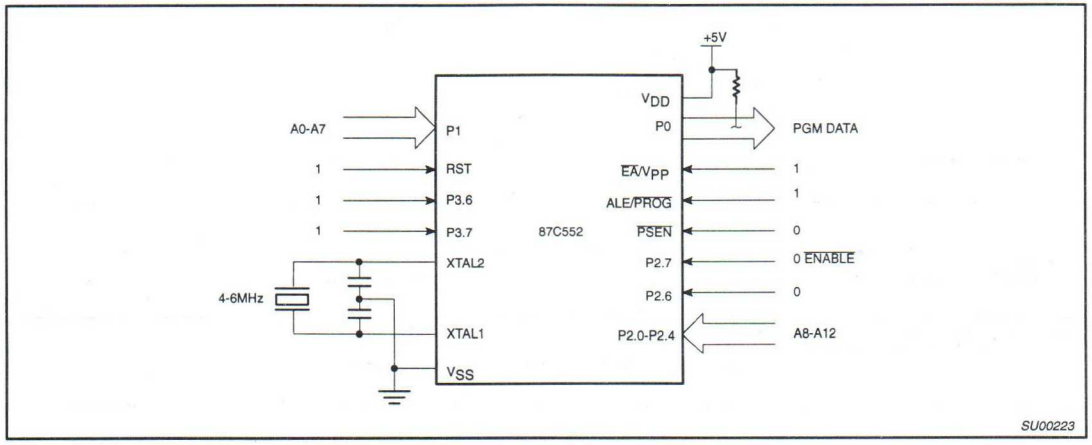
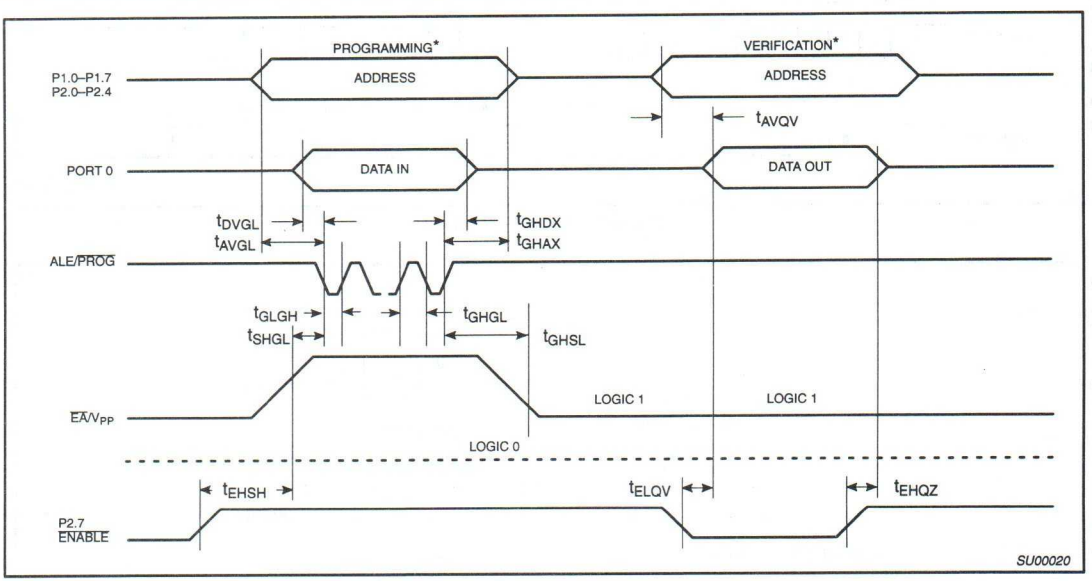



**Figure 17. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{DD}$ = 5V±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




* FOR PROGRAMMING VERIFICATION SEE FIGURE 17.
FOR VERIFICATION CONDITIONS SEE TABLE 3.

**Figure 18. EPROM Programming and Verification**

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# Single-chip 8-bit microcontroller P83CE558/P80CE558/P89CE558

## 1. FEATURES

- 80C51 central processing unit
- 32 K × 8 ROM respectively FEEPROM (Flash-EEPROM), expandable externally to 64 Kbytes
- ROM/FEEPROM Code protection
- 1024 × 8 RAM, expandable externally to 64 Kbytes
- Two standard 16-bit timer/counters
- An additional 16-bit timer/counter coupled to four capture registers and three compare registers
- A 10-bit ADC with eight multiplexed analog inputs and programmable autoscan
- Two 8-bit resolution, pulse width modulation outputs
- Five 8-bit I/O ports plus one 8-bit input port shared with analog inputs
- I²C-bus serial I/O port with byte oriented master and slave functions
- Full-duplex UART compatible with the standard 80C51
- On-chip watchdog timer
- 15 interrupt sources with 2 priority levels (2 to 6 external sources possible)
- Extended temperature range (–40 to +85°C)
- 4.5 to 5.5 V supply voltage range
- Frequency range for 80C51-family standard oscillator: 3.5 MHz to 16 MHz
- PLL oscillator with 32 kHz reference and software-selectable system clock frequency
- Seconds Timer
- Software enable/disable of ALE output pulse
- Electromagnetic compatibility improvements
- Wake-up from Power-down by external or seconds interrupt

## 2. GENERAL DESCRIPTION

The P80CE558/P83CE558/P89CE558 (hereafter generically referred to as P8xCE558) single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The P8xCE558 has the same instruction set as the 80C51. Three versions of the derivative exist:

- P83CE558 — 32 Kbytes mask programmable ROM
- P80CE558 — ROMless version of the P83CE558
- P89CE558 — 32 Kbytes FEEPROM (Flash-EEPROM)

The P8xCE558 contains a non-volatile 32 Kbytes mask programmable ROM (P83CE558) or electrically erasable FEEPROM respectively (P89CE558), a volatile 1024 × 8 read/write data memory, five 8-bit I/O ports, one 8-bit input port, two 16-bit timer/event counters (identical to the timers of the 80C51), an additional 16-bit timer coupled to capture and compare latches, a 15-source, two-priority-level, nested interrupt structure, an 8-input ADC, a dual DAC pulse width modulated interface, two serial interfaces (UART and I²C-bus), a "watchdog" timer, an on-chip oscillator and timing circuits. For systems that require extra capability the P8xCE558 can be expanded using standard TTL compatible memories and logic.

In addition, the P8xCE558 has two software selectable modes of power reduction — Idle Mode and power-down mode. The Idle Mode freezes the CPU while allowing the RAM, timers, serial ports, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic as well as bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte, and 17 three- byte. With a 16 MHz system clock, 58% of the instructions are executed in 0.75 µs and 40% in 1.5 µs. Multiply and divide instructions require 3 µs.

## 3. ORDERING INFORMATION

| EXTENDED TYPE NUMBER | PACKAGE | | | FREQUENCY RANGE (MHz) | TEMPERATURE RANGE (°C) |
|---|---|---|---|---|---|
| | NAME | DESCRIPTION | CODE | | |
| **ROMless** | | | | | |
| P80CE558EBB | QFP80 | Plastic Quad Flat Pack; 80 leads | SOT318-1 | 3.5 to 16 | 0 to +70 |
| P80CE558EFB | QFP80 | Plastic Quad Flat Pack; 80 leads | SOT318-1 | 3.5 to 16 | –40 to +85 |
| **ROM coded** | | | | | |
| P83CE558EBB/YYY[1] | QFP80 | Plastic Quad Flat Pack; 80 leads | SOT318-1 | 3.5 to 16 | 0 to +70 |
| P83CE558EFB/YYY[1] | QFP80 | Plastic Quad Flat Pack; 80 leads | SOT318-1 | 3.5 to 16 | –40 to +85 |
| **FEEPROM** | | | | | |
| P89CE558EBB | QFP80 | Plastic Quad Flat Pack; 80 leads | SOT318-1 | 3.5 to 16 | 0 to +70 |
| P89CE558EFB | QFP80 | Plastic Quad Flat Pack; 80 leads | SOT318-1 | 3.5 to 16 | –40 to +85 |

**NOTE:**
1. YYY denotes the ROM code number

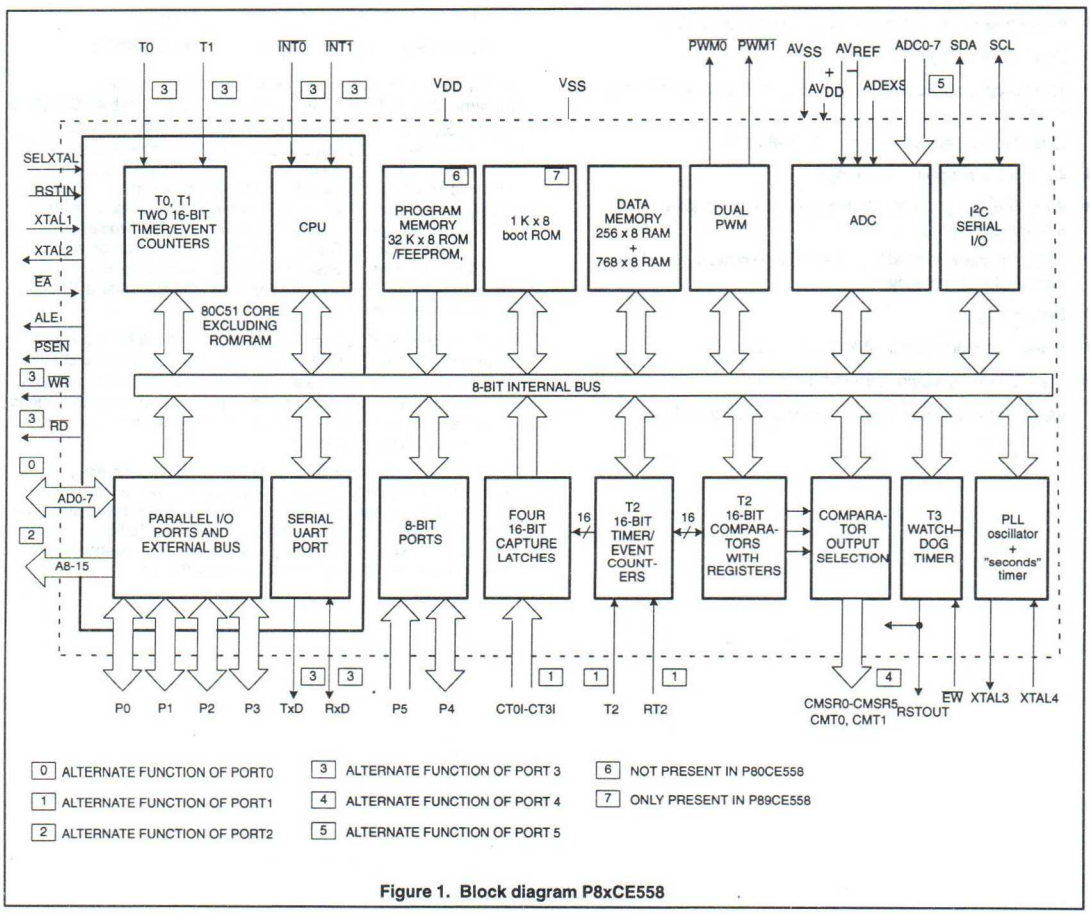
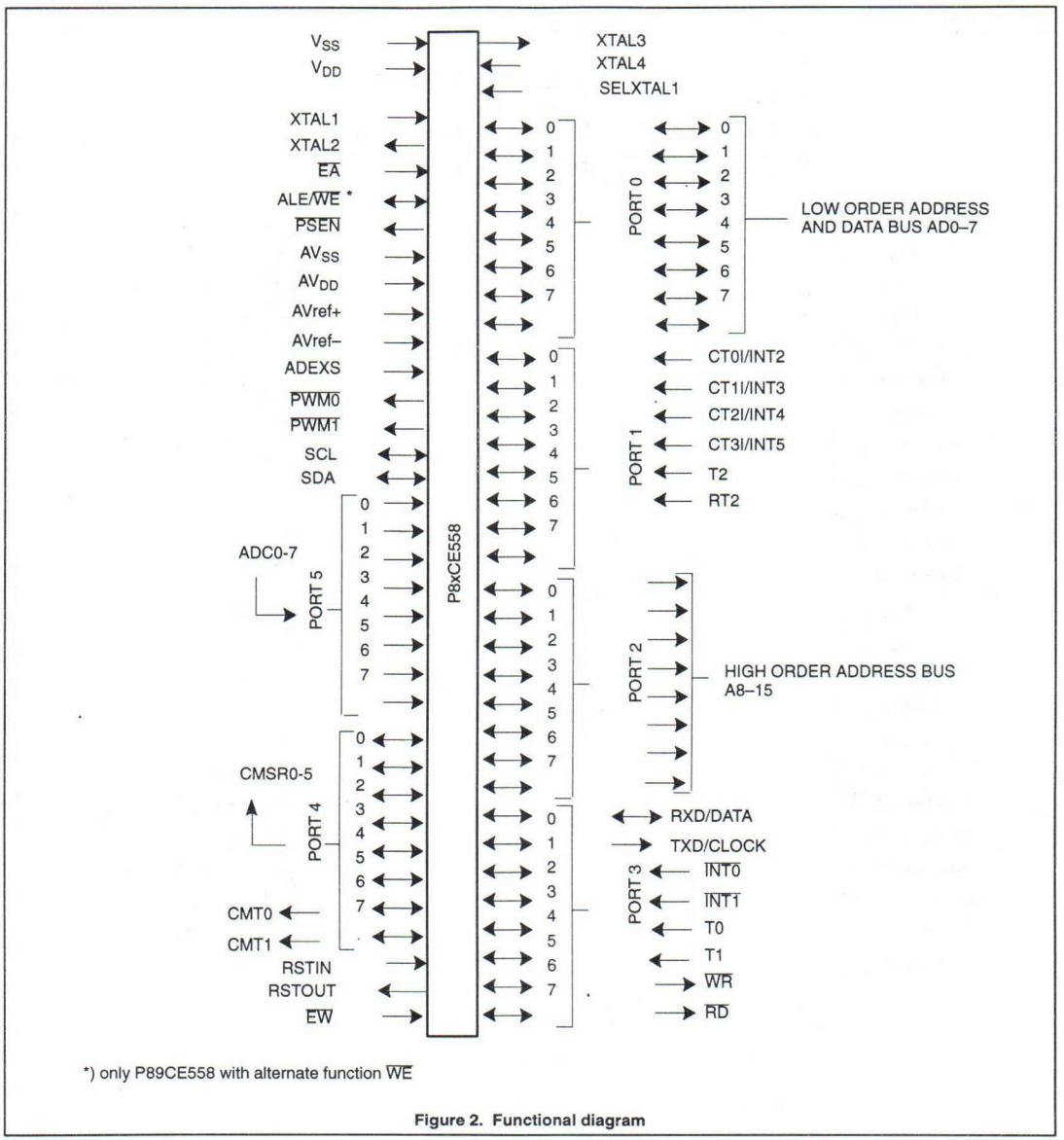


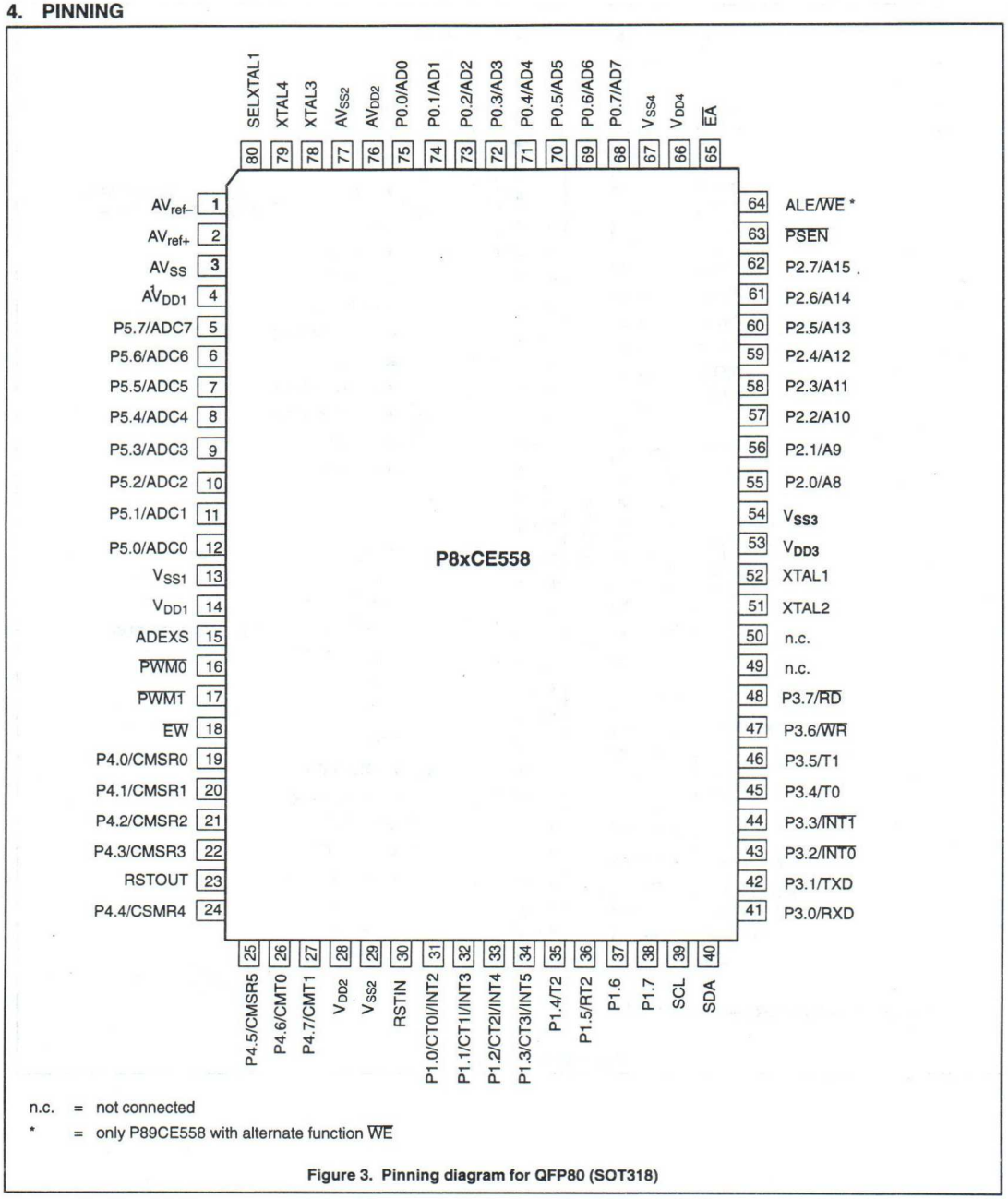

**Figure 1. Block diagram P8xCE558**




*) only P89CE558 with alternate function WE

**Figure 2. Functional diagram**

## 4. PINNING




n.c. = not connected

* = only P89CE558 with alternate function $\overline{WE}$

**Figure 3. Pinning diagram for QFP80 (SOT318)**

## 4.1 PIN DESCRIPTION

| SYMBOL | PIN | DESCRIPTION |
|---|---|---|
| $AV_{ref-}$<br>$AV_{ref+}$ | 1<br>2 | **Low end** of analog to digital conversion reference resistor<br>**High end** of analog to digital conversion reference resistor. |
| $AV_{SS1}$<br>$AV_{DD1}$ | 3<br>4 | **Analog** ground for ADC<br>**Analog** power supply (+5 V) for ADC |
| $AV_{SS2}$<br>$AV_{DD2}$ | 77<br>76 | **Analog** ground; for PLL oscillator<br>**Analog** power supply; (+5 V) for PLL oscillator |
| P5.7 – P5.0 | 5 – 12 | **Port 5**<br>8-bit input port<br>Port pin: Alternative function<br>P5.0–P5.7: Eight input channels to ADC (ADC0–ADC7) |
| $V_{DD1}$, $V_{DD2}$,<br>$V_{DD3}$, $V_{DD4}$ | 14, 28,<br>53, 66 | **Digital power supply:** +5 V power supply pins during normal operation and power reduction modes. All pins must be connected. |
| $V_{SS1}$, $V_{SS2}$<br>$V_{SS3}$, $V_{SS4}$ | 13, 29,<br>54, 67 | **Digital ground:** circuit ground potential. All pins must be connected. |
| ADEXS | 15 | **Start ADC operation:** Input starting analog to digital conversion triggered by a programmable edge (ADC operation can also be started by software). This pin must not float |
| $\overline{PWM0}$ | 16 | **Pulse width modulation** output 0 |
| $\overline{PWM1}$ | 17 | **Pulse width modulation** output 1 |
| $\overline{EW}$ | 18 | **Enable watchdog timer:** Enable for T3 watchdog timer and disable Power-down Mode.This pin must not float. |
| P4.0 – P4.7 | 19 – 22<br>24 – 27 | **Port 4**<br>8-bit quasi-bidirectional I/O port<br>P4.0: CMSR0 }<br>P4.1: CMSR1 }<br>P4.2: CMSR2 } compare and set/reset<br>P4.3: CMSR3 } outputs on a match with timer T2<br>P4.4: CMSR4 }<br>P4.5: CMSR5 }<br>P4.6: CMT0 } compare and toggle outputs<br>P4.7: CMT1 } on a match with timer T2 |
| RSTIN | 30 | **Reset:** Input to reset the P8xCE558. |
| RSTOUT | 23 | **Reset:** Output of the P8xCE558 for resetting peripheral devices during initialization and Watchdog Timer overflow. |
| P1.0 – P1.7 | 31 – 38 | **Port 1**<br>8-bit quasi-bidirectional I/O port<br>P1.0: CT0I/INT2}<br>P1.1: CT1I/INT3} : Capture timer inputs for<br>P1.2: CT2I/INT4} timer T2 or external interrupt inputs<br>P1.3: CT3I/INT5}<br>P1.4: T2 : T2 event input, rising edge triggered<br>P1.5: RT2 : T2 timer reset input, rising edge triggered<br>P1.6<br>P1.7 |
| SCL | 39 | **$I^2C$-bus serial clock** I/O port |
| SDA | 40 | **$I^2C$-bus serial data** I/O port<br>If SCL and SDA are not used, they must be connected to $V_{SS}$. |

**PIN DESCRIPTION** (Continued)

| SYMBOL | PIN | DESCRIPTION |
|---|---|---|
| P3.0 – P3.7 | 41 – 48 | 8-bit quasi-bidirectional I/O port<br>Port pin — Alternative function<br>P3.0 — RXD : Serial input port<br>P3.1 — TXD : Serial output port<br>P3.2 — $\overline{INT0}$ : External interrupt<br>P3.3 — $\overline{INT1}$ : External interrupt<br>P3.4 — T0 : Timer 0 external input<br>P3.5 — T1 : Timer 1 external input<br>P3.6 — $\overline{WR}$ : External data memory write strobe<br>P3.7 — $\overline{RD}$ : External data memory read strobe |
| N.C. | 49 – 50 | Not connected pins. |
| XTAL2 | 51 | **Crystal pin 2**: output of the inverting amplifier that forms the oscillator. Left open-circuit when an external oscillator clock is used. |
| XTAL1 | 52 | **Crystal pin 1**: input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external oscillator clock signal when an external oscillator is used. Must be connected to logic HIGH if the PLL oscillator is selected (SELXTAL1 = LOW). |
| P2.0 – P2.7 | 55 – 62 | **Port2:** 8-bit quasi-bidirectional I/O port with internal pull-ups.During access to external memories (RAM/ROM) that use 16-bit addresses (MOVX@DPTR) Port 2 emits the high order address byte. The alternative function of P2.7 for the P89CE558 is the output enable signal for verify/read modes (active low).<br>Port 2 can sink/source one TTL (=4 LSTTL) input. It can drive CMOS inputs without external pull-ups. |
| $\overline{PSEN}$ | 63 | **Program Store Enable output:** read strobe to the external program memory via Port 0 and 2. Is activated twice each machine cycle during fetches from external program memory. When executing out of external program memory two activations of $\overline{PSEN}$ are skipped during each access to external data memory. $\overline{PSEN}$ is not activated (remains HIGH) during no fetches from external program memory. $\overline{PSEN}$ can sink/source 8 LSTTL inputs. It can drive CMOS inputs without external pull-ups. |
| ALE/$\overline{WE}$ | 64 | **Address Latch Enable output**: latches the low byte of the address during access of external memory in normal operation. It is activated every six oscillator periods except during an external data memory access. ALE/$\overline{WE}$ can sink/-source 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up. The alternative function for the P89CE558 is the programming pulse input $\overline{WE}$.<br>To prohibit the toggling of ALE pin (RFI noise reduction) the bit RFI in the PCON Register (PCON.5) must be set by software. This bit is cleared on RESET and can be set and cleared by software. When set, ALE pin will be pulled down internally, switching an external address latch to a quiet state. The MOVX instruction will still toggle ALE if external memory is accessed.<br>ALE will retain its normal high value during Idle Mode and a low value during Power-down Mode while in the "RFI" mode. Additionally during internal access ($\overline{EA}$ = 1) ALE will toggle normally when the address exceeds the internal program memory size. During external access ($\overline{EA}$ = 0) ALE will always toggle normally, whether the flag "RFI" is set or not. |
| $\overline{EA}$ | 65 | **External Access Input:** If, during RESET, $\overline{EA}$ is held at a TTL level HIGH the CPU executes out of the internal program memory, provided the program counter is less than 32768. If, during RESET, $\overline{EA}$ is held at a TTL level LOW the CPU executes out of external program memory via Port 0 and Port 2. $\overline{EA}$ is not allowed to float. $\overline{EA}$ is latched during RESET and don't care after RESET. |
| P0.7–P0.0 | 68 –75 | **Port 0:** 8-bit open drain bidirectional I/O port. It is also the multiplexed low-order address and data bus during accesses to external memory (during theses accesses internal pull-ups are activated). Port 0 can sink/source 8 LSTTL inputs. |
| XTAL3 | 78 | **Crystal pin**, output of the inverting amplifier that forms the 32 kHz oscillator |
| XTAL4 | 79 | **Crystal pin**, input to the inverting amplifier that forms the 32 kHz oscillator. XTAL3 and XTAL4 are pulled LOW if the PLL oscillator is not selected (SELXTAL1 = HIGH) or if Reset is active. |
| SELXTAL1 | 80 | Must be connected to logic HIGH level to select the HF oscillator, using the XTAL1/XTAL2 crystal. If pulled low the PLL is selected for clocking of the controller, using the XTAL3/ XTAL4 crystal. |

**NOTE:**

1. To avoid a 'latch-up' effect at Power-on, the voltage at any pin at any time must not be higher or lower than $V_{DD}$+ 0.5 V or $V_{SS}$– 0.5 V respectively.

## 5. ELECTROMAGNETIC COMPATIBILITY (EMC) IMPROVEMENTS

Primary attention was paid on the reduction of electromagnetic emission of the microcontroller P8xCE558.

The following features effect in reducing the electromagnetic emission and additionally improve the electromagnetic susceptibility:

- Four supply voltage pins ($V_{DD}$) and four ground pins ($V_{SS}$) with pairs of $V_{DD}$ and $V_{SS}$ at two adjacent pins at each side of the package.
- Separated $V_{DD}$ pins for the internal logic and the port buffers
- Internal decoupling capacitance improves the EMC radiation behavior and the EMC immunity
- External capacitors are to be located as close as possible between pins $V_{DD1}$ and $V_{SS1}$, $V_{DD2}$ and $V_{SS2}$, $V_{DD3}$ and $V_{SS3}$ as well as $V_{DD4}$ and $V_{SS4}$ ; ceramic chip capacitors are recommended (100nF).

Useful in applications that require no external memory or temporarily no external memory:

- The ALE output signal (pulses at a frequency of $f_{CLK}/6$) can be disabled under software control (bit 5 in the SFR PCON: "RFI"); if disabled, no ALE pulse will occur. ALE pin will be pulled down internally, switching an external address latch to a quiet state. The MOVX instruction will still toggle ALE (external data memory is accessed). ALE will retain its normal HIGH value during Idle Mode and a LOW value during Power-down mode while in the "RFI" reduction mode. Additionally during internal access ($\overline{EA}$ = 1) ALE will toggle normally when the address exceeds the internal program memory size. During external access ($\overline{EA}$ = 0) ALE will always toggle normally, whether the flag "RFI" is set or not.

## 6. FUNCTIONAL DESCRIPTION

### 6.1 General

The P8xCE558 is a stand-alone high-performance microcontroller designed for use in real time applications such as instrumentation, industrial control, medium to high-end consumer applications and specific automotive control applications.

In addition to the 80C51 standard functions, the device provides a number of dedicated hardware functions for these applications.

The P8xCE558 is a control-oriented CPU with on-chip program and data memory. It can be extended with external program memory up to 64 Kbytes. It can also access up to 64 Kbytes of external data memory. For systems requiring extra capability, the P8xCE558 can be expanded using standard memories and peripherals.

The P8xCE558 has two software selectable modes of reduced activity for further power reduction – Idle and Power-down. The Idle Mode freezes the CPU while allowing the RAM, timers, serial ports and interrupt system to continue functioning. The Power-down Mode saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative. The Power-down Mode can be terminated by an external Reset, by the seconds interrupt and by any one of the two external interrupts. (See description Wake-up from Power-down Mode.)

### 6.2 Memory organization

The central processing unit (CPU) manipulates operands in three memory spaces; these are the 64 Kbytes external data memory, 1024 bytes internal data memory (consisting of 256 bytes standard RAM and 768 bytes AUX-RAM) and the 32 Kbytes internal and/or 64 Kbytes external program memory (see Figure 4).

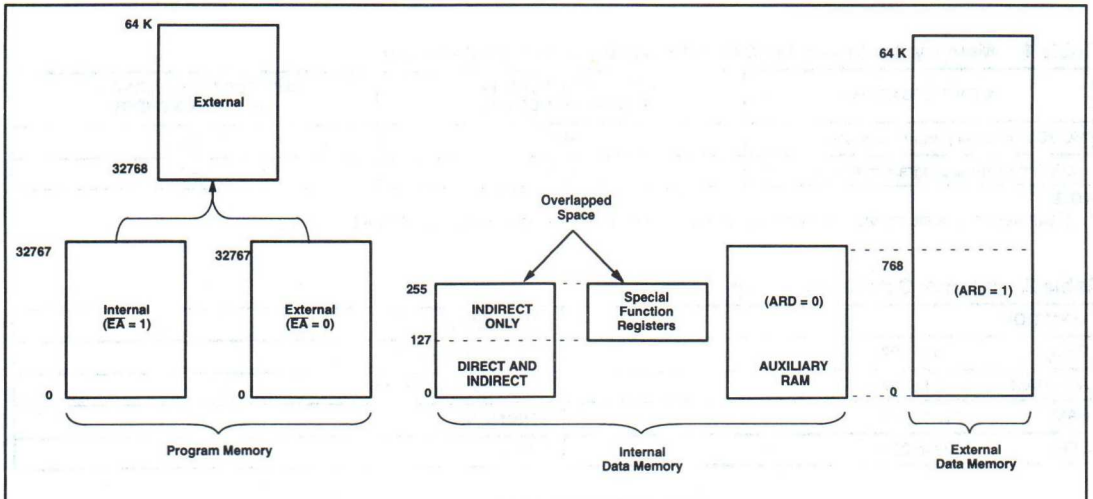



**Figure 4. Memory map & address space**

#### 6.2.1 Program Memory

The program memory of the P8xCE558 consists of 32 Kbytes ROM respectively FEEPROM ("Flash Memory") on-chip, externally expandable up to 64 Kbytes. If, during RESET, the $\overline{EA}$ pin was held HIGH, the P8xCE558 executes out of the internal program memory unless the address exceeds 7FFFH. Locations 8000H through 0FFFFH are then fetched from the external program memory. If the $\overline{EA}$ pin was held LOW during RESET the P8xCE558 fetches all instructions from the external program memory. The $\overline{EA}$ input is latched during RESET and is don't care after RESET.

The internal program memory content is protected, by setting a mask programmable security bit (ROM) or by the software programmable security byte (FEEPROM) respectively, i.e. it cannot be read out at any time by any test mode or by any instruction in the external program memory space. The MOVC instructions are the only ones which have access to program code in the internal or external program memory. The $\overline{EA}$ input is latched during RESET and is 'don't care' after RESET. This implementation prevents from reading internal program code by switching from external program memory to internal program memory during MOVC instruction or an instruction that handles immediate data. Table 1 lists the access to the internal and external program memory with MOVC instructions when the security feature has been activated.

#### 6.2.2 Internal Data Memory

The internal data memory is divided into three physically separated parts:

256 bytes of RAM, 768 bytes of AUX-RAM, and a 128 bytes special function area. These can be addressed each in a different way (see also Table 2).

- RAM 0 to 127 can be addressed directly and indirectly as in the 80C51. Address pointers are R0 and R1 of the selected registerbank.
- RAM 128 to 255 can only be addressed indirectly. Address pointers are R0 and R1 of the selected registerbank.
- AUX-RAM 0 to 767 is also indirectly addressable as external DATA MEMORY locations 0 to 767 via MOVX-Datapointer instruction, unless it is disabled by setting ARD = 1.
  AUX-RAM 0 to 767 is indirectly addressable via pageregister (XRAMP) and MOVX-Ri instructions, unless it is disabled by setting ARD = 1 (see Figure 5).
  When executing from internal program memory, an access to AUX-RAM 0 to 767 will not affect the ports P0, P2, P3.6 and P3.7.

  An access to external DATA MEMORY locations higher than 767 will be performed with the MOVX @ DPTR instructions in the same way as in the 80C51 structure, so with P0 and P2 as data/address bus and P3.6 and P3.7 as write and read timing signals. Note that the external DATA MEMORY cannot be accessed with R0 and R1 as address pointer if the AUX-RAM is enabled (ARD = 0, default).
- The Special Function Registers (SFR) can only be addressed directly in the address range from 128 to 255 (see Table 5).
- Four register banks, each 8 registers wide, occupy locations 0 through 31 in the lower RAM area. Only one of these banks may be enabled at a time. The next 16 bytes, locations 32 through 47, contain 128 directly addressable bit locations.The stack can be located anywhere in the internal 256 bytes RAM.The stack depth is only limited by the available internal RAM space of 256 bytes (see Figure 7).

  All registers except the program counter and the four register banks reside in the Special Function Register address space.

**Table 1. Memory Access by the MOVC Instruction for Protected ROMs**

| MOVC LOCATION | ACCESS TO INTERNAL PROGRAM MEMORY | ACCESS TO EXTERNAL PROGRAM MEMORY |
|---|---|---|
| MOVC in internal program memory | YES | YES |
| MOVC in external program memory | NO | YES |

**NOTE:**
1. If the security feature has not been activated, there are no restrictions for MOVC instructions.

**Table 2. Internal Data Memory Map**

| LOCATION | | ADDRESSED |
|---|---|---|
| RAM | 0 to 127 | Direct and indirect |
| AUX-RAM | 0 to 767 | Indirect only with MOVX |
| RAM | 128 to 255 | Indirect only |
| SFR | 128 to 255 | Direct only |

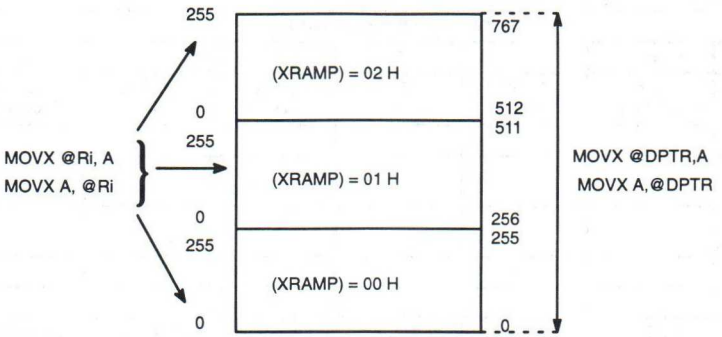



**Figure 5. Indirect addressing of AUX-RAM (768 Bytes), ARD bit in PCON = 0**

#### 6.2.2.1 AUX-RAM Page Register XRAMP

The AUX-RAM Page Register is used to select one of three 256 bytes pages of the internal 768 bytes AUX-RAM for MOVX-accesses via R0 or R1. Its reset value is (XXXXXX00).

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **XRAMP (FAH)** | x | x | x | x | x | x | XRAMP1 | XRAMP0 |

x: undefined during read, a write operation must write "0" to these locations

**Figure 6. AUX-RAM page register.**

### Table 3. Description of XRAMP Bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| XRAMP.2–7 | XRAMPx | reserved for future use |
| XRAMP.1 | XRAMP1 | AUX-RAM page select bit 1 |
| XRAMP.0 | XRAMP0 | AUX-RAM page select bit 0 |

### Table 4. Memory Locations for All Possible MOVX Accesses

| ARD[1] | XRAMP1 | XRAMP0 | MOVX @Ri,A and MOVX A,@Ri instructions access: |
|---|---|---|---|
| 0 | 0 | 0 | AUX-RAM locations 0 .. 255 (reset condition) |
| 0 | 0 | 1 | AUX-RAM locations 256 .. 511 |
| 0 | 1 | 0 | AUX-RAM locations 512 .. 767 |
| 0 | 1 | 1 | no valid memory access; reserved for future use |
| 1 | X | X | External RAM locations 0 .. 255 |
| | | | **MOVX @DPTR,A and MOVX A,@DPTR instructions access:** |
| 0 | X | X | AUX-RAM locations 0 .. 767 (reset condition)<br>External RAM locations 768 .. 65535 |
| 1 | X | X | External RAM locations 0 .. 65535 |

**NOTE:**
1. ARD (AUX-RAM Disable) is a bit in the Special Function Register PCON

**Table 5. Special Function Register Memory Map and Reset Values**

| | HIGH NIBBLE OF SFR ADDRESS | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| LOW | 8 | 9 | A | B | C | D | E | F |
| 0 | P0 % 11111111 | P1 % 11111111 | P2 % 11111111 | P3 % 11111111 | P4 % 11111111 | PSW % 00000000 | ACC % 00000000 | B % 00000000 |
| 1 | SP 00000111 | | | | | | | |
| 2 | DPL 00000000 | | | | | | | |
| 3 | DPH 00000000 | | | | | | | |
| 4 | | | | | | | | |
| 5 | | | | | | | | |
| 6 | ADRSL0 # XXXXXXXX | ADRSL1 # XXXXXXXX | ADRSL2 # XXXXXXXX | ADRSL3 # XXXXXXXX | ADRSL4 # XXXXXXXX | ADRSL5 # XXXXXXXX | ADRSL6 # XXXXXXXX | ADRSL7 # XXXXXXXX |
| 7 | PCON 00000000 | | | | P5 # XXXXXXXX | ADCON 00000000 | ADPSS 00000000 | ADRSH # 000000XX |
| 8 | TCON % 00000000 | S0CON % 00000000 | IEN0 % 00000000 | IP0 % X0000000 | TM2IR % 00000000 | S1CON % 00000000 | IEN1 % 00000000 | IP1 % 00000000 |
| 9 | TMOD 00000000 | S0BUF XXXXXXXX | CML0 00000000 | | CMH0 00000000 | S1STA # 11111000 | | PLLCON 00001101 |
| A | TL0 00000000 | | CML1 00000000 | | CMH1 00000000 | S1DAT 00000000 | TM2CON 00000000 | XRAMP XXXXXX00 |
| B | TL1 00000000 | | CML2 00000000 | | CMH2 00000000 | S1ADR 00000000 | CTCON 00000000 | FMCON * 000X0000 |
| C | TH0 00000000 | | CTL0 # XXXXXXXX | | CTH0 # XXXXXXXX | | TML2 # 00000000 | PWM0 00000000 |
| D | TH1 00000000 | | CTL1 # XXXXXXXX | | CTH1 # XXXXXXXX | | TMH2 # 00000000 | PWM1 00000000 |
| E | | | CTL2 # XXXXXXXX | | CTH2 # XXXXXXXX | | STE 11000000 | PWMP 00000000 |
| F | | | CTL3 # XXXXXXXX | | CTH3 # XXXXXXXX | | RTE 00000000 | T3 00000000 |

**NOTES:**
% = Bit addressable register
# = Read only register
X = Undefined
* = FMCON only in P89CE558

## 6.3 Addressing

The P8xCE558 has five methods for addressing:

- Register
- Direct
- Register-Indirect
- Immediate
- Base-Register plus Index-Register-Indirect

The first three methods can be used for addressing destination operands. Most instructions have a "destination/source" field that specifies the data type, addressing methods and operands involved. For operations other than MOVs, the destination operand is also a source operand.

Access to memory addresses is as follows:

- Register in one of the four register banks through Register, Direct or Register-Indirect addressing
- 1024 bytes of internal RAM through Direct or Register-Indirect addressing.
  - Bytes 0–127 of internal RAM may be addressed directly/indirectly. Bytes 128–255 of internal RAM share their address location with the SFRs and so may only be addressed indirectly as data RAM.
  - Bytes 0–767 of AUX-RAM can only be addressed indirectly via MOVX.
- Special Function Register through direct addressing at address locations 128–255 (see Figure 8).
- External data memory through Register-Indirect addressing
- Program memory look-up tables through Base-Register plus Index-Register-Indirect addressing

| BYTE ADDRESS (HEX) | BIT ADDRESS (HEX) | | | | | | | | BYTE ADDRESS (DECIMAL) |
|---|---|---|---|---|---|---|---|---|---|
| FFH | (MSB) | | | | | | | (LSB) | 255 |
| 2FH | 7F | 7E | 7D | 7C | 7B | 7A | 79 | 78 | 47 |
| 2EH | 77 | 76 | 75 | 74 | 73 | 72 | 71 | 70 | 46 |
| 2DH | 6F | 6E | 6D | 6C | 6B | 6A | 69 | 68 | 45 |
| 2CH | 67 | 66 | 65 | 64 | 63 | 62 | 61 | 60 | 44 |
| 2BH | 5F | 5E | 5D | 5C | 5B | 5A | 59 | 58 | 43 |
| 2AH | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 42 |
| 29H | 4F | 4E | 4D | 4C | 4B | 4A | 49 | 48 | 41 |
| 28H | 47 | 46 | 45 | 44 | 43 | 42 | 41 | 40 | 40 |
| 27H | 3F | 3E | 3D | 3C | 3B | 3A | 39 | 38 | 39 |
| 26H | 37 | 36 | 35 | 34 | 33 | 32 | 31 | 30 | 38 |
| 25H | 2F | 2E | 2D | 2C | 2B | 2A | 29 | 28 | 37 |
| 24H | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 36 |
| 23H | 1F | 1E | 1D | 1C | 1B | 1A | 19 | 18 | 35 |
| 22H | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 34 |
| 21H | 0F | 0E | 0D | 0C | 0B | 0A | 09 | 08 | 33 |
| 20H | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 | 32 |
| 1FH | Bank 3 | | | | | | | | 31 |
| 18H | | | | | | | | | 24 |
| 17H | Bank 2 | | | | | | | | 23 |
| 10H | | | | | | | | | 16 |
| 0FH | Bank 1 | | | | | | | | 15 |
| 08H | | | | | | | | | 8 |
| 07H | Bank 0 | | | | | | | | 7 |
| 00H | | | | | | | | | 0 |

**Figure 7. RAM bit addresses**

| DIRECT BYTE ADDRESS (HEX) | BIT ADDRESS (HEX) (MSB) | | | | | | | (LSB) | REGISTER MNEMONIC |
|---|---|---|---|---|---|---|---|---|---|
| FFH | | | | | | | | | |
| | PT2 | PCM2 | PCM1 | PCM0 | PCT3 | PCT2 | PCT1 | PCT0 | |
| F8H | FF | FE | FD | FC | FB | FA | F9 | F8 | IP1 |
| | | | | | | | | | |
| F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | B |
| | ET2 | ECM2 | ECM1 | ECM0 | ECT3 | ECT2 | ECT1 | ECT0 | |
| E8H | EF | EE | ED | EC | EB | EA | E9 | E8 | IEN1 |
| | | | | | | | | | |
| E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | ACC |
| | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 | |
| D8H | DF | DE | DD | DC | DB | DA | D9 | D8 | S1CON |
| | CY | AC | F0 | RS1 | RS0 | OV | F1 | P | |
| D0H | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | PSW |
| | T2OV | CMI2 | CMI1 | CMI0 | CTI3 | CTI2 | CTI1 | CTI0 | |
| C8H | CF | CE | CD | CC | CB | CA | C9 | C8 | TM2IR |
| | | | | | | | | | |
| C0H | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 | P4 |
| | – | PAD | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 | |
| B8H | BF | BE | BD | BC | BB | BA | B9 | B8 | IP0 |
| | | | | | | | | | |
| B0H | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | P3 |
| | EA | EAD | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 | |
| A8H | AF | AE | AD | AC | AB | AA | A9 | A8 | IEN0 |
| | | | | | | | | | |
| A0H | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | P2 |
| | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | |
| 98H | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | S0CON |
| | | | | | | | | | |
| 90H | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | P1 |
| | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | |
| 88H | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | TCON |
| | | | | | | | | | |
| 80H | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | P0 |

**Figure 8. Special Function Register bit addresses**

## 6.4 I/O Facilities

The P8xCE558 has six 8-bit ports. Ports 0 to 3 are the same as in the 80C51, with the exception of the additional functions of Port 1. The parallel I/O function of Port 4 is equal to that of Ports 1, 2 and 3. Port 5 has a parallel input port function, but has no function as an output port.

The SDA and SCL lines serve the serial port SIO1 ($I^2C$). Because the $I^2C$-bus may be active while the device is disconnected from $V_{DD}$, these pins, are provided with open drain drivers.

Ports 0, 1, 2, 3, 4 and 5 perform the following alternative functions:

Port 0 : provides the multiplexed low-order address and data bus used for expanding the P8xCE558 with standard memories and peripherals.

Port 1 : Port 1 is used for a number of special functions:

4 capture inputs (or external interrupt request inputs if capture information is not utilized)
- external counter input
- external counter reset input

Port 2 : provides the high-order address bus when the P8xCE558 is expanded with external Program Memory and/or external Data Memory.

Port 3 : pins can be configured individually to provide:
- external interrupt request inputs
- counter inputs
- receiver input and transmitter output of seri port SIO 0 (UART)
- control signals to read and write external Data Memory

Port 4 : can be configured to provide signals indicating a match between timer counter T2 and its compare registers.

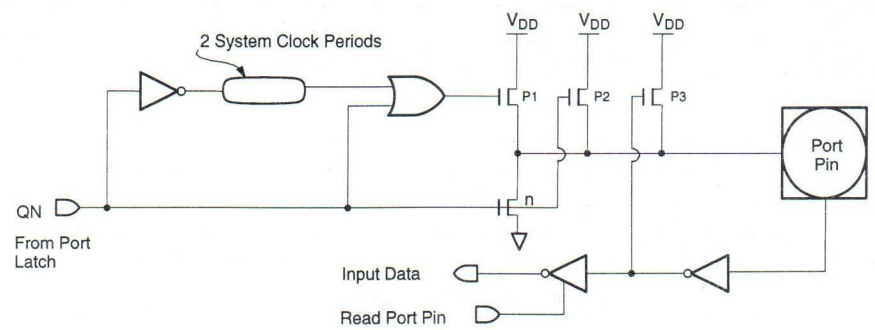
Port 5 : may be used in conjunction with the ADC interface. Unused analog inputs can be used as digital inputs. As Port 5 lines may be used as inputs to the ADC, these digital inputs have an inherent hysteresis to prevent the input logic from drawing too much current from the power lines when driven by analog signals. Channel to channel crosstalk should be taken into consideration when both digital and analog signals are simultaneously input to Port 5 (see DC characteristics).

All ports are bidirectional with the exception of Port 5 which is an input port.

Pins of which the alternative function is not used may be used as normal bidirectional I/Os.

The generation or use of a Port 1, Port 3 or Port 4 pin as an alternative function is carried out automatically by the P8xCE558 provided the associated Special Function Register bit is set HIGH.

The pull-up arrangements of Ports 1 – 4 are shown in Figure 9.




P1 is turned on for 2 system clock periods after QN makes a 1-to-0 transition.
During this time, P1 also turns on P3 through the inverter to form an additional pull up.

**Figure 9. I/O buffers in the P8xCE558 (Ports 1, 2, 3 and 4)**

## 6.5 Pulse Width Modulated Outputs

The P8xCE558 contains two pulse width modulated output channels (see Figure 13). These channels generate pulses of programmable length and interval. The repetition frequency is defined by an 8-bit prescaler PWMP, which supplies the clock for the counter. The prescaler and counter are common to both PWM channels. The 8-bit counter counts module 255, i.e., from 0 to 254 inclusive. The value of the 8-bit counter is compared to the contents of two registers: PWM0 and PWM1. Provided the contents of either of these registers is greater than the counter value, the corresponding $\overline{PWM0}$ or $\overline{PWM1}$ output is set LOW. If the contents of these registers are equal to, or less than the counter value, the output will be HIGH. The pulse-width-ratio is therefore defined by the contents of the registers PWM0 and PWM1. The pulse-width-ratio is in the range of 0/255 to 255/255 and may be programmed in increments of 1/255.

Buffered PWM outputs may be used to drive DC motors. The rotation speed of the motor would be proportional to the contents of PWMn. The PWM outputs may also be configured as a dual DAC. In this application, the PWM outputs must be integrated using conventional operational amplifier circuitry. If the resulting output voltages have to be accurate, external buffers with their own analog supply should be used to buffer the PWM outputs before they are integrated. The repetition frequency fpwm, at the PWMn outputs is give by:

$$fpwm = \frac{f_{CLK}}{2 \times (1 + PWMP) \times 255}$$

This gives a repetition frequency range of 123 Hz to 31.4 kHz ($f_{CLK}$ = 16 MHz). By loading the PWM registers with either 00H or FFH, the PWM channels will output a constant HIGH or LOW level, respectively. Since the 8-bit counter counts modulo 255, it can never actually reach the value of the PWM registers when they are loaded with FFH.

When a compare register (PWM0 or PWM1) is loaded with a new value, the associated output is updated immediately. It does not have to wait until the end of the current counter period. Both $\overline{PWMn}$ output pins are driven by push-pull drivers. These pins are not used for any other purpose.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **PWMP (FEH)** | PWMP.7 | PWMP.6 | PWMP.5 | PWMP.4 | PWMP.3 | PWMP.2 | PWMP.1 | PWMP.0 |

**Figure 10. Prescaler frequency control register PWMP.**

**Table 6. Description of PWMP Bits**

| BIT | FUNCTION |
|---|---|
| PWMP.0 to 7 | Prescaler division factor = (PWMP) + 1 |

**NOTE:**
1. Reading PWMP gives the current reload value. The actual count of the prescaler cannot be read.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM0 (FCH)** | PWM0.7 | PWM0.6 | PWM0.5 | PWM0.4 | PWM0.3 | PWM0.2 | PWM0.1 | PWM0.0 |

**Figure 11. Pulse width register PWM0.**

**Table 7. Description of PWM0 bits**

| BIT | FUNCTION |
|---|---|
| PWM0.0 to 7 | LOW/HIGH ration of $\overline{PWM0}$ signal = $\frac{(PWM0)}{255 - (PWM0)}$ |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM1 (FDH)** | PWM1.7 | PWM1.6 | PWM1.5 | PWM1.4 | PWM1.3 | PWM1.2 | PWM1.1 | PWM1.0 |

**Figure 12. Pulse width register PWM1.**

**Table 8. Description of PWM1 bits**

| BIT | FUNCTION |
|---|---|
| PWM1.0 to 7 | LOW/HIGH ration of $\overline{PWM1}$ signal = $\frac{(PWM1)}{255 - (PWM1)}$ |

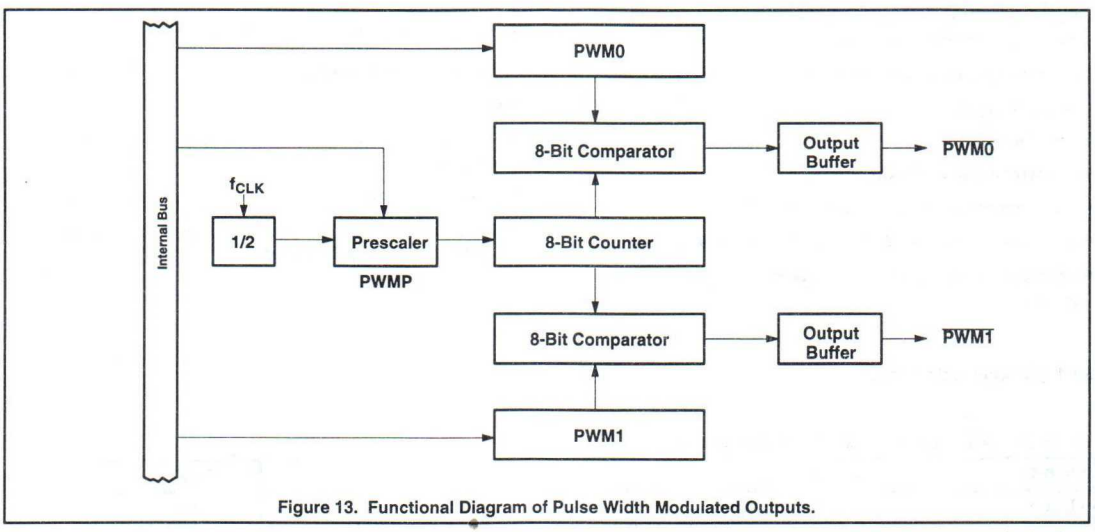



**Figure 13. Functional Diagram of Pulse Width Modulated Outputs.**

## 6.6 Analog/Digital Converter (ADC)

The P8xCE558 A/D Converter is a 10-bit, successive approximation ADC with 8 multiplexed analog input channels. It additionally contains a high input impedance comparator, a DAC built with 1024 series resistors and analog switches, registers and control logic.

Input voltage range is from $AV_{ref-}$ (typical 0V) to $AV_{ref+}$ (typical +5V). A set of 8 buffer registers (10-bit) store the conversion results of the proper analog input channel each.

11 Special Function Registers (SFR) perform the user software interface to the ADC: a control SFR (ADCON), an analog port scan-select SFR (ADPSS), 8 input channel related conversion result SFR with the 8 lower result bits (ADRSL0...ADRSL7), one common result SFR for the upper 2 result bits (ADRSH). An extra SFR (P5) allows for reading digital input port data as an alternative function of the 8 analog input pins.

In order to have a minimum of ADC service overhead in the microcontroller program, the ADC is able to operate autonomously within its user configurable autoscan function.

The functional diagram of the ADC is shown in Figure 15.

**Feature Overview:**

- 10-bit resolution.
- 8 multiplexed analog inputs.
- Programmable autoscan of the analog inputs.
- Bit oriented 8-bit scan-select register to select analog inputs.
- Continuous scan or one time scan configurable from 1 to 8 analog inputs.
- Start of a conversion by software or with an external signal.
- Eight 10-bit buffer registers, one register for each analog input channel.
- Interrupt request after one channel scan loop.
- Programmable prescaler (dividing by 2, 4, 6, 8) to adapt to different system clock frequencies.
- Conversion time for one A/D conversion: 15 µs ... 50 µs
- Differential non-linearity : DLe ±1 LSB.
- Integral non-linearity : ILe ±2 LSB.
- Offset error : OSe ±2LSB.
- Gain error : Ge ±0.4 %.
- Absolute voltage error : Ae ±3 LSB.
- Channel to channel matching : Mctc ±1LSB.
- Crosstalk between analog inputs : Ct < –60dB. @100 kHz.
- Monotonic and no missing codes.
- Separated analog ($AV_{DD}$, $AV_{SS}$) and digital ($V_{DD}$, $V_{SS}$) supply voltages.
- Reference voltage at two special pins : $AV_{REF-}$ and $AV_{REF+}$.

For further information on the ADC characteristics, refer to the "DC CHARACTERISTICS" section.

### 6.1.1 Functional description:

**Table 9. A/D Special Function Registers**

| SYMBOL | NAME | ACCESS |
|---|---|---|
| ADCON | A/D control register | read/write |
| ADPSS | Analog port scan-select register | read/write |
| ADRSLn | 8 A/D result registers, the 8 lower bits (n: 0...7) | read only |
| ADRSH | A/D result register, the 2 higher bits | read only |
| P5 | Digital input port (shared with analog inputs) | read only |

### A/D Control Register ADCON

The Special Function Register ADCON contains control and status bits for the A/D Converter peripheral block. The reset value of ADCON is (00000000). Its hardware address is D7H. ADCON is not bit addressable.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **ADCON (D7H)** | ADPR1 | ADPR0 | ADPOS | ADINT | ADSST | ADCSA | ADSRE | ADSFE |

**Figure 14. ADC control register.**

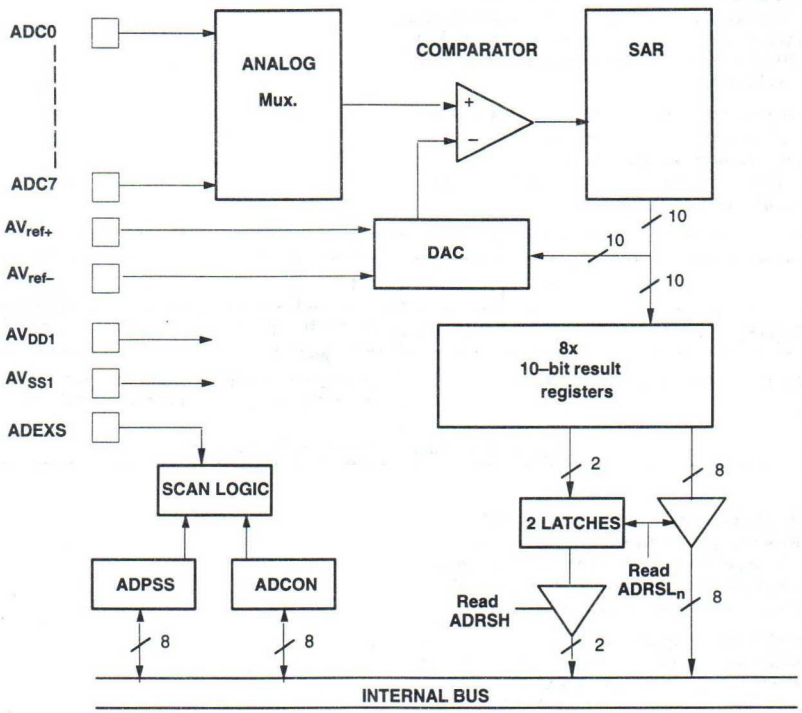



**Figure 15. Functional diagram of AD converter.**

## Table 10. Description of ADCON bits

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| ADCON.7 | ADPR1 | Control bit for the prescaler. |
| ADCON.6 | ADPR0 | Control bit for the prescaler.<br>ADPR1=0 ADPR0=0 Prescaler divides by 2 (default by reset)<br>ADPR1=0 ADPR0=1 Prescaler divides by 4<br>ADPR1=1 ADPR0=0 Prescaler divides by 6<br>ADPR1=1 ADPR0=1 Prescaler divides by 8 |
| ADCON.5 | ADPOS | ADPOS is reserved for future use. Must be '0' if ADCON is written. |
| ADCON.4 | ADINT | ADC interrupt flag. This flag is set when all selected analog inputs are converted, as well in continuous scan as in one-time scan mode. An interrupt is invoked if this interrupt is enabled. ADINT must be cleared by software. It cannot be set by software. |
| ADCON.3 | ADSST | ADC start and status. Setting this bit by software or by hardware (via ADEXS input) starts the A/D conversion of the selected analog inputs. ADSST stays a 'one' in continuous scan mode. In one-time scan mode, ADSST is cleared by hardware when the last selected analog input channel has been converted. As long as ADSST is '1', new start commands to the ADC-block are ignored.<br>An A/D conversion in progress is aborted if ADSST is cleared by software. |
| ADCON.2 | ADCSA | 1 = Continuous scan of the selected analog inputs after a start of an A/D conversion.<br>0 = One-time scan of the selected analog inputs after a start of an A/D conversion. |
| ADCON.1 | ADSRE | 1 = A rising edge at input ADEXS will start the A/D conversion and generate a capture signal.<br>0 = A rising edge at input ADEXS has no effect. |
| ADCON.0 | ADSFE | 1 = A falling edge at input ADEXS will start the A/D conversion and generate a capture signal.<br>0 = A falling edge at input ADEXS has no effect. |

## A/D Input Port Scan-Select Register ADPSS

The Special Function Register ADPSS contains control bits to select the analog input channel(s) to be scanned for A/D conversion. The reset value of ADPSS is (00000000). Its hardware address is E7H. ADPSS is not bit addressable.

If all bits are '0' then no A/D conversion can be started. If ADPSS is written while an A/D conversion is in progress (ADSST in the ADCON register is '1') then the autoscan loop with the previous selected analog inputs is completed first. The next autoscan loop is performed with the new selected analog inputs.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **ADPSS (E7H)** | ADPSS7 | ADPSS6 | ADPSS5 | ADPSS4 | ADPSS3 | ADPSS2 | ADPSS1 | ADPSS0 |

ADPSS7–0 For each individual bit position: 0 = The corresponding analog input is skipped in the auto-scan loop.
1 = The corresponding analog input is included in the auto-scan loop.

**Figure 16. A/D input port scan-select register.**

## A/D Result Registers ADRSLn and ADRSH:

The binary result code of A/D conversions is accessed by these Special Function Registers. The result SFR are read only registers. The read value after reset is indeterminate. Their data are not affected by chip reset. They are not bit addressable.

There are 8 Special Function Registers ADRSLn (ADRSL0...ADRSL7) – A/D Result Low byte – and one general SFR ADRSH – A/D Result High byte – . Each of ADRSLn is associated with the coincidently indexed analog input channel ADCn (ADC0/P5.0...ADC7/P5.7). Reading an ADRSLn register by software copies at the same time the two highest bits of the 10-bit conversion result into two latches, thus preserving them until the next read of any ADRSLn register. These two latches form bit positions 0 and 1 of SFR ADRSH, the upper 6 bits of ADRSH are always read as '0'.

Thus it is ensured to get the 10-bit result of the same single A/D conversion by reading any register ADRSLn first and after it the register ADRSH.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **ADRSLn** | ADRSn.7 | ADRSn.6 | ADRSn.5 | ADRSn.4 | ADRSn.3 | ADRSn.2 | ADRSn.1 | ADRSn.0 |

(n: 0...7)

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **ADRSH** | 0 | 0 | 0 | 0 | 0 | 0 | ADRSn.9 | ADRSn.8 |

**Figure 17. A/D Result Registers.**

### Digital Input Port Register P5

Port 5 Special Function Register P5 always represents the binary value of the logic level at input pins P5.0/ADC0...P5.7/ADC7. P5 is not affected by chip reset. P5 is a read only register. Its hardware address is C7H. P5 is not bit addressable.

Reading Special Function Register P5 does not affect A/D conversions. But it is recommended to use the digital input port function of the hardware Port 5 only as an alternative to analog input voltage conversions. Simultaneous mixed operation is discouraged for the sake of A/D conversion result reliability and accuracy.

For further information on Port 5, refer to the "I/O facilities" section.

For further information on A/D Special Function Registers, refer to the "Internal Data Memory" section.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| P5 (C7H) | P5.7 | P5.6 | P5.5 | P5.4 | P5.3 | P5.2 | P5.1 | P5.0 |

**Figure 18. Digital input port register P5.**

### Reset

After a RESET of the microcontroller the ADCON and ADPSS register bits are initialized to zero. Registers ADRSLn and ADRSH are not initialized by a RESET.

### Idle and Power-down Mode

The A/D Converter is active only when the microcontroller is in normal operating mode. If the Idle or Power-down Mode is activated, then the ADC is switched off and put into a power saving idle state – a conversion in progress is aborted, a previously set ADSST flag is cleared and the internal clock is halted. The conversion result registers are not affected.

The interrupt flag ADINT will not be set by activation of Idle or Power-down Mode. A previously set flag ADINT will not be cleared by the hardware. (Note: ADINT cannot be cleared by hardware at all, except for a RESET – it must be cleared by the user software.)

After a wakeup from Idle or Power-down Mode a set flag ADINT indicates that at least one autoscan loop was finished completely before the microcontroller was put into the respective power reduction mode and it indicates that the stored result data may be fetched now – if desired.

For further information on Idle and Power-down Mode, refer to the "Power reduction modes" section.

### Timing

A programmable prescaler is controlled by the bits ADPR1 and ADPR0 in register ADCON to adapt the conversion time for different microcontroller clock frequencies.

Table 11 shows conversion times (tconv) for one A/D conversion at some convenient system clock frequencies (fclk) and ADC prescaler divisors (m), which are user selectable by the bits ADCON.7/ADPR1 and ADCON.6/ADPR0.

For conversion times outside the limits for tconv the specified ADC characteristics are not guaranteed; (prohibited conversion times are put in brackets):

**Table 11. Conversion time configuration examples (tconv/μs)**

| m | $f_{CLK}$ | | | |
|---|---|---|---|---|
| | 6 MHz | 8 MHz | 12 MHz | 16 MHz |
| 2 | 26 | 19.5 | [13] | [9.75] |
| 4 | 50 | 37.5 | 25 | 18.75 |
| 6 | [74] | [55.5] | 37 | 27.75 |
| 8 | [98] | [73.5] | 49 | 36.75 |

Conversion time tconv = (6 m + 1) machine cycles

A conversion time tconv consists of one sample time period (which equals two bit conversion times), 10 bit conversion time periods and one machine cycle to store the result.

After result storage an extra initializing time period follows to select the next analog input channel (according to the contents of SFR ADPSS), before the input signal is sampled.

Thus the time period between two adjacent conversions within an autoscan loop is larger than the pure time tconv. This autoscan cycle time is ( 7 m ) machine cycles.

At the start of an autoscan conversion the time between writing to SFR ADCON and the first analog input signal sampling depends on the current prescaler value (m) and the relative time offset between this write operation and the internal (divided) ADC clock. This gives a variation range for the A/D conversion start time of ( m / 2 ) machine cycles.

### 6.6.2 Configuration and Operation

Every A/D conversion is an autoscan conversion. The two user selectable general operation modes are continuous scan and one-time scan mode.

The desired analog input port channel/s for conversion is/are selected by programming A/D input port scan-select bits in SFR ADPSS. An analog input channel is included in the autoscan loop if the corresponding bit in ADPSS is 1, a channel is skipped if the corresponding bit in ADPSS is 0.

An autoscan is always started according to the lowest bit position of ADPSS that contains a 1.

An autoscan conversion is started by setting the flag ADSST in register ADCON either by software or by an external start signal at input pin ADEXS, if enabled. Either no edge (external start totally disabled), a rising edge or/and a falling edge of ADEXS is selectable for external conversion start by the bits ADSRE and ADSFE in register ADCON.

After completion of an A/D conversion the 10-bit result is stored in the corresponding 10-bit buffer register. Then the next analog input is selected according to the next higher set bit position in ADPSS, converted and stored, and so on. When the result of the last conversion of this autoscan loop is stored, flag ADCON.4/ADINT, the ADC interrupt flag, is set. It is not cleared by interrupt hardware – it must be cleared by software.

In continuous scan mode (ADCON.2/ADCSA=1) the ADC start and status flag ADCON.3/ADSST retains the set state and the autoscan loop restarts from the beginning. In one-time scan mode (ADCSA=0) conversions stop after the last selected analog input was converted, ADINT is set and ADSST is cleared automatically.

ADSST cannot be set (neither externally nor by software) as long as ADINT=1, i.e. as long as ADINT is set, a new conversion start – by setting flag ADSST – is inhibited; actually it is only delayed until ADINT is cleared.

(If a '1' is written to ADSST while ADINT=1, this new value is internally latched and preserved, not setting ADSST until ADCON.4/ADINT=0. In this state, a read of SFR ADCON will display ADCON.3/ADSST=0, because always the effective ADC status is read.)

Note that under software control the analog inputs can also be converted in arbitrary order, when one-time scan mode is selected and in SFR ADPSS only one bit is set at a time. In this case ADINT is set and ADSST is cleared after every conversion.

### 6.6.3 Resolution and Characteristics

The ADC system has its own analog supply pins $AV_{DD}$ and $AV_{SS}$. It is referenced by two special reference voltage input pins sourcing the resistance ladder of the DAC: $AV_{ref+}$ and $AV_{ref-}$. The voltage between $AV_{REF+}$ and $AV_{REF-}$ defines the full-scale range. Due to the 10-bit resolution the full scale range is divided into 1024 unit steps. The unit step voltage is 1 LSB, which is typically 5 mV ($AV_{ref+}$ = 5.12 V, $AV_{ref-}$ = 0 V = $AV_{SS}$).

The DAC's resistance ladder has 1023 equally spaced taps, separated by a unit resistance 'R'. The first tap is located 0.5 x R above $AV_{ref-}$, the last tap is located 1.5 x R below $AV_{ref+}$. This results in a total ladder resistance of 1024 x R. This structure ensures that the DAC is monotonic and results in a symmetrical quantization error. For input voltages between $AV_{ref-}$ and ($AV_{ref-}$ + 1/2 LSB) the 10-bit conversion result code will be 00 0000 0000 B = 000H = 0D. For input voltages between ($AV_{ref+}$ – 3/2 LSB) and $AV_{ref+}$ the 10-bit conversion result code will be 11 1111 1111 B = 3FFH = 1023D.

The result code corresponding to an analog input voltage ($AV_{in}$) can be calculated from the formula:

$$\text{ResultCode} = 1024 \times \frac{AV_{IN} - AV_{ref-}}{AV_{ref+} - AV_{ref-}}$$

The analog input voltage should be stable when it is sampled for conversion. At any times the input voltage slew rate must be less than 10 V/ms (5 V conversion range) in order to prevent an undefined result.

This maximum input voltage slew rate can be ensured by an RC low pass filter with R = 2k2 and C = 100 nF. The capacitor between analog input pin and analog ground pin shall be placed close to the pins in order to have maximum effect in minimizing input noise coupling.

## 6.7 Timer/Counters

The P8xCE558 contains three 16-bit timer/event counters: Timer 0, Timer 1 and Timer T2 and one 8-bit timer, T3. Timer 0 and Timer 1 may be programmed to carry out the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests

### 6.7.1 Timer 0 and Timer 1

Timers 0 and 1 each have a control bit in SFR TMOD that selects the timer or counter function of the corresponding timer.

In the timer function, the register is incremented every machine cycle. Thus, one can think of it as counting machine cycles. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

In the counter function, the register is incremented in response to a 1-to-0 transition at the corresponding external input pin, T0 or T1. In this function, the external input is sampled during S5P2 of every machine cycle. When the samples show a HIGH in one cycle and a LOW in the next cycle, the counter is incremented. Thus, it takes two machine cycles (24 oscillator periods) to recognize a 1-to-0 transition. There are no restrictions on the duty cycle of the external input signal, but to insure that a given level is sampled at least once before it changes, it should be held for at least one full machine cycle.

Timer 0 and Timer 1 can be programmed independently to operate in one of four modes:

- **Mode 0:**
  8-bit timer or 8-bit counter each with divide-by-32 prescaler
- **Mode 1:**
  16-bit time-interval or event counter
- **Mode 2:**
  8-bit time-interval or event counter with automatic reload upon overflow
- **Mode 3:**
  –Timer 0: one 8-bit time-interval or event counter and one 8-bit time-interval counter
  –Timer 1: stopped

When Timer 0 is in Mode 3, Timer 1 can be programmed to operate in Modes 0, 1 or 2 but cannot set an interrupt request flag or generate an interrupt. However the overflow from Timer 1 can be used to pulse the serial port baud-rate generator.

With a 16 MHz crystal, the counting frequency of these timer/counters is as follows:

- In the timer function, the timer is incremented at a frequency of 1.33 MHz – a division by 12 of the system clock frequency
- 0 Hz to an upper limit of 0.66 MHz (1/24 of the system clock frequency) when programmed for external inputs

Both internal and external inputs can be gated to the counter by a second external source for directly measuring pulse durations.

When configured as a counter, the register is incremented on every falling edge on the corresponding input pin, T0 or T1. The incremented register value can be read earliest during the second machine cycle after that one, during which the incrementing pulse occurred.

The counters are started and stopped under software control. Each one sets its interrupt request flag when it overflows from all HIGHs to all LOWs (or automatic reload value), with the exception of mode 3 as previously described.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **TMOD (89H)** | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 |
| | Timer 1 | | | | Timer 0 | | | |

**Figure 19. Timer/Counter mode control (TMOD) register.**

## Table 12. Description of TMOD bits

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| Gate | TMOD.7<br>TMOD.3 | Gating control when set. Timer/Counter "x" is enabled only while "$\overline{INTx}$" pin is high and "TRx" control pin is set. When cleared Timer "x" is enabled whenever "TRx" control bit is set. |
| C/T | TMOD.6<br>TMOD.2 | Timer or Counter Selector cleared for Timer operation (input from internal system clock). Set for Counter operation (input from "Tx" input pin). |
| M1<br>M0 | TMOD.5<br>TMOD.1<br>TMOD.4<br>TMOD.0 | Timer 0, Timer 1 mode select see Table 13. |

## Table 13. Timer 0 / Timer 1 operation select

| M1 | M0 | OPERATING |
|---|---|---|
| 0 | 0 | 8048 Timer "TLx" serves as 5-bit prescaler. |
| 0 | 1 | 16-bit Timer/Counter "THx" and "TLx" are cascaded; there is no prescaler. |
| 1 | 0 | 8-bit auto-reload Timer/Counter "THx" holds a value which is to be reloaded into "TLx" each time it overflows. |
| 1 | 1 | (Timer 0) TL0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits. TH0 is an 8-bit timer only controlled by Timer 1 control bits. |
| 1 | 1 | (Timer 1) Timer/Counter 1 stopped. |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **TCON (88H)** | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |

**Figure 20. Timer/Counter mode control (TCON) register.**

**Table 14. Description of TCON bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| TF1 | TCON.7 | Timer 1 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine. |
| TR1 | TCON.6 | Timer 1 run control bit. Set/cleared by software to turn Timer/Counter on/off. |
| TF0 | TCON.5 | Timer 0 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine. |
| TR0 | TCON.4 | Timer 0 run control bit. Set/cleared by software to turn Timer/Counter on/off. |
| IE1 | TCON.3 | Interrupt 1 edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed. |
| IT1 | TCON.2 | Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts. |
| IE0 | TCON.1 | Interrupt 0 edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed. |
| IT0 | TCON.0 | Interrupt 0 type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts. |

### 6.7.2 Timer T2

Timer T2 is a 16 bit timer/counter which has capture and compare facilities. The operational diagram is shown in Figure 21.

The 16 bit timer/counter is clocked via a prescaler with a programmable division factor of 1, 2, 4 or 8. The input of the prescaler is clocked with 1/12 of the clock frequency, or by an external source connected to the T2 input, or it is switched off. The maximum repetition rate of the external clock source is $f_{CLK}/12$, twice that of Timer 0 and Timer 1. The prescaler is incremented on a rising edge. It is cleared if its division factor or its input source is changed, or if the timer/counter is reset (see also Figure 22: TM2CON). T2 is readable 'on the fly', without any extra read latches; this means that software precautions have to be taken against misinterpretation at overflow from least to most significant byte while T2 is being read. T2 is not loadable and is reset by the RST signal or at the positive edge of the input signal RT2, if enabled. In the Idle or Power-down Mode the timer/counter and prescaler are reset and halted.

T2 is connected to four 16-bit Capture Registers: CT0, CT1, CT2 and CT3. A rising or falling edge on the inputs CT0I, CT1I, CT2I or CT3I (alternative function of Port 1) results in loading the contents of T2 into the respective Capture Registers and an interrupt request.

Using the Capture Register CTCON (see Figure 23), these inputs may invoke capture and interrupt request on a positive, a negative edge or on both edges. If neither a positive nor a negative edge is selected for capture input, no capture or interrupt request can be generated by this input.

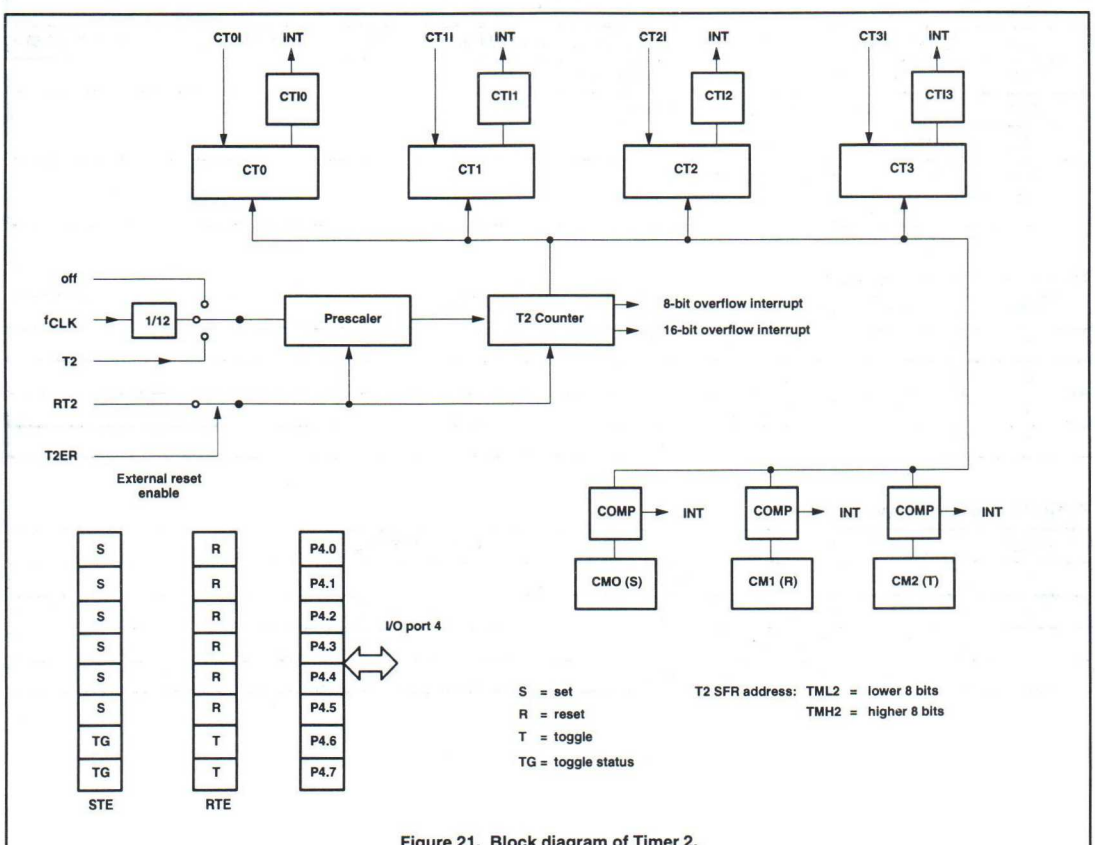



**Figure 21. Block diagram of Timer 2.**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **TM2CON (EAH)** | T2IS1 | T2IS0 | T2ER | T2BO | T2P1 | T2P0 | T2MS1 | T2MS0 |

**Figure 22. T2 control register (TM2CON).**

**Table 15. Description of TM2CON bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| T2IS1 | TM2CON.7 | Timer T2 16-bit overflow interrupt select |
| T2IS0 | TM2CON.6 | Timer T2 byte overflow interrupt select |
| T2ER | TM2CON.5 | Timer T2 external reset enable. When this bit is set, Timer T2 may be reset by a rising edge on RT2 (P1.5). |
| T2BO | TM2CON.4 | Timer T2 byte overflow interrupt flag |
| T2P1 | TM2CON.3 | Timer T2 prescaler select |
| T2P0 | TM2CON.2 | |
| T2MS1 | TM2CON.1 | Timer T2 mode select |
| T2MS0 | TM2CON.0 | |

**Table 16. Timer 2 prescaler select**

| T2P1 | T2P0 | TIMER T2 CLOCK |
|---|---|---|
| 0 | 0 | Clock source |
| 0 | 1 | Clock source/2 |
| 1 | 0 | Clock source/4 |
| 1 | 1 | Clock source/8 |

**Table 17. Timer 2 mode select**

| T2MS1 | T2MS0 | MODE SELECTED |
|---|---|---|
| 0 | 0 | Timer T2 halted (off) |
| 0 | 1 | T2 clock source = $f_{CLK}/12$ |
| 1 | 0 | Test mode; do not use |
| 1 | 1 | T2 clock source = pin T2 |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **CTCON (E8H)** | CTN3 | CTP3 | CTN2 | CTP2 | CTN1 | CTP1 | CTN0 | CTP0 |

**Figure 23. Capture control register (CTCON).**

**Table 18. Description of CTCON bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| CTN3 | CTCON.7 | Capture Register 3 triggered by a falling edge on CT3I |
| CTP3 | CTCON.6 | Capture Register 3 triggered by a rising edge on CT3I |
| CTN2 | CTCON.5 | Capture Register 2 triggered by a falling edge on CT2I |
| CTP2 | CTCON.4 | Capture Register 2 triggered by a rising edge on CT2I |
| CTN1 | CTCON.3 | Capture Register 1 triggered by a falling edge on CT1I |
| CTP1 | CTCON.2 | Capture Register 1 triggered by a rising edge on CT1I |
| CTN0 | CTCON.1 | Capture Register 0 triggered by a falling edge on CT0I |
| CTP0 | CTCON.0 | Capture Register 0 triggered by a rising edge on CT0I |

The contents of the Compare Registers CM0, CM1 and CM2 are continuously compared with the counter value of Timer T2. When a match occurs, an interrupt may be invoked. A match of CM0 sets the bits 0–5 of Port 4, a CM1 match resets these bits and a CM2 match toggles bits 6 and 7 of Port 4, provided these functions are enabled by the STE respectively RTE registers. A match of CM0 and CM1 at the same time results in resetting bits 0–5 of Port 4. CM0, CM1 and CM2 are reset by the RSTIN signal.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **TM2IR (C8H)** | T2OV | CMI2 | CMI1 | CMI0 | CTI3 | CTI2 | CTI1 | CTI0 |

**Figure 24. Interrupt flag register (TM2IR).**

**Table 19. Description of TM2IR bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| T2OV | TM2IR.7 | Timer T2 16-bit overflow interrupt flag |
| CMI2 | TM2IR.6 | CM2 interrupt flag |
| CMI1 | TM2IR.5 | CM1 interrupt flag |
| CMI0 | TM2IR.4 | CM0 interrupt flag |
| CTI3 | TM2IR.3 | CT3 interrupt flag |
| CTI2 | TM2IR.2 | CT2 interrupt flag |
| CTI1 | TM2IR.1 | CT1 interrupt flag |
| CTI0 | TM2IR.0 | CT0 interrupt flag |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| STE (EEH) | TG47 | TG46 | SP45 | SP44 | SP43 | SP42 | SP41 | SP40 |

**Figure 25. Set enable register (STE).**

**Table 20. Description of STE bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| TG47 | STE.7 | If "1" then P4.7 is reset on the next toggle, if LOW P4.7 is set on the next toggle |
| TG46 | STE.6 | If "1" then P4.6 is reset on the next toggle, if LOW P4.6 is set on the next toggle |
| SP45 | STE.5 | If "1" then P4.5 is set on a match between CM0 and Timer T2 |
| SP44 | STE.4 | If "1" then P4.4 is set on a match between CM0 and Timer T2 |
| SP43 | STE.3 | If "1" then P4.3 is set on a match between CM0 and Timer T2 |
| SP42 | STE.2 | If "1" then P4.2 is set on a match between CM0 and Timer T2 |
| SP41 | STE.1 | If "1" then P4.1 is set on a match between CM0 and Timer T2 |
| SP40 | STE.0 | If "1" then P4.0 is set on a match between CM0 and Timer T2 |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| RTE (EFH) | TP47 | TP46 | RP45 | RP44 | RP43 | RP42 | RP41 | RP40 |

**Figure 26. Reset/Toggle enable register (RTE).**

**Table 21. Description of RTE bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| TP47 | RTE.7 | If "1" then P4.7 toggles on a match between CM2 and Timer T2 |
| TP46 | RTE.6 | If "1" then P4.6 toggles on a match between CM2 and Timer T2 |
| RP45 | RTE.5 | If "1" then P4.5 toggles on a match between CM1 and Timer T2 |
| RP44 | RTE.4 | If "1" then P4.4 toggles on a match between CM1 and Timer T2 |
| RP43 | RTE.3 | If "1" then P4.3 toggles on a match between CM1 and Timer T2 |
| RP42 | RTE.2 | If "1" then P4.2 toggles on a match between CM1 and Timer T2 |
| RP41 | RTE.1 | If "1" then P4.1 toggles on a match between CM1 and Timer T2 |
| RP40 | RTE.0 | If "1" then P4.0 toggles on a match between CM1 and Timer T2 |

For more information concerning the TM2CON, CTCON, TM2IR and the STE/RTE registers see IC20 handbook, chapter "80C51 family hardware description".

Port 4 can be read and written by software without affecting the toggle, set and reset signals. At a byte overflow of the least significant byte, or at a 16-bit overflow of the timer/counter, an interrupt sharing the same interrupt vector is requested. Either one or both of these overflows can be programmed to request an interrupt.

All interrupt flags must be reset by software.

## 6.8 Watchdog Timer T3

In addition to Timer T2 and the standard timers, a watchdog timer (T3) consisting of an 11-bit prescaler and an 8-bit timer is also incorporated (see Figure 27).

The timer is incremented every 1.5 ms, derived from the system clock frequency of 16 MHz by the following:

$$f_{timer} = \frac{f_{CLK}}{12 \times 2048}$$

When a timer overflow occurs, the microcontroller is reset and a reset output pulse is generated at pin RSTOUT. Also the PLL control register is reset.

To prevent a system reset the timer must be reloaded in time by the application software. If the processor suffers a hardware/software malfunction, the software will fail to reload the timer. This failure will produce a reset upon overflow thus preventing the processor running out of control.

The watchdog timer can only be reloaded if the condition flag WLE = PCON.4 has been previously set by software.

At the moment the counter is loaded the condition flag is automatically cleared.

The time interval between the timer's reloading and the occurrence of a reset depends on the reloaded value. For example, this may range from 1.5 ms to 0.375 s when using an oscillator frequency of 16 MHz.

In the Idle state the watchdog timer and reset circuitry remain active.

The watchdog timer is controlled by the watchdog enable pin ($\overline{EW}$). A LOW level enables the watchdog timer and disables the Power-down Mode. A HIGH level disables the watchdog timer and enables the Power-down Mode.

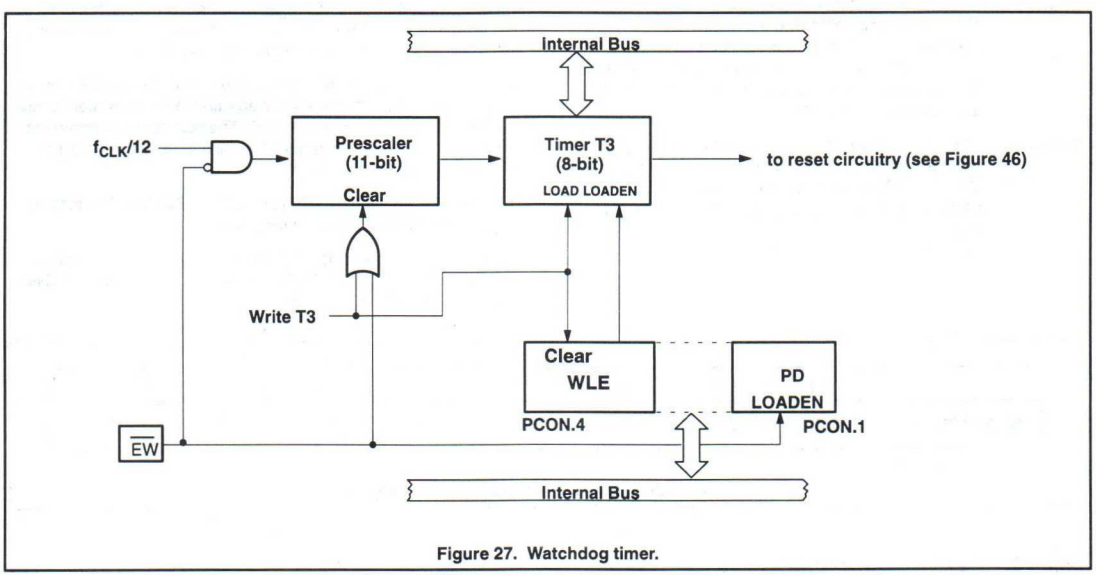



**Figure 27. Watchdog timer.**

## 6.9 Serial I/O

The P8xCE558 is equipped with two independent serial ports: SIO0 and SI01. SIO0 is the full duplex UART port, identical to the PCB80C51 serial port. SIO1 is an I²C-bus serial I/O interface with byte oriented master and slave functions.

### 6.9.1 SIO0 (UART)

SIO 0 is a full duplex serial I/O port – it can transmit and receive simultaneously. This serial port is also receive-buffered. It can commence reception of a second byte before the previously received byte has been read from the receive register. If, however, the first byte has still not been read by the time reception of the second byte is complete, one of the bytes will be lost. The SIO0 receive and transmit registers are both accessed via the S0BUF special function register. Writing to S0BUF loads the transmit register, and reading S0BUF accesses to a physically separate receive register. SIO0 can operate in 4 modes:

**Mode 0:** Serial data is transmitted and received through RXD. TXD outputs the shift clock. 8 data bits are transmitted/received (LSB first). The baud rate is fixed at 1/12 of the oscillator frequency. A write into S0CON should be avoided during a transmission to avoid spikes on RXD/TXD.

**Mode 1:** 10 bits are transmitted via TXD or received through RXD: a start bit (0), 8 data bits (LSB first), and a stop bit(1). On receive, the stop bit is put into RB8 (S0CON special function register). The baud rate is variable.

**Mode 2:** 11 bits are transmitted through TXD or received through RXD: a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). On transmit, the 9th data bit (TB8 in S0CON) can be assigned the value of 0 or 1. With nominal software, TB8 can be the parity bit (P in PSW). During a receive, the 9th data bit is stored in RB8 (S0CON), and the stop bit is ignored. The baud rate is programmable to either 1/32 or 1/64 of the oscillator frequency.

**Mode 3:** 11 bits are transmitted through TXD or received through RXD: a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). Mode 3 is the same as Mode 2 except the baud rate which is variable in Mode 3.

In all four modes, transmission is initiated by any instruction that writes to the S0BUF function register. Reception is initiated in Mode 0 when RI = 0 and REN = 1. In the other three modes, reception is initiated by the incoming start bit provided that REN = 1.

Modes 2 and 3 are provided for multiprocessor communications. In these modes, 9 data bits are received with the 9th bit written to RB8. The 9th bit is followed by the stop bit. The port can be programmed so that with receiving the stop bit, the serial port interrupt will be activated if, and only if RB8 = 1.

This feature is enabled by setting bit SM2 in S0CON. This feature may be used in multiprocessor systems.

For more information about how to use the UART in combination with the registers S0CON, PCON, IEN0, S0BUF and Timer register refer to the 80C51 Data Handbook IC20.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **S0CON (98H)** | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |

**Figure 28. Serial port control (S0CON) register.**

**Table 22. Description of S0CON bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| SM0 | S0CON.7 | This bit is used to select the serial port mode. See Table 23. |
| SM1 | S0CON.6 | This bit is used to select the serial port mode. See Table 23. |
| SM2 | S0CON.5 | Enables the multiprocessor communication feature in modes 2 and 3. In mode 2 or 3, if SM2 is set to 1, then RI will not be activated if the received 9th data bit (RB8) is 0. In mode 1, if SM2 = 1, then RI will not be activated if a valid stop bit was not received. In mode 0, SM2 should be 0. |
| REN | S0CON.4 | Enables serial reception. Set by software to enable reception. Clear by software to disable reception. |
| TB8 | S0CON.3 | The 9th data bit that will be transmitted in modes 2 and 3. Set or clear by software as desired. |
| RB8 | S0CON.2 | In modes 2 and 3, RB8 is the 9th data bit that was received. In mode 1, if SM2 = 0, RB8 is the stop bit that was received. In mode 0, RB8 is not used. |
| TI | S0CON.1 | The transmit interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software. |
| RI | S0CON.0 | The receive interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software. |

**Table 23. Description of S0CON bits**

| SM0 | SM1 | MODE | DESCRIPTION | BAUD RATE |
|---|---|---|---|---|
| 0 | 0 | 0 | Shift register | $f_{CLK}/12$ |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{CLK}/64$ or $f_{CLK}/32$ |
| 1 | 1 | 3 | 9-bit UART | variable |

#### 6.9.2 SIO1 ($I^2C$-bus Interface)

The SIO1 of the P8xCE558 provides the fast-mode, which allows a fourthfold increase of the bitrate up to 400 kHz. Nevertheless it is downward compatible, i.e. it can be used in a 0 to 100 Kbit/s $I^2C$ bus system.

Except from the bit rate selection (see Table 25) and the timing of the SCL and SDA signals (see AC electrical characteristics in section 11) the SIO circuit is the same as described in detail in the 80C51 Data Handbook IC20 for the 8xC552 microcontroller.

The $I^2C$-bus is a simple bidirectional 2-wire bus for efficient inter-IC data exchange. Features of the $I^2C$-bus are:

- Only two bus lines are required: a serial clock line (SCL) and a serial data line (SDA)
- Each device connected to the bus is software addressable by a unique address
- Masters can operate as Master-transmitter or as Master-receiver
- It's a true multi-master bus including collision detection and arbitration to prevent data corruption if two or more masters simultaneously initiate data transfer
- Serial clock synchronization allows devices with different bit rates to communicate via the same serial bus
- ICs can be added to or removed from an $I^2C$-bus system without affecting any other circuit on the bus
- Fault diagnostics and debugging are simple; malfunctions can be immediately traced

For more information on the $I^2C$-bus specification (including fast-mode) please refer to the Philips publication number 9398 393 40011 and/or the 80C51 Data Handbook IC20.

The on-chip $I^2C$ logic provides a serial interface that meets the $I^2C$-bus specification, supporting all $I^2C$-bus modes of operation, they are:

- Master transmitter
- Master receiver
- Slave transmitter
- Slave receiver

The SI01 logic performs a byte oriented data transport, clock generation, address recognition and bus control arbitration are all controlled by hardware. Via two pins the external $I^2C$-bus is interfaced to the SIO1 logic:
SCL serial clock I/O and SDA serial data I/O, (see Special Function Register bit S1CON.6/ENS1 for enabling the SIO1 logic).

The SIO1 logic handles byte transfer autonomously. It keeps track of the serial transfers, and a status register (S1STA) reflects the status of SIO1 and the $I^2C$-bus.

Via the following four Special Function Registers the CPU interfaces to the $I^2C$ logic.

| | |
|---|---|
| S1CON | control register. Bit addressable by the CPU |
| S1STA | status register whose contents may be used as a vector to service routines. |
| S1DAT | data shift register. The data byte is stable as long as S1CON.3/SI=1. |
| S1ADR | slave address register. It's LSB enables/ disables general call address recognition. |

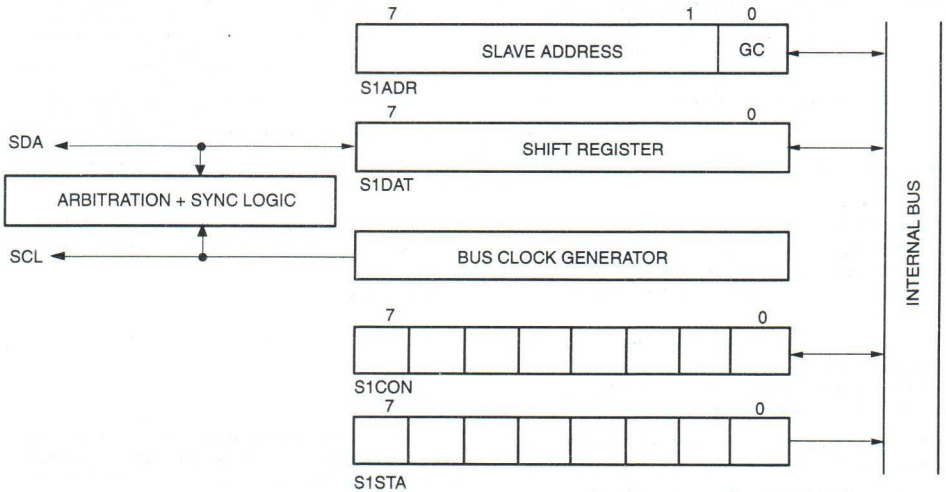



**Figure 29. Block diagram of $I^2C$ serial I/O interface.**

**The Control Register, S1CON:**
The CPU can read from and write to this 8-bit, directly addressable SFR. Two bits are affected by the SIO1 hardware: the SI bit is set when a serial interrupt is requested, and the STO bit is cleared when a STOP condition is present on the $I^2C$ bus. The STO bit is also cleared when ENS1 = 0.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **S1CON (D8H)** | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |

**Figure 30. Serial control (S1CON) register.**

## Table 24. Description of S1CON bits

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| CR2 | S1CON.7 | Clock rate bit 2, see Table 25. |
| ENS1 | S1CON.6 | ENS1 = 0: Serial I/O disabled and reset. SDA and SCL outputs are high-Z.<br>ENS1 = 1: Serial I/O enabled. |
| STA | S1CON.5 | START flag. When this bit is set in slave mode, the hardware checks the $I^2C$ bus and generates a START condition if the bus is free or after the bus becomes free. If the device operates in master mode it will generate a repeated START condition. |
| STO | S1CON.4 | STOP flag. If this bit is set in a master mode a STOP condition is generated. A STOP condition detected on the $I^2C$ bus clears this bit. This bit may also be set in slave mode in order to recover from an error condition. In this case no STOP condition is generated to the $I^2C$ bus, but the hardware releases the SDA and SCL lines and switches to the not selected receiver mode. The STOP flag is cleared by the hardware. |
| SI | S1CON.3 | Serial Interrupt flag. This flag is set, and an interrupt request is generated, after any of the following events occur:<br>– A START condition is generated in master mode.<br>– The own slave address has been received during AA = 1.<br>– The general call address has been received while S1ADR.0 and AA = 1.<br>– A data byte has been received or transmitted in master mode (even if arbitration is lost).<br>– A data byte has been received or transmitted as selected slave.<br>– A STOP or START condition is received as selected slave receiver or transmitter.<br>While the SI flag is set, SCL remains LOW and the serial transfer is suspended. SI must be reset by software. |
| AA | S1CON.2 | Assert Acknowledge flag. When this bit is set, an acknowledge is returned after any one of the following conditions:<br>– Own slave address is received.<br>– General call address is received (S1ADR.0 = 1).<br>– A data byte is received, while the device is programmed to be a master receiver.<br>– A data byte is received. while the device is a selected slave receiver.<br>When the bit is reset, no acknowledge is returned. Consequently, no interrupt is requested when the own address or general call address is received. |
| CR1<br>CR0 | S1CON.1<br>S1CON.0 | Clock rate bits 1 and 0, see Table 25. |

When SIO1 is in a master mode serial clock frequency is determined by the clock rate bits CR2, CR1 and CR0. The various bit rates are shown in Table 25.

**Table 25. Selection of $I^2C$-bus bit rate**

| CR2 | CR1 | CR0 | BIT RATE (kHz) at $f_{CLK}$ | |
|---|---|---|---|---|
| | | | 12MHz | 16MHz |
| 1 | 0 | 0 | 50 | 66.7 |
| 1 | 0 | 1 | 3.75 | 5 |
| 1 | 1 | 0 | 75 | 100 |
| 1 | 1 | 1 | 100 | – |
| 0 | 0 | 0 | 200 [1] | 266.7 [1] |
| 0 | 0 | 1 | 7.5 | 10 |
| 0 | 1 | 0 | 300 [1] | 400 [1] |
| 0 | 1 | 1 | 400 [1] | – |

**NOTE:**
1. These bit rates are for "fast-mode" $I^2C$ bus applications and cannot be used for bit rates up to 100 kbit/sec.

The data shown in Table 25 do not apply to SIO1 in a slave mode. In the slave modes, SIO1 will automatically synchronize with any clock frequency up to 400kHz.

**Serial status register S1STA**

S1STA is a read only register.

The contents of the status register may be used as a vector to a service routine. This optimizes the response time of the software and consequently that of the $I^2C$-bus.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **S1STA (D9H)** | SC4 | SC3 | SC2 | SC1 | SC0 | 0 | 0 | 0 |

**Figure 31. Serial status (S1STA) register.**

**Table 26. Description of S1STA bits**

| BIT | FUNCTION |
|---|---|
| S1STA.7 to 3 | 5-bit status code |
| S1STA.2 to 0 | These bits are held LOW (for service routine vector increment 8) |

The following is a list of the status codes:

**Table 27. MST/TRX mode**

| S1STA VALUE | DESCRIPTION |
|---|---|
| 08H | A START condition has been transmitted |
| 10H | A repeated START condition has been transmitted |
| 18H | SLA and W have been transmitted, ACK has been received |
| 20H | SLA and W have been transmitted, $\overline{ACK}$ received |
| 28H | DATA and S1DAT has been transmitted, ACK received |
| 30H | DATA and S1DAT has been transmitted, $\overline{ACK}$ received |
| 38H | Arbitration lost in SLA, R/W or DATA |

**Table 28. MST/REC mode**

| S1STA VALUE | DESCRIPTION |
|---|---|
| 38H | Arbitration lost while returning $\overline{ACK}$ |
| 40H | SLA and R have been transmitted, ACK received |
| 48H | SLA and R have been transmitted, $\overline{ACK}$ received |
| 50H | DATA has been received, ACK returned |
| 58H | DATA has been received, $\overline{ACK}$ returned |

**Table 29. SLV/REC mode**

| S1STA VALUE | DESCRIPTION |
|---|---|
| 60H | Own SLA and W have been received, ACK returned |
| 68H | Arbitration lost in SLA, R/W as MST. Own SLA and W have been received, $\overline{ACK}$ returned |
| 70H | General CALL has been received, ACK returned |
| 78H | Arbitration lost in SLA, R/W as MST. General call has been received |
| 80H | Previously addressed with own SLA. DATA byte received, ACK returned |
| 88H | Previously addressed with own SLA. DATA byte received, $\overline{ACK}$ returned |
| 90H | Previously addressed with general call. DATA byte has been received, ACK has been returned |
| 98H | Previously addressed with general call. DATA byte has been received, $\overline{ACK}$ has been returned |
| A0H | A STOP condition or repeated START condition has been received while still addressed as SLV/REC or SLV/TRX |

**Table 30. SLV/TRX mode**

| S1STA VALUE | DESCRIPTION |
|---|---|
| A8H | Own SLA and R have been received, ACK returned |
| B0H | Arbitration lost in SLA, R/W as MST. Own SLA and R have been received, ACK returned |
| B8H | DATA byte has been transmitted, ACK returned |
| C0H | DATA byte has been transmitted, $\overline{ACK}$ returned |
| C8H | Last DATA byte has been transmitted (AA = logic 0), ACK received |

**Table 31. Miscellaneous**

| S1STA VALUE | DESCRIPTION |
|---|---|
| 00H | Bus error during MST mode or selected SLV mode, due to an erroneous START or STOP condition |
| F8H | No relevant information available, SI not set |

Abbreviations used:

| | | |
|---|---|---|
| SLA | : | 7-bit slave address |
| R | : | Read bit |
| W | : | Write bit |
| ACK | : | Acknowledgement (acknowledge bit = 0) |
| $\overline{ACK}$ | : | Not acknowledgement (acknowledge bit = 1) |
| DATA | : | 8-bit data byte to or from $I^2C$-bus |
| MST | : | Master |
| SLV | : | Slave |
| TRX | : | Transmitter |
| REC | : | Receiver |

**The data shift register S1DAT**

This register contains the serial data to be transmitted or data which has been received. Bit 7 is transmitted or received first; i.e., data is shifted from right to left.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **S1DAT (DAH)** | S1DAT.7 | S1DAT.6 | S1DAT.5 | S1DAT.4 | S1DAT.3 | S1DAT.2 | S1DAT.1 | S1DAT.0 |

**Figure 32. Data shift register.**

**The address register S1ADR**

This 8-bit register may be loaded with the 7-bit slave address to which the controller will respond when programmed as a slave receiver/transmitter. The LSB (GC) is used to determine whether the general call address is recognized.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **S1ADR (DBH)** | SLA6 | SLA5 | SLA4 | SLA3 | SLA2 | SLA1 | SLA0 | GC |

**Figure 33. Address register.**

## Table 32. Description of S1ADR bits

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| SLA6 to 0 | S1ADR.7 to 1 | Own slave address |
| GC | S1ADR.0 | 0 = general call address is not recognized<br>1 = general call address is recognized |

## 6.10 Interrupt System

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronously to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. Interrupt response time in a single-interrupt system is in the range from 2.25µs to 6.75µs when using a 16MHz crystal. The latency time depends on the sequence of instructions executed directly after an interrupt request.

The P8xCE558 acknowledges interrupt requests from 15 sources as follows (see Figure 34):

- $\overline{INT0}$ and $\overline{INT1}$ external interrupts
- Timer 0 and Timer 1 internal timer/counter interrupts
- Timer 2 internal timer/counter byte and/or 16-bit overflow, 3 compare and 4 capture interrupts (or 4 additional external interrupts) [1]
- UART serial I/O port receive/transmit interrupt
- I²C-bus interface serial I/O interrupt
- ADC autoscan completion interrupt
- 'Seconds' timer interrupt SEC (ored with $\overline{INT1}$).

For details about seconds timer interrupts, please refer to chapter 6.13.4.

The External Interrupts $\overline{INT0}$ and $\overline{INT1}$ can each be either level-activated or transition-activated, depending on bits IT0 and IT1 in register TCON. The flags that actually generate these interrupts are bits IE0 and IE1 in TCON. When an external interrupt is generated, the corresponding request flag is cleared by the hardware when the service routine is vectored to only if the interrupt was transition-activated. If the interrupt was level-activated then the interrupt request flag remains set until the external interrupt pin $\overline{INTx}$ goes high. Consequently the external source has to hold the request active until the requested interrupt is actually generated. Then it has to deactivate the request before the interrupt service routine is completed, or else another interrupt will be generated. As these external interrupts are active LOW a "wire-ORing" of several interrupt sources to one input pin allows expansion.

The Timer 0 and Timer 1 Interrupts are generated by TF0 and TF1, which are set by a rollover in their respective timer/counter register (except for Timer 0 in Mode 3 of the serial interface). When a Timer interrupt is generated, the flag that generated it is cleared by the on-chip hardware when the service routine is vectored to.

The eight Timer/Counter T2 Interrupt sources are: 4 capture Interrupts [(1)], 3 compare interrupts and an overflow interrupt. The appropriate interrupt request flags must be cleared by software.

The UART Serial Port Interrupt is generated by the logical OR of RI and TI. Neither of these flags is cleared by hardware. The service routine will normally have to determine whether it was RI or TI that generated the interrupt, and the bit will have to be cleared by software.

The I²C Interrupt is generated by bit SI in register S1CON. This flag has to be cleared by software.

The ADC Interrupt is generated by bit ADINT, which is set when of all selected analog inputs to be scanned, the conversion is finished. ADINT must be cleared by software. It cannot be set by software.

The 'Seconds' timer Interrupt is generated by bit SECINT in register PLLCON. This flag has to be cleared by software. Note that the 'Seconds' timer can only be used with the 32 kHz PLL oscillator.

All of the bits that generate interrupts can be set or cleared by software, with the same result as though it had been set or cleared by hardware (except the ADC interrupt request flag ADINT, which cannot be set by software). That is, interrupts can be generated or pending interrupts can be cancelled in software.

The Interrupts X0, T0, X1, T1, SEC, S0 and S1 are capable to terminate the Idle Mode.

### Interrupt Enable Registers

Each interrupt source can be individually enabled or disabled by setting or clearing a bit in the interrupt enable special function registers IEN0 and IEN1. All interrupt sources can also be globally disabled by clearing bit EA in IEN0. The interrupt enable registers are described in Figures 34 and 36.

### Interrupt Priority Structure

Each interrupt source can be assigned one of two priority levels. Interrupt priority levels are defined by the interrupt priority special function registers IP0 and IP1. IP0 and IP1 are described in Figures 37 and 38.

Interrupt priority levels are as follows:
"0"—low priority
"1"—high priority

A low priority interrupt may be interrupted by a high priority interrupt. A high priority interrupt cannot be interrupted by any other interrupt source. If two requests of different priority occur simultaneously, the high priority level request is serviced. If requests of the same priority are received simultaneously, an internal polling sequence determines which request is serviced. Thus, within each priority level, there is a second priority structure determined by the polling sequence. This second priority structure is shown in Table 37.

### Interrupt Handling

The interrupt sources are sampled at S5P2 of every machine cycle. The samples are polled during the following machine cycle. If one of the flags was in a set condition at S5P2 of the previous machine cycle, the polling cycle will find it and the interrupt system will generate an LCALL to the appropriate service routine, provided this hardware- generated LCALL is not blocked by any of the following conditions:

1. An interrupt of higher or equal priority level is already in progress.
2. The current machine cycle is not the final cycle in the execution of the instruction in progress. (No interrupt request will be serviced until the instruction in progress is completed.)
3. The instruction in progress is RETI or any access to the interrupt priority or interrupt enable registers. (No interrupt will be serviced after RETI or after a read or write to IP0, IP1, IE0, or IE1 until at least one other instruction has been subsequently executed.)

**NOTE:**
1. If a capture register is unused and it's contents is of no interest, then the corresponding input pin CTnI/P1.n (n: 0...3) may be used as a (configurable) positive and/or negative edge triggered additional external interrupt input (INT2, INT3, INT4, INT5).

The polling cycle is repeated with every machine cycle, and the values polled are the values present at S5P2 of the previous machine cycle. Note that if an interrupt flag is active but is not being responded to because of one of the above conditions, and if the flag is inactive when the blocking condition is removed, then the blocked interrupt will not be serviced. Thus, the fact that the interrupt flag was once active but not serviced is not remembered. Every polling cycle is new.

The processor acknowledges an interrupt request by executing a hardware-generated LCALL to the appropriate service routine. In some cases it also clears the flag which generated the interrupt, and in others it does not. It clears the Timer 0, Timer 1, and external interrupt flags. An external interrupt flag (IE0 or IE1) is cleared only if it was transition-activated. All other interrupt flags are not cleared by hardware and must be cleared by the software. The LCALL pushes the contents of the program counter on to the stack (but it does not save the PSW) and reloads the PC with an address that depends on the source of the interrupt being vectored to as shown in Table 38.

Execution proceeds from the vector address until the RETI instruction is encountered. The RETI instruction clears the "priority level active" flip-flop that was set when this interrupt was acknowledged. It then pops the top two bytes from the stack and reloads the program counter. Execution of the interrupted program continues from where it was interrupted.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **IEN0 (A8H)** | EA | EAD | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 |

**Figure 34. Interrupt enable register (IEN0).**

**Table 33. Description of IEN0 bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| EA | IEN0.7 | Global enable/disable control<br>0 = No interrupt is enabled<br>1 = Any individually enabled interrupt will be accepted |
| EAD | IEN0.6 | Enable ADC interrupt |
| ES1 | IEN0.5 | Enable SIO1 ($I^2C$) interrupt |
| ES0 | IEN0.4 | Enable SIO0 (UART) interrupt |
| ET1 | IEN0.3 | Enable Timer 1 interrupt |
| EX1 | IEN0.2 | Enable External interrupt 1 / Seconds interrupt |
| ET0 | IEN0.1 | Enable Timer 0 interrupt |
| EX0 | IEN0.0 | Enable External interrupt 0 |

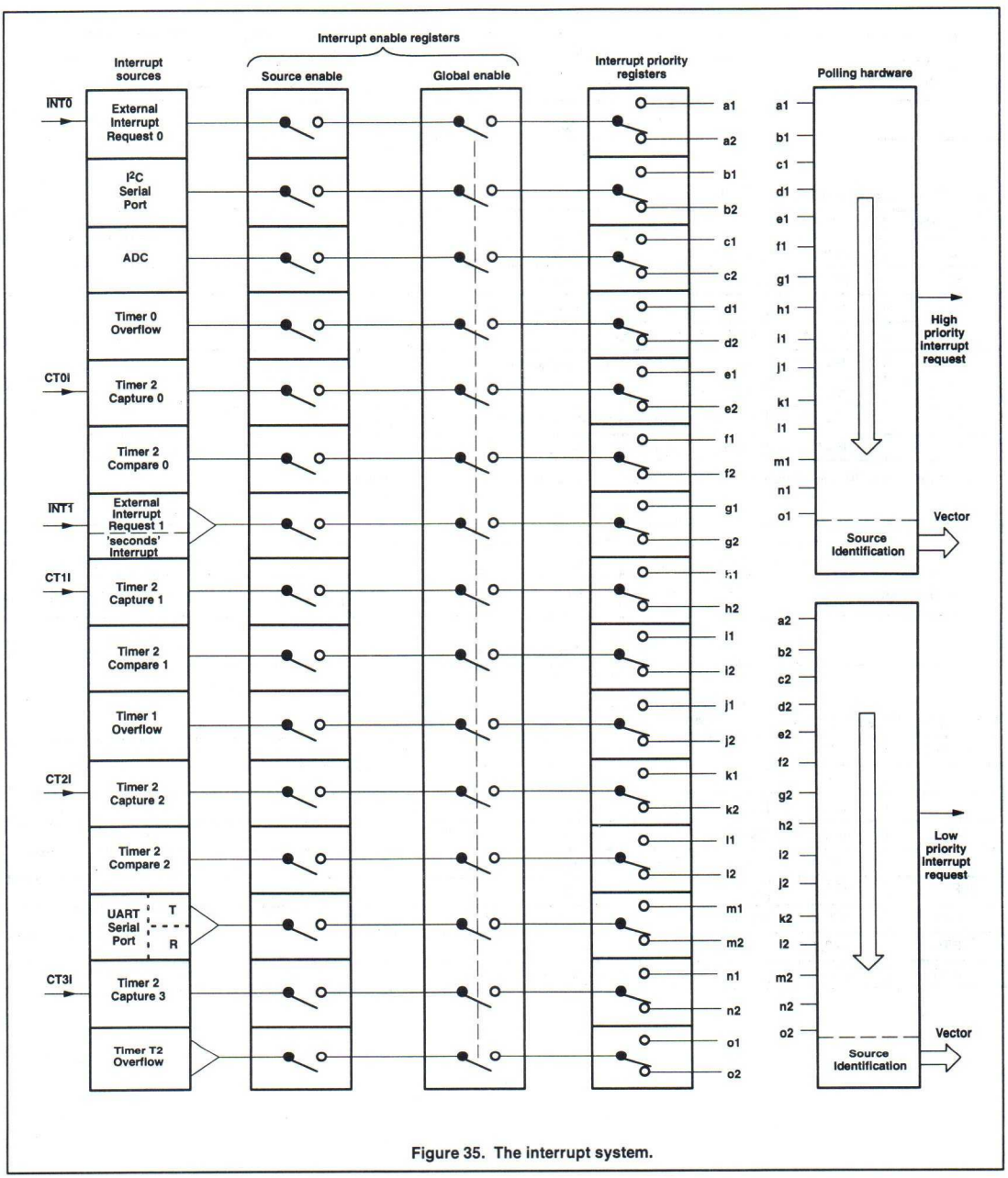



Figure 35. The interrupt system.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **IEN1 (E8H)** | ET2 | ECM2 | ECM1 | ECM0 | ECT3 | ECT2 | ECT1 | ECT0 |

**Figure 36. Interrupt enable register (IEN1).**

**Table 34. Description of IEN1 bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| ET2 | IEN1.7 | Enable T2 overflow interrupt(s) |
| ECM2 | IEN1.6 | Enable T2 comparator 2 interrupt |
| ECM1 | IEN1.5 | Enable T2 comparator 1 interrupt |
| ECM0 | IEN1.4 | Enable T2 comparator 0 interrupt |
| ECT3 | IEN1.3 | Enable T2 capture register 3 interrupt |
| ECT2 | IEN1.2 | Enable T2 capture register 2 interrupt |
| ECT1 | IEN1.1 | Enable T2 capture register 1 interrupt |
| ECT0 | IEN1.0 | Enable T2 capture register 0 interrupt |

If the enable bit is 0, then the interrupt is disabled, if the enable bit is 1, then the interrupt is enabled.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **IP0 (B8H)** | – | PAD | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 |

**Figure 37. Interrupt priority register (IP0).**

**Table 35. Description of IP0 bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| – | IP0.7 | Reserved for future use |
| PAD | IP0.6 | ADC interrupt priority level |
| PS1 | IP0.5 | SIO1 ($I^2C$) interrupt priority level |
| PS0 | IP0.4 | SIO0 (UART) interrupt priority level |
| PT1 | IP0.3 | Timer 1 interrupt priority level |
| PX1 | IP0.2 | External interrupt 1/Seconds interrupt priority level |
| PT0 | IP0.1 | Timer 0 interrupt priority level |
| PX0 | IP0.0 | External interrupt 0 priority level |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **IP1 (F8H)** | PT2 | PCM2 | PCM1 | PCM0 | PCT3 | PCT2 | PCT1 | PCT0 |

**Figure 38. Interrupt priority register (IP1).**

**Table 36. Description of IP1 bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| PT2 | IP1.7 | T2 overflow interrupt(s) priority level |
| PCM2 | IP1.6 | T2 comparator 2 interrupt priority level |
| PCM1 | IP1.5 | T2 comparator 1 interrupt priority level |
| PCM0 | IP1.4 | T2 comparator 0 interrupt priority level |
| PCT3 | IP1.3 | T2 capture register 3 interrupt priority level |
| PCT2 | IP1.2 | T2 capture register 2 interrupt priority level |
| PCT1 | IP1.1 | T2 capture register 1 interrupt priority level |
| PCT0 | IP1.0 | T2 capture register 0 interrupt priority level |

**Table 37. Interrupt Priority Structure**

| SOURCE | NAME | PRIORITY WITHIN LEVEL |
|---|---|---|
| | | (highest) ↑ |
| External interrupt 0 | X0 | |
| SIO1 ($I^2C$) | S1 | |
| ADC completion | ADC | |
| Timer 0 overflow | T0 | |
| Timer 2 capture 0 | CT0 | |
| Timer 2 compare 0 | CM0 | |
| External interrupt 1/Seconds interrupt | X1/SEC | |
| Timer 2 capture 1 | CT1 | |
| Timer 2 compare 1 | CM1 | |
| Timer 1 overflow | T1 | |
| Timer 2 capture 2 | CT2 | |
| Timer 2 compare 2 | CM2 | |
| SIO0 (UART) | S0 | |
| Timer 2 capture 3 | CT3 | |
| Timer 2 overflow | T2 | ↓ (lowest) |

**Table 38. Interrupt Vector Addresses**

| SOURCE | NAME | VECTOR ADDRESS |
|---|---|---|
| External interrupt 0 | X0 | 0003H |
| Timer 0 overflow | T0 | 000BH |
| External interrupt 1/Seconds interrupt | X1/SEC | 0013H |
| Timer 1 overflow | T1 | 001BH |
| SIO0 (UART) | S0 | 0023H |
| SIO1 ($I^2C$) | S1 | 002BH |
| Timer 2 capture 0 | CT0 | 0033H |
| Timer 2 capture 1 | CT1 | 003BH |
| Timer 2 capture 2 | CT2 | 0043H |
| Timer 2 capture 3 | CT3 | 004BH |
| ADC completion | ADC | 0053H |
| Timer 2 compare 0 | CM0 | 005BH |
| Timer 2 compare 1 | CM1 | 0063H |
| Timer 2 compare 2 | CM2 | 006BH |
| Timer 2 overflow | T2 | 0073H |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PCON (87H) | SMOD | ARD | RFI | WLE | GF1 | GF0 | PD | IDL |

**Figure 39. Power control register (PCON).**

**Table 39. Description of PCON bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| SMOD | PCON.7 | Double Baud rate bit. When set to logic 1 the baud rate is doubled when the serial port SIO0 is being used in modes 1, 2, or 3. |
| ARD | PCON.6 | AUX-RAM disable bit. When set to a 1 the internal 768 bytes AUX-RAM is disabled, so that all MOVX-Instructions access the external data memory – as it is with the standard PCB80C51. |
| RFI | PCON.5 | Reduced radio frequency interference bit. When set to a 1 the toggling of ALE pin is prohibited. This bit is cleared on RESET (see also sections Features (EMC) and Pinning). |
| WLE | PCON.4 | Watchdog load enable. This flag must be set by software prior to loading timer T3 (watchdog timer). It is cleared when timer T3 is loaded. |
| GF1 | PCON.3 | General-purpose flag bit |
| GF0 | PCON.2 | General-purpose flag bit |
| PD | PCON.1 | Power-down bit. Setting this bit activates the power-down mode. It can only be set if input $\overline{EW}$ is high. |
| IDL | PCON.0 | Idle Mode bit. Setting this bit activates the Idle Mode. |

## 6.11 Power Reduction Modes

Two software-selectable modes of reduced power consumption are implemented. These are the Idle Mode and the Power-down Mode.

Idle Mode operation permits the interrupt, serial ports and timer blocks T0, T1 and T3 to function while the CPU is halted. The following functions are switched off when the microcontroller enters the Idle Mode:

- CPU (halted)
- Timer 2 (stopped and reset)
- PWM0, PWM1 (reset, output = HIGH)
- ADC (aborted if conversion in progress)

The following functions remain active during Idle Mode. These functions may generate an interrupt or reset and thus terminate the Idle Mode:

- Timer 0, Timer 1, Timer 3 (Watchdog timer)
- UART
- $I^2C$
- External interrupt
- Seconds Timer

In Power-down Mode the system clock is halted. If the PLL oscillator is selected (SELXTAL1 = 0) and the RUN32 bit is set, the 32 kHz oscillator keeps running, otherwise it is stopped. If the HF-oscillator (SELXTAL1 = 1) is selected, it is stopped after setting the bit PD in the PCON register.

**Table 40. External Pin Status During Idle and Power-Down Modes**

| MODE | MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 | PORT 4 | SCL/SDA | $\overline{PWM0/PWM1}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | data | data | data | data | data | operative (1) | HIGH |
| Idle | External | 1 | 1 | high-Z | data | address | data | data | operative (1) | HIGH |
| Power-down | Internal | 0 | 0 | data | data | data | data | data | high-Z | HIGH |
| Power-down | External | 0 | 0 | high-Z | data | data | data | data | high-Z | HIGH |

**NOTE:**

1. In Idle Mode SCL and SDA can be active as outputs only if SIO1 is enabled; if SIO1 is disabled (S1CON.6/ENS1 = 0) these pins are in a high-impedance state.

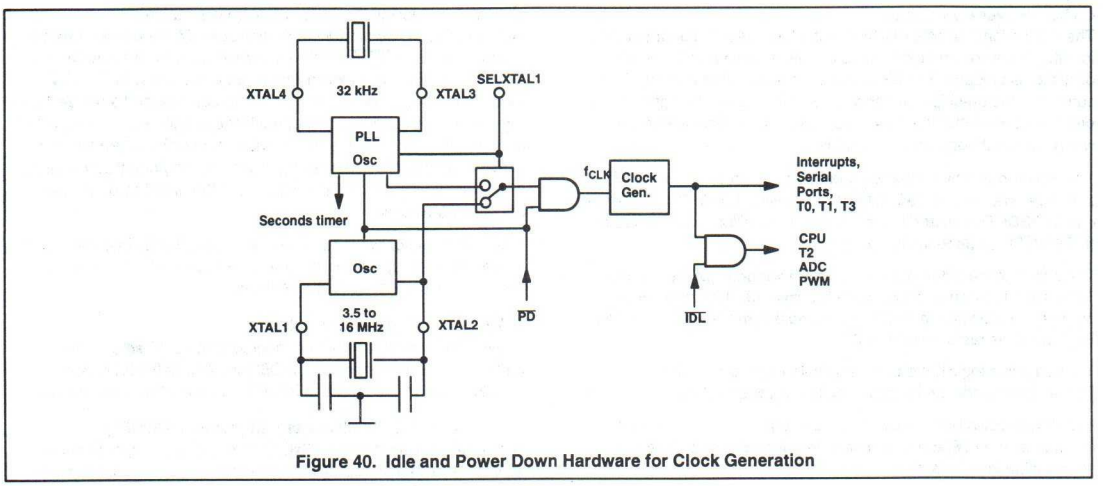



**Figure 40. Idle and Power Down Hardware for Clock Generation**

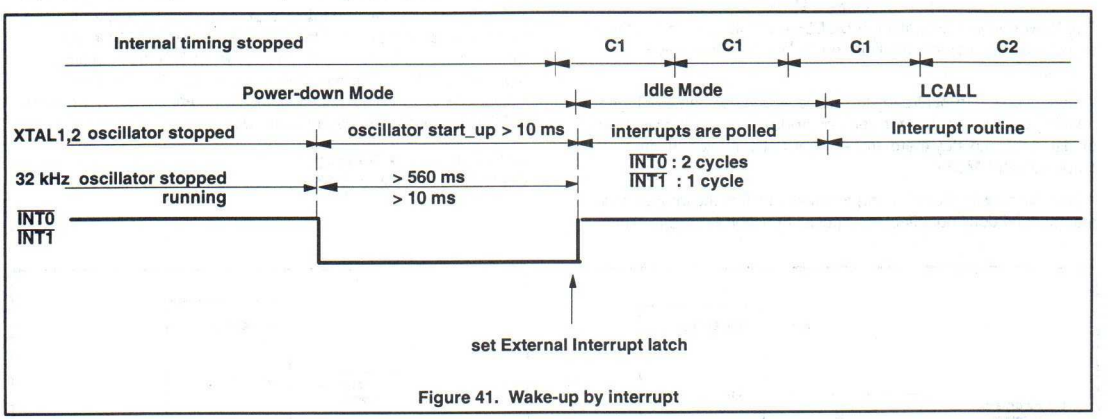



**Figure 41. Wake-up by interrupt**

### 6.11.1 Power Control Register

The modes Idle and Power-down are activated by software via the Special Function Register PCON. Its hardware address is 87H. PCON is not bit addressable. The reset value of PCON is (00000000).

### 6.11.2 Idle Mode

The instruction that sets PCON.0 is the last instruction executed in the normal operating mode before Idle Mode is activated. Once in the Idle Mode, the CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator, RAM and all other registers maintain their data during Idle Mode. The status of external pins during Idle Mode is shown in Table 40.

There are three ways to terminate the Idle Mode:

Activation of any enabled interrupt X0, T0, X1, SEC, T1, S0 or S1 will cause PCON.0 to be cleared by hardware terminating Idle Mode but only, if there is no interrupt in service with the same or higher priority. The interrupt is serviced, and following return from interrupt instruction RETI, the next instruction to be executed will be the one which follows the instruction that wrote a logic 1 to PCON.0.

The flag bits GF0 and GF1 may be used to determine whether the interrupt was received during normal execution or during Idle Mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When Idle Mode is terminated by an interrupt, the service routine can examine the status of the flag bits.

The second way of terminating the Idle Mode is with an external hardware reset. Since the oscillator is still running, the hardware reset is required to be active for two machine cycles (24 HF oscillator periods) to complete the reset operation if the HF oscillator is selected.

When the PLL oscillator is selected a hardware reset of $\geq$ 1 µsec (but no longer than 10 ms) is required and the microcontroller will typically restart within 63 msec after the reset has finished.

The third way of terminating the Idle Mode is by internal watchdog reset. The microcontroller restarts after 3 machine cycles in all cases.

#### 6.11.3 Power-down Mode

The instruction that sets PCON.1 is the last executed prior to going into the Power-down Mode. Once in Power-down Mode, the HF oscillator is stopped. The 32 kHz oscillator may stay running. The content of the on-chip RAM and the Special Function Registers are preserved. Note that the Power-down Mode can not be entered when the watchdog has been enabled.

The Power-down Mode can be terminated by an external RESET in the same way as in the 80C51 (RAM is saved, but SFRs are cleared due to RESET) or in addition by any one of the external interrupts ($\overline{INT0}$, $\overline{INT1}$) or Seconds interrupt.

The status of the external pins during Power-down Mode is shown in Table 40. If the Power-down Mode is activated while in external program memory, the port data that is held in the Special Function Register P2 is restored to Port 2.

If the data is a logic1, the port pin is held HIGH during the Power-down Mode by the strong pull-up transistor P1 (see Figure 9).

The Power-down Mode should not be entered within an interrupt routine because Wake-up with an external or 'Seconds' interrupt is not possible in that case.

#### 6.11.4 Wake-up from Power-down Mode

The Power-down Mode of the P8xCE558 can also be terminated by any one of the three enabled interrupts, $\overline{INT0}$, $\overline{INT1}$ or Seconds interrupt.

If there is an interrupt already in service, which has same or higher priority as the Wake-up interrupt, Power-down Mode will switch over to Idle Mode and stay there until an interrupt of higher priority terminates Idle Mode.

A termination with these interrupts does not affect the internal data memory and does not affect the Special Function Registers. This gives the possibility to exit Power-down without changing the port output levels. To terminate the Power-down Mode with an external interrupt, $\overline{INT0}$ or $\overline{INT1}$ must be switched to be level-sensitive and must be enabled. The external interrupt input signal $\overline{INT0}$ or $\overline{INT1}$ must be kept LOW till the oscillator has restarted and stabilized (see Figure 41). A Seconds interrupt will terminate the Power-down Mode if it is enabled and $\overline{INT1}$ is level sensitive. In order to prevent any interrupt priority problems during Wake-up, the priority of the desired Wake-up interrupt should be higher than the priorities of all other enabled interrupt sources.

The instruction following the one that put the device into the Power-down Mode will be the first one which will be executed after the interrupt routine has been serviced.

## 6.12 Oscillator Circuits

The input signal SELXTAL1 connected to logic "1" selects the XTAL1, 2 oscillator (standard 80C51) instead of the XTAL3, 4 oscillator, which is halted and XTAL3, 4 must not be connected.

#### 6.12.1 XTAL1, 2 Oscillator circuit (standard 80C51)

The oscillator circuit of the P8xCE558 is a single-stage inverting amplifier in a Pierce oscillator configuration. The circuitry between the XTAL1 and XTAL2 is basically an inverter biased to the transfer point. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuitry. Both are operated in parallel resonance. XTAL1 is the high gain amplifier input, and XTAL2 is the output (see Figure 42). To drive the P8xCE558 externally, XTAL1 is driven from an external source and XTAL2 left open-circuit (see Figure 43).

#### 6.12.2 XTAL3, 4 Circuitry

Please refer to chapter 6.13.1

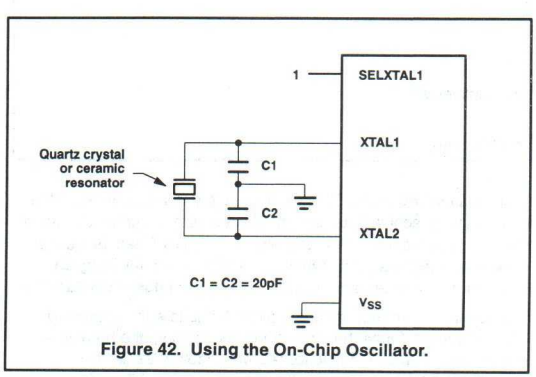



Figure 42. Using the On-Chip Oscillator.

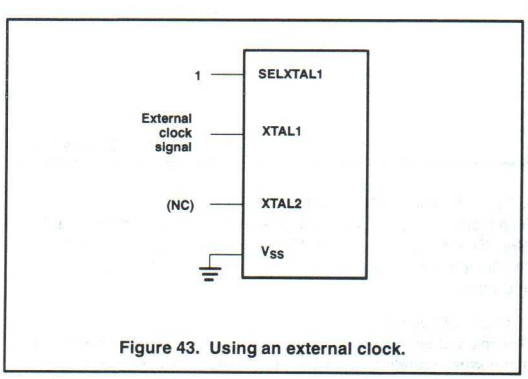



Figure 43. Using an external clock.

## 6.13 32kHz PLL Oscillator with Seconds Timer

### 6.13.1 XTAL3,4 Oscillator Circuitry

The input signal SELXTAL1 connected to logic "0" selects the 32kHz oscillator together with the PLL instead of the XTAL1,2 oscillator, which is halted. XTAL2 is floating in that case.

The 32kHz oscillator consists of an inverter, which forms a Pierce oscillator with the on-chip components C1,C2,Rf and an external crystal of 32768 Hz.

During the following situations, the inverter is switched to tristate and XTAL3 is pulled to Vss :

- during Power-down Mode, when the PLL control register bit RUN32 (PLLCON.7) was set to '0';
- during Reset (RSTIN = HIGH) ;
- when the XTAL1,2 oscillator is selected (SELXTAL1 = HIGH).

### 6.13.2 PLL CCO

A current controlled oscillator (CCO) generates a clock frequency $f_{CCO}$ of approx. 32 , 38 , 44 or 50 MHz , controlled by the PLL, with the 32kHz oscillator as the reference clock. The system clock frequency $f_{CLK}$ can be varied under software control by changing the contents of the PLL control register (PLLCON):

$f_{CCO}$ can be changed via the PLLCON bits FSEL(1:0) (see Table 41). The maximum locking time is 10 ms[1].

During the stabilization phase, no time critical routines should be executed.

The system clock frequency $f_{CLK}$ is derived from $f_{CCO}$ under control of the PLLCON bits FSEL(4:0) (see Table 41).

If only FSEL(4:2) is changed but not FSEL(1:0), then it takes about 1us until the new frequency is available.

Changing the system clock frequency has to be done in two steps.

**From HIGH to LOW frequencies:**
First change (FSEL(4:2), then FSEL (1:0).

**From LOW to HIGH frequencies:**
First change only FSEL (1:0) and after a stabilization phase of 10 ms change FSEL (4:2).

### 6.13.3 PLL Control Register – PLLCON

PLLCON is a special function register, which can be read and written by software. It contains the control bits:

- to select one of several system clock frequencies (see Table 41);
- the seconds interrupt flag: SECINT
- to enable the seconds interrupt flag: ENSECI
- the RUN32 bit, which defines if during Power-down Mode the 32kHz oscillator is halted or stays running.

PLLCON is initialized to 0DH upon Reset (RSTIN = '1') or Watchdog Timer Overflow. PLLCON = 0DH corresponds to a system clock frequency of 11.01 MHz.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **PLLCON (F9H)** | RUN32 | ENSECI | SECINT | FSEL.4 | FSEL.3 | FSEL.2 | FSEL.1 | FSEL.0 |

**Figure 44. PLL control register (PLLCON).**

**Table 41. PLLCON**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| RUN32 | PLLCON.7 | RUN32 = 0: The 32 kHz oscillator halts during Power-down.<br>RUN32 = 1: The 32 kHz oscillator stays running during Power-down. |
| ENSECI | PLLCON.6 | Enable the seconds interrupt. (enabling INT1 is also required) |
| SECINT | PLLCON.5 | Seconds interrupt requested by an overflow of the seconds timer (which occurs every second) or via writing a '1' to this bit. SECINT can only be cleared by writing a '0' to this bit . |
| FSEL.4 to FSEL.0 | PLLCON.4 to PLLCON.0 | System clock frequency in MHz |

| | | FSEL[4:2] | | |
|---|---|---|---|---|
| | | 100 | 011 | 010 |
| FSEL[1:0] | 11 | 3.93 | 7.86 | 15.73 |
| | 10 | 4.72 | 9.44 | |
| | 01 | 5.51 | 11.01 | |
| | 00 | 6.29 | 12.58 | |

Other combinations, than mentioned above, are reserved and may not be selected. This allows to generate the standard baudrates 19200, 9600, 4800, 2400 and 1200 Baud, when using the UART and Timer1.

**NOTE:**
1. This parameter is characterized.

### 6.13.4 Seconds Timer

This counter provides an overflow signal every second, when the 32kHz oscillator is running.

The overflow output sets the interrupt flag SECINT. This interrupt can be disabled/enabled by ENSECI. If SECINT is enabled, it is logically ORed with INT1 (external interrupt 1).

Seconds interrupt and INT1 therefor share the same priority and vector. The software has to check both flags SECINT (PLLCON.5) and IE1 (TCON.3), to distinguish between the two interrupt sources. SECINT can only be cleared via writing a '0' to this bit .

The external interrupts INT0 , INT1 or the seconds interrupt can Wake-up the PLL oscillator and the microcontroller as described in chapter "Wake-up from Power-down Mode".

For a Wake-up via INT1 or seconds interrupt, IE1 must be enabled and level-sensitive.

A further function of the seconds timer is to control the start-up timing of the microcontroller after Reset or after Wake-up from Power-down. It controls the stretching of the reset pulse to the microcontroller and controls releasing the system clock to the microcontroller.

A RSTIN signal of 1us at minimum will reset the microcontroller.

In case of Reset or Wake-up with halted 32kHz oscillator: From RSTIN falling edge or Wake-up interrupt it takes 560ms at maximum for the start-up of the 32kHz oscillator itself and the stabilization of the PLL's.

In case of Wake-up with running 32kHz oscillator: From Wake-up interrupt it takes about 1ms for the stabilization of the PLL's.

After this start-up time, the microcontroller is supplied with the system clock and – in case of a reset – the internally stretched reset signal overlaps about 45us, to guarantee a proper initialization of the microcontroller.

For further information refer to section ***6.11 Power reduction modes***.

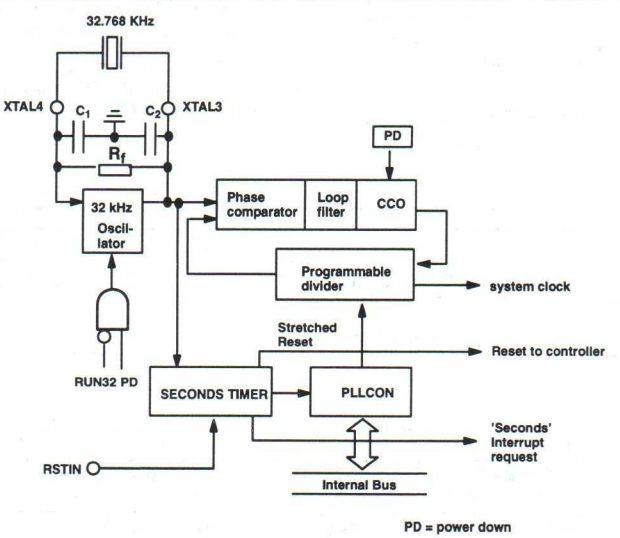



**Figure 45. Block diagram PLL**

## 6.14 Reset Circuitry

The reset input pin RSTIN is connected to a Schmitt trigger for noise reduction (see Figure 46). Is the HF-oscillator selected a Reset is accomplished by holding the RSTIN pin HIGH for at least 2 machine cycles (24 system clock periods). Is the PLL-oscillator selected the RSTIN-pulse must have a width of 1 µs at least, independent of the 32 kHz-oscillator is running or not (see PLL description). The CPU responds by executing an internal reset. The RSTOUT pin represents the signal resetting the CPU and can be used to reset peripheral devices.

The RSTOUT level also could be high due to a Watchdog timer overflow.

The length of the output pulse from T3 is 3 machine cycles. A pulse of such short duration is necessary in order to recover from a processor or system fault as fast as possible.

During Reset, ALE and $\overline{PSEN}$ output a HIGH level. In order to perform a correct reset, this level must not be affected by external elements.

A Reset leaves the internal registers as shown in Table 5.

The internal RAM is not affected by Reset. At power-on, the RAM content is indeterminate.

## 6.15 Power-on Reset

An automatic Reset can be obtained by switching on $V_{DD}$, if the RSTIN pin is connected to $V_{DD}$ via a capacitor, as shown in Figure 47.

Is the HF oscillator selected the $V_{DD}$ rise time must not exceed 10 ms and the capacitor should be at least 2.2 µF. The decrease of the RSTIN pin voltage depends on the capacitor and the internal resistor $R_{RST}$. That voltage must remain above the lower threshold for at minimum the HF-oscillator start-up time plus 2 machine cycles. Is the PLL-oscillator selected a 0.1 µF capacitor is sufficient to obtain an automatic reset.

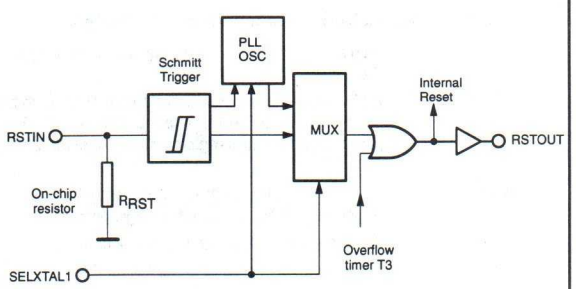



**Figure 46. On-chip Reset Configuration**

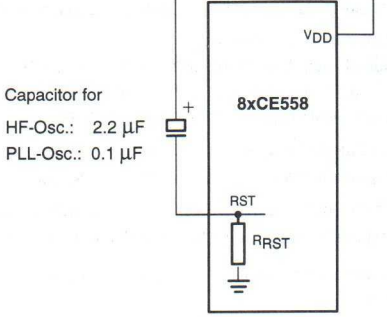



**Figure 47. Power-on Reset**

## 7. INSTRUCTION SET

The P8xCE558 uses the powerful instruction set of the PCB80C51. It consists of 49 single-byte, 45 two-byte and 17 three-byte instructions. Using a 16 MHz quartz, 64 of the instructions are executed in 0.75 µs, 45 in 1,5 µs and the multiply, divide instructions in 3 µs.

A summary of the instruction set is given in Table 43.

The P8xCE558 has additional Special Function Registers to control the on-chip peripherals.

### 7.1 Addressing Modes

Most instructions have a "destination, source" field that specifies the data type, addressing modes and operands involved. For all these instructions, except for MOVs, the destination operand is also the source operand (e.g., ADD A,R7).

There are five kinds of addressing modes:

- Register Addressing
  - R0 – R7 (4 banks)
  - A,B,C (bit), AB (2 bytes), DPTR (double byte)
- Direct Addressing
  - lower 128 bytes of internal Main RAM (including the 4 R0–R7 register banks)
  - Special Function Registers
  - 128 bits in a subset of the internal Main RAM
  - 128 bits in a subset of the Special Function Registers
- Register-Indirect Addressing
  - internal Main RAM (@R0, @R1, @SP [PUSH/POP])
  - internal Auxiliary RAM (@R0, @R1, @DPTR)
  - external Data Memory (@R0, @R1, @DPTR)
- Immediate Addressing
  - Program Memory (in-code 8 bit or 16 bit constant)
- Base-Register-plus Index-Register-Indirect Addressing
  - Program Memory look-up table (@DPTR+A, @PC+A)

The first three addressing modes are usable for destination operands.

#### 7.1.1 80C51 Family Instruction Set

**Table 42. Instruction that affect Flag settings[1]**

| INSTRUCTION | FLAG | | |
|---|---|---|---|
| | C | OV | AC |
| ADD | X | X | X |
| ADDC | X | X | X |
| SUBB | X | X | X |
| MUL | 0 | X | |
| DIV | 0 | X | |
| DA | X | X | |
| RRC | X | | |
| RLC | X | | |
| SETB C | 1 | | |
| CLR C | 0 | | |
| CPL C | X | | |
| ANL C, bit | X | | |
| ANL C,/bit | X | | |
| ANL C, bit | X | | |
| ORL C, bit | X | | |
| MOV C, bit | X | | |
| CJNE | X | | |

**NOTES:**

1. Note that operations on SFR byte address 208 or bit addresses 209-215 (i.e., the PSW or bits in the PSW) will also affect flag settings.

**Notes on instruction set and addressing modes:**

| | |
|---|---|
| Rn | Register R7-R0 of the currently selected Register Bank. |
| direct | 8-bit internal data location's address. This could be an Internal Data RAM location (0-127) or a SFR [i.e., I/O port, control register, status register, etc. (128-255)]. |
| @Ri | 8-bit RAM location addressed indirectly through register R1 or R0 of the actual register bank. |
| #data | 8-bit constant included in the instruction. |
| #data 16 | 16-bit constant included in the instruction |
| addr 16 | 16-bit destination address. Used by LCALL and LJMP. A branch can be anywhere within the 64 Kbytes Program Memory address space. |
| addr 11 | 11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 Kbytes page of program memory as the first byte of the following instruction. |
| rel | Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is –128 to +127 bytes relative to first byte of the following instruction. |
| bit | Direct Addressed bit in Internal Data RAM or Special Function Register. |

**Hexadecimal opcode cross-reference to Table 43:**

| | | |
|---|---|---|
| * | : | 8, 9, A, B, C, D, E. F. |
| ** | : | 11, 31, 51, 71, 91, B1, D1, F1. |
| *** | : | 01, 21, 41, 61, 81, A1, C1, E1. |

**Table 43. 80C51 Instruction Set Summary**

| MNEMONIC | | DESCRIPTION | BYTE | CYCLES | OPCODE (HEX.) |
|---|---|---|---|---|---|
| **ARITHMETIC OPERATIONS** | | | | | |
| ADD | A,Rn | Add register to Accumulator | 1 | 1 | 2* |
| ADD | A,direct | Add direct byte to Accumulator | 2 | 1 | 25 |
| ADD | A,@Ri | Add indirect RAM to Accumulator | 1 | 1 | 26, 27 |
| ADD | A,#data | Add immediate data to Accumulator | 2 | 1 | 24 |
| ADDC | A,Rn | Add register to Accumulator with carry | 1 | 1 | 3* |
| ADDC | A,direct | Add direct byte to Accumulator with carry | 2 | 1 | 35 |
| ADDC | A,@Ri | Add indirect RAM to Accumulator with carry | 1 | 1 | 36, 37 |
| ADDC | A,#data | Add immediate data to ACC with carry | 2 | 1 | 34 |
| SUBB | A,Rn | Subtract Register from ACC with borrow | 1 | 1 | 9* |
| SUBB | A,direct | Subtract direct byte from ACC with borrow | 2 | 1 | 95 |
| SUBB | A,@Ri | Subtract indirect RAM from ACC with borrow | 1 | 1 | 96, 97 |
| SUBB | A,#data | Subtract immediate data from ACC with borrow | 2 | 1 | 94 |
| INC | A | Increment Accumulator | 1 | 1 | 04 |
| INC | Rn | Increment register | 1 | 1 | 0* |
| INC | direct | Increment direct byte | 2 | 1 | 05 |
| INC | @Ri | Increment indirect RAM | 1 | 1 | 06, 07 |
| DEC | A | Decrement Accumulator | 1 | 1 | 14 |
| DEC | Rn | Decrement Register | 1 | 1 | 1* |
| DEC | direct | Decrement direct byte | 2 | 1 | 15 |
| DEC | @Ri | Decrement indirect RAM | 1 | 1 | 16, 17 |
| INC | DPTR | Increment Data Pointer | 1 | 2 | A3 |
| MUL | AB | Multiply A and B | 1 | 4 | A4 |
| DIV | AB | Divide A by B | 1 | 4 | 84 |
| DA | A | Decimal Adjust Accumulator | 1 | 1 | D4 |
| **LOGICAL OPERATIONS** | | | | | |
| ANL | A,Rn | AND Register to Accumulator | 1 | 1 | 5* |
| ANL | A,direct | AND direct byte to Accumulator | 2 | 1 | 55 |
| ANL | A,@Ri | AND indirect RAM to Accumulator | 1 | 1 | 56, 57 |
| ANL | A,#data | AND immediate data to Accumulator | 2 | 1 | 54 |
| ANL | direct,A | AND Accumulator to direct byte | 2 | 1 | 52 |
| ANL | direct,#data | AND immediate data to direct byte | 3 | 2 | 53 |
| ORL | A,Rn | OR register to Accumulator | 1 | 1 | 4* |
| ORL | A,direct | OR direct byte to Accumulator | 2 | 1 | 45 |
| ORL | A,@Ri | OR indirect RAM to Accumulator | 1 | 1 | 46, 47 |
| ORL | A,#data | OR immediate data to Accumulator | 2 | 1 | 44 |
| ORL | direct,A | OR Accumulator to direct byte | 2 | 1 | 42 |
| ORL | direct,#data | OR immediate data to direct byte | 3 | 2 | 43 |
| XRL | A,Rn | Exclusive-OR register to Accumulator | 1 | 1 | 6* |
| XRL | A,direct | Exclusive-OR direct byte to Accumulator | 2 | 1 | 65 |
| XRL | A,@Ri | Exclusive-OR indirect RAM to Accumulator | 1 | 1 | 66, 67 |

**Table 43. 80C51 Instruction Set Summary** (Continued)

| MNEMONIC | | DESCRIPTION | BYTE | CYCLES | OPCODE (HEX.) |
|---|---|---|---|---|---|
| **LOGICAL OPERATIONS** (Continued) | | | | | |
| XRL | A,#data | Exclusive-OR immediate data to Accumulator | 2 | 1 | 64 |
| XRL | direct,A | Exclusive-OR Accumulator to direct byte | 2 | 1 | 62 |
| XRL | direct,#data | Exclusive-OR immediate data to direct byte | 3 | 2 | 63 |
| CLR | A | Clear Accumulator | 1 | 1 | E4 |
| CPL | A | Complement Accumulator | 1 | 1 | F4 |
| RL | A | Rotate Accumulator left | 1 | 1 | 23 |
| RLC | A | Rotate Accumulator left through the carry | 1 | 1 | 33 |
| RR | A | Rotate Accumulator right | 1 | 1 | 03 |
| RRC | A | Rotate Accumulator right through the carry | 1 | 1 | 13 |
| SWAP | A | Swap nibbles within the Accumulator | 1 | 1 | C4 |
| **DATA TRANSFER** | | | | | |
| MOV | A,Rn | Move register to Accumulator | 1 | 1 | E* |
| MOV | A,direct | Move direct byte to Accumulator | 2 | 1 | E5 |
| MOV | A,@Ri | Move indirect RAM to Accumulator | 1 | 1 | E6, E7 |
| MOV | A,#data | Move immediate data to Accumulator | 2 | 1 | 74 |
| MOV | Rn,A | Move Accumulator to register | 1 | 1 | F* |
| MOV | Rn,direct | Move direct byte to register | 2 | 2 | A* |
| MOV | RN,#data | Move immediate data to register | 2 | 1 | 7* |
| MOV | direct,A | Move Accumulator to direct byte | 2 | 1 | F5 |
| MOV | direct,Rn | Move register to direct byte | 2 | 2 | 8* |
| MOV | direct,direct | Move direct byte to direct | 3 | 2 | 85 |
| MOV | direct,@Ri | Move indirect RAM to direct byte | 2 | 2 | 86, 87 |
| MOV | direct,#data | Move immediate data to direct byte | 3 | 2 | 75 |
| MOV | @Ri,A | Move Accumulator to indirect RAM | 1 | 1 | F6, F7 |
| MOV | @Ri,direct | Move direct byte to indirect RAM | 2 | 2 | A6, A7 |
| MOV | @Ri,#data | Move immediate data to indirect RAM | 2 | 1 | 76, 77 |
| MOV | DPTR,#data16 | Load Data Pointer with a 16-bit constant | 3 | 2 | 90 |
| MOVC | A,@A+DPTR | Move Code byte relative to DPTR to ACC | 1 | 2 | 93 |
| MOVC | A,@A+PC | Move Code byte relative to PC to ACC | 1 | 2 | 83 |
| MOVX | A,@Ri | Move AUX-RAM (8-bit addr) to ACC | 1 | 2 | E3, E3 |
| MOVX | A,@DPTR | Move AUX-RAM (16-bit addr) to $A_{CC}$ | 1 | 2 | E0 |
| MOVX | @Ri,A | Move ACC to AUX-RAM (8-bit addr) | 1 | 2 | F2, F3 |
| MOVX | @DPTR,A | Move ACC to AUX-RAM (16-bit addr) | 1 | 2 | F0 |
| PUSH | direct | Push direct byte onto stack | 2 | 2 | C0 |
| POP | direct | Pop direct byte from stack | 2 | 2 | D0 |
| XCH | A,Rn | Exchange register with Accumulator | 1 | 1 | C* |
| XCH | A,direct | Exchange direct byte with Accumulator | 2 | 1 | C5 |
| XCH | A,@Ri | Exchange indirect RAM with Accumulator | 1 | 1 | C6, C7 |
| XCHD | A,@Ri | Exchange low-order digit indirect RAM with ACC | 1 | 1 | D6, D7 |

**Table 43. 80C51 Instruction Set Summary** (Continued)

| MNEMONIC | | DESCRIPTION | BYTE / CYCLES | | OPCODE (HEX.) |
|---|---|---|---|---|---|
| **BOOLEAN VARIABLE MANIPULATION** | | | | | |
| CLR | C | Clear carry | 1 | 1 | C3 |
| CLR | bit | Clear direct bit | 2 | 1 | C2 |
| SETB | C | Set carry | 1 | 1 | D3 |
| SETB | bit | Set direct bit | 2 | 1 | D2 |
| CPL | C | Complement carry | 1 | 1 | B3 |
| CPL | bit | Complement direct bit | 2 | 1 | B2 |
| ANL | C,bit | AND direct bit to carry | 2 | 2 | B2 |
| ANL | C,/bit | AND complement of direct bit to carry | 2 | 2 | B0 |
| ORL | C,bit | OR direct bit to carry | 2 | 2 | 72 |
| ORL | C,/bit | OR complement of direct bit to carry | 2 | 2 | A0 |
| MOV | C,bit | Move direct bit to carry | 2 | 1 | A2 |
| MOV | bit,C | Move carry to direct bit | 2 | 2 | 92 |
| JC | rel | Jump if carry is set | 2 | 2 | 40 |
| JNC | rel | Jump if carry not set | 2 | 2 | 50 |
| JB | rel | Jump if direct bit is set | 2 | 2 | 20 |
| JNB | rel | Jump if direct bit is not set | 2 | 2 | 30 |
| JBC | bit,rel | Jump if direct bit is set and clear bit | 3 | 2 | 10 |
| **PROGRAM BRANCHING** | | | | | |
| ACALL | addr11 | Absolute subroutine call | 2 | 2 | **1addr |
| LCALL | addr16 | Long subroutine call | 3 | 2 | 12 |
| RET | | Return from subroutine | 1 | 2 | 22 |
| RETI | | Return from interrupt | 1 | 2 | 32 |
| AJMP | addr11 | Absolute jump | 2 | 2 | ***1addr |
| LJMP | addr16 | Long jump | 3 | 2 | 02 |
| SJMP | rel | Short jump (relative addr) | 2 | 2 | 80 |
| JMP | @A+DPTR | Jump indirect relative to the DPTR | 1 | 2 | 73 |
| JZ | rel | Jump if Accumulator is zero | 2 | 2 | 60 |
| JNZ | rel | Jump if Accumulator is not zero | 2 | 2 | 70 |
| CJNE | A,direct,rel | Compare direct byte to ACC and jump if not equal | 3 | 2 | B5 |
| CJNE | A,#data,rel | Compare immediate to ACC and jump if not equal | 3 | 2 | B4 |
| CJNE | RN,#data,rel | Compare immediate to register and jump if not equal | 3 | 2 | B* |
| CJNE | @Ri,#data,rel | Compare immediate to indirect and jump if not equal | 3 | 2 | B6, B7 |
| DJNZ | Rn,rel | Decrement register and jump if not zero | 2 | 2 | D* |
| DJNZ | direct,rel | Decrement direct byte and jump if not zero | 3 | 2 | D5 |
| NOP | | No operation | 1 | 1 | 00 |

**NOTE:**

1. All mnemonics copyrighted © Intel Corporation 1980

**Table 44. Instruction map P8xCE558**

**second hexadecimal character of opcode**

| first hexadecimal character of opcode | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | NOP | AJMP addr11 | LJMP addr16 | RR A | INC A | INC dir | INC @ Ri | | INC Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | JBC bit, rel | ACALL addr11 | LCALL addr16 | RRC A | DEC A | DEC dir | DEC @ Ri | | DEC Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 2 | JB bit, rel | AJMP addr11 | RET | RL A | ADD A, #data | ADD A, dir | ADD A, @ Ri | | ADD A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 3 | JNB bit, rel | ACALL addr11 | RETI | RLC A | ADDC A, #data | ADDC A, dir | ADDC A, @ Ri | | ADDC A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 4 | JC rel | AJMP addr11 | ORL dir, A | ORL dir, #data | ORL A, #data | ORL A, dir | ORL A, @ Ri | | ORL A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 5 | JNC rel | ACALL addr11 | ANL dir, A | ANL dir, #data | ANL A, #data | ANL A, dir | ANL A, @ Ri | | ANL A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 6 | JZ rel | AJMP addr11 | XRL dir, A | XRL dir, #data | XRL A, #data | XRL A, dir | XRL A, @ Ri | | XRL A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 7 | JNZ rel | ACALL addr11 | ORL C, bit | JMP @A+DPTR | MOV A, #data | MOV dir,#data | MOV @ Ri, #data | | MOV Rr, #data | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 8 | SJMP rel | AJMP addr11 | ANL C, bit | MOVC A, @A+PC | DIV AB | MOV dir, dir | MOV dir, @ Ri | | MOV dir, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 9 | MOV DPTR,#data16 | ACALL addr11 | MOV bit, C | MOVC A,@A+DPTR | SUBB A, #data | SUBB A, dir | SUBB A, @ Ri | | SUBB A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| A | ORL C,/bit | AJMP addr11 | MOV C, bit | INC DPTR | MUL AB | | MOV @ Ri, dir | | MOV Rr, dir | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| B | ANL C,/bit | ACALL addr11 | CPL bit | CPL C | CJNE A,#data,rel | CJNE A,dir, rel | CJNE @Ri,#data,rel | | CJNE Rr, #data, rel | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| C | PUSH dir | AJMP addr11 | CLR bit | CLR C | SWAP A | XCH A, dir | XCH A, @ Ri | | XCH A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| D | POP dir | ACALL addr11 | SETB bit | SETB C | DA A | DNJZ dir, rel | XCHD A, @ Ri | | DJNZ Rr, rel | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| E | MOVX A, @DPTR | AJMP addr11 | MOVX A, @Ri | | CLR A | MOV A, dir *) | MOV A, @ Ri | | MOV A, Rr | | | | | | | |
| | | | 0 | 1 | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| F | MOVX @DPTR, A | ACALL addr11 | MOVX A, @Ri, A | | CPL A | MOV dir, A | MOV @ Ri, A | | MOV Rr, A | | | | | | | |
| | | | 0 | 1 | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |

***) MOV A, ACC is not a valid instruction**

# 8. FLASH EEPROM

## 8.1 General

- 32 Kbytes electrically erasable internal program memory with Block-and Page-Erase option ("Flash Memory").
- Internal fixed boot ROM.
- Up to 32 Kbytes external program memory in combination with the internal FEEPROM ($\overline{EA}$=1).
- Up to 64 Kbytes external program memory if the internal program memory is switched off ($\overline{EA}$=0).

The FEEPROM can be read and written byte-wise. Full Erase, Block Erase, and Page erase will erase 32 Kbytes, 256 bytes and 32 bytes respectively. In-circuit programming and out-of-circuit programming is possible. On-chip erase and write timing generation and on chip high voltage generation contribute to a user friendly interface.

## 8.2 Features

- Read:
  byte-wise
- Write:
  byte-wise within 2.5 ms.
  (previously erased by a page, block or full erase).
- Erase:
  Page Erase (32 bytes) within 5 ms.
  Block Erase (256 bytes) within 5 ms.
  Full Erase (32 Kbytes) within 5 ms.
  Erased bytes contain FFH.
- Endurance:
  100 erase and write cycles each byte at $T_{amb}$ = 22°C
- Retention:
  10 years
- Out-of-circuit programming:
  Parallel programming with 87C51 compatible hardware Interface to programmer.
- In-circuit programming:
  Serial programming via RS232 interface under boot ROM program control. Auto baud rate selection.
  Intel Hex Object file Format.
  The user program can call routines in the boot ROM for erase, write and verify of the FEEPROM.
- High programming voltage generation: on chip
- Zero point on-chip oscillator and timer to generate the write and erase time durations.
- Programmable security for the code in the FEEPROM to prevent software piracy. The Security Byte is located in the highest address (7FFFH) of the FEEPROM.
- Supply voltage monitoring circuit on-chip to prevent loss of information in the FEEPROM during power-on and power-off.

## 8.3 Memory Map

Figure 48 shows the memory map of the user program memory and the boot ROM. They are located in the same program address space. Two bits UBS1 and UBS0 of the FEEPROM control special function register FMCON select between the two memory blocks.

**User program memory selection**
If UBS1 and UBS0 are both 0, then the user program memory is mapped into the 64 K program memory space and the boot ROM cannot be selected. This is the situation after a reset when $\overline{PSEN}$ and ALE have not been pulled down during reset. Program execution starts at 0000H in the internal FEEPROM or in the external program memory dependent on the level of $\overline{EA}$ during reset.

**Boot ROM selection**
After a reset program execution starts in the boot ROM when during reset $\overline{PSEN}$ and $\overline{EA}$ are pulled down while ALE stay high. The boot ROM size is 1 Kbyte. Besides the serial in-circuit programming routine the boot ROM contains the routines for erase, write and verify of the FEEPROM, which can be called by the user program (LCALL to the address space between 63 K and 64 K).

**Switching between user program memory and boot ROM**
Switching between user program memory (internal or external) and boot ROM is possible if UBS1 and UBS0 are 0,1. Then in the program memory address space between 0 and 63k the user program memory is selected and in the memory space between 63 K and 64 K the boot ROM is selected.

To switch from user program memory to boot ROM first UBS0 must be set (UBS1 stay 0) and a jump or call instruction to a location >63 K must be executed.

At the moment of crossing the 63 K address border by a return instruction the switching from boot ROM to user memory (internal or external) is performed. After crossing the 63 K address border UBS1 and UBS0 are cleared and the total 64 K memory space is mapped as user program memory. By clearing UBS1 and UBS0, no special requirements to the user program are necessary to do that after a read or erase or write routine.

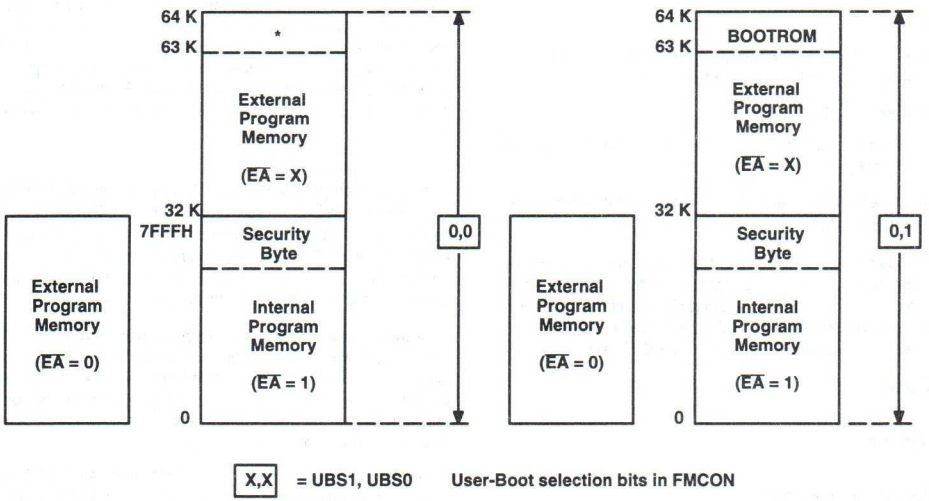
A small restriction for memory switching is that no memory switching is allowed from or to the address space between 63 K and 64 K of the user program memory because the UBS bits must stay 0 in this range. This restriction can be avoided if the memory switching is always done by a subroutine in the address range between 0 and 63 K.

**Description**

The user program code in the FEEPROM is executed as in the standard 80C51 microcontroller. Erase and write cycles in the FEEPROM are always performed under control of the boot program in the boot ROM in the address space between 63 K and 64 K. Address and data parameters are passed via DPTR and accumulator A respectively. During an erase or write cycle in the FEEPROM no other access or program execution in the FEEPROM is possible. All interrupts must be disabled when the user program calls a user routine in the boot ROM.

The boot routine for serial programming takes care of addressing, data transfer, verify, high voltage control, error message and return to the user program memory. It also contains the serial communication routine.

The FEEPROM control register FMCON is a special function register. It contains the control bits for verify, write, erase and boot ROM switching.



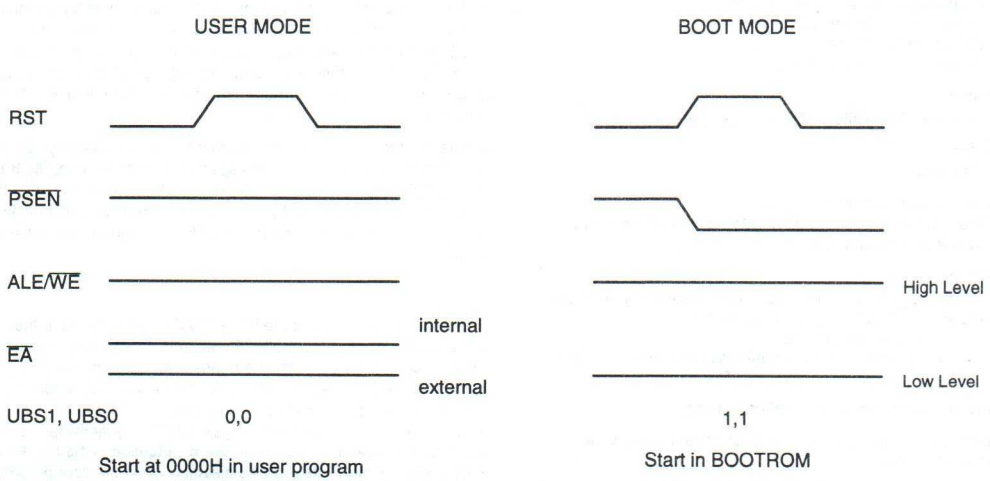




**Figure 48. Program memory map and operation modes**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **FMCON (FB)** | UBS1 | UBS0 | HV | – [1] | FCB3 | FCB2 | FCB1 | FCB0 |

**Figure 49. FEEPROM control register.**

**NOTE:**
1. Reserved for future use; a write operation must write "0" to the location.

## Table 45. Description of FMCON bits

| UBS1 | UBS0 | User - Boot selection bits |
|---|---|---|
| 0 | 0 | User memory mapped from 0 to 64 K. |
| 0 | 1 | User memory mapped from 0 to 63 K.<br>Boot ROM mapped from 63 K to 64 K. |
| 1 | 0 | User memory mapped from 0 to 63 K, but UBS1 bit cleared by hardware in this user address range.<br>Boot ROM mapped from 63 K to 64 K. User software should not write "1" UBS1. |
| 1 | 1 | Boot ROM mapped from 0 to 64 K. User software should not write "1" UBS1. |
| | | |
| **HV** | | **High voltage indication bit.** Read only. Is "1" as long as the high voltage for an erase or write operation is present. |

| FCB3 | FCB2 | FCB1 | FCB0 | Function Code Bits |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | Value after Reset. |
| 0 | 1 | 0 | 1 | Byte Write or byte read (verify) |
| 1 | 1 | 0 | 0 | Page Erase (32 bytes boundaries). |
| 0 | 0 | 1 | 1 | Block Erase (256 bytes boundaries). |
| 1 | 0 | 1 | 0 | Full Erase (32 Kbytes). |

The four FCB bits are write protected if the security feature is activated. Then only instructions in the internal program memory (FEEPROM) are able to write FCB (3–0), boot ROM and external program memory instructions cannot change FCB (3–0) except the full erase code can be loaded.

The duration of a write or erase operation is determined by the FEEPROM timer. This timer includes a zero point RC oscillator and cannot be controlled by software.

For calling a user routine in the boot ROM first all interrupts must be disabled and the DPTR and A have to be loaded with the desired values. After setting UBS0 = 1 and UBS1 = 0 and selecting the function via FCB-bits the respective user routine has to be called.

The table below lists the boot ROM user routines, which can be called by the user program. The content of FMCON, A and DPTR before the call is described by "(IN)" and the contents after the return is described by "(OUT)". The boot ROM user routines do not change other registers or Data memory.

| BOOT-ROM ROUTINE | CALL ADDRESS | FMCON (IN) | FMCON (OUT) | ACC (IN) | ACC (OUT) | DPTR (IN) | DPTR (OUT) |
|---|---|---|---|---|---|---|---|
| BYTE_READ | FFBAH | 45H | 15H | XXH | BYTE | BYTE ADDRESS | BYTE ADDRESS |
| BYTE_WRITE | FFADH | 45H | 15H | BYTE | BYTE (V) | BYTE ADDRESS | BYTE ADDRESS |
| PAGE_ERASE | FFAAH | 4CH | 1CH | XXH | 08H | PAGE ADDRESS 1) | PAGE ADDRESS 2) |
| BLOCK_ERASE | FFA5H | 43H | 13H | XXH | 02H | BLOCK ADDRESS 3) | BLOCK ADDRESS 4) |
| FULL_ERASE | FFA0H | 4AH | 1AH | XXH | 0AH | XXXXH | 0018H |

X = don't care or not defined
V = verified byte (read back)
1) = 5 LSB's of DPTR are don't care
2) = 5 LSB's of DPTR are "0"
3) = 8 LSB's of DPTR are don't care
4) = 8 LSB's of DPTR contain 08H.

Example of user software (internal or external) that calls the **Page Erase** routine in the boot ROM to erase a page in the FEEPROM (32 bytes) starting at address location 1260H.

```
CLR EA              ; Disable all interrupts
MOV DPTR, # 1260H   ; Load page-address
MOV FMCON, # 4CH    ; Load Page-Erase code
LCALL 0FFAAH        ; Call Page-Erase routine
                    ; in boot ROM (inherent delay
                      5 ms)
MOV FMCON, #00H     ; Clear FMCON for security
SETB EA             ; Enable interrupts again
```

Example of user software (internal or external) that calls the **Byte-Write** routine in the boot ROM to write the content of R5 into the FEEPROM address location 1263H.

```
CLR EA              ; Disable all interrupts
MOV DPTR, # 1263H   ; Load byte address
MOV A, R5           ; Load byte to be written
MOV FMCON, # 45H    ; Load byte-write code
LCALL 0FFADH        ; Call byte-write routine
                    ; in boot ROM (inherent
                      delay 2.5 ms)
MOV FMCON, #00H     ; Clear FMCON for security
SETB EA             ; Enable interrupts again
XRL A, R5           ; Compare the "read-back" byte
JNZ ERROR           ; Jump if verify error
```

## 8.4 Security

The security feature protects against software piracy and prevents that the content of the FEEPROM can be read undesirable. The Security Byte is located in the highest address location 7FFFH of the FEEPROM.

The Security Byte should be 50H to activate and 00H or FFH to deactivate the security feature. This security code is chosen in such a way that single bit failures will not deactivate the security feature.

If the security feature is deactivated, then there are no access restrictions to the FEEPROM.

If the security feature is activated, then the external program memory has no access to the FEEPROM with the MOVC instructions. Also bits FCB (3–0) of FMCON cannot be written by external program code or boot ROM code. This prevents in-circuit programming and verification. Only the Full Erase code can be written to FCB (0–3) of FMCON. Note that for the internal program code no restrictions exist if the security feature is activated. At the end of a full erase operation the security feature is deactivated. Also parallel programming and verify is inhibited if the security feature is activated, only a full erase is possible. Note that the security mode does not change immediately when the security code is written into the security byte 7FFFH, but after a reset or power-on. This allows the verification of the loaded code in the FEEPROM, including the Security Byte.

## 8.5 Parallel Programming

Unlike standard EPROM programming, no high programming supply voltage must be applied to the $\overline{EA}$ pin and only one programming pulse must be applied to the ALE/$\overline{WE}$ pin. The parallel programming mode is entered with the steady signals RST=1, PSEN=0, $\overline{EA}$=1 and SELXTAL1 = 1. The XTAL1,2 clock must have a frequency between 4 and 6MHz. The following table shows the logic levels for programming, erasing, verifying and read signature.

| MODE | ALE/$\overline{WE}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|
| Full erase | ⊓ | 1 | 1 | 0 | 1 |
| Program FEEPROM | ⊓ | 1 | 0 | 1 | 1 |
| Verify FEEPROM | 1 | 0 | 0 | 1 | 1 |
| Read signature | 1 | 0 | 0 | 0 | 0 |

ALE/$\overline{WE}$ Write Enable signal (program/erase), active low
P2.6, P2.7, P3.6, P3.7 control signals

**Data and address bits:**

| | | | |
|---|---|---|---|
| P0.0 – P0.7 | : | D0 – D7 | Program data input / verify or read data output |
| P1.0 – P1.7 | : | A0 – A7 | Input low order address bits. |
| P2.0 – P2.5, P3.4 | : | A8 – A14 | Input high order address bits. |

The P89CE558 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. These bytes are read by the same procedure as for a normal verification of locations 30H and 31H, except that P3.6 and P3.7 need to be pulled to LOW.

| ADDRESS | CONTENT | MEANING |
|---|---|---|
| 30H | 15H | Philips |
| 31H | B5H | P89CE558 |

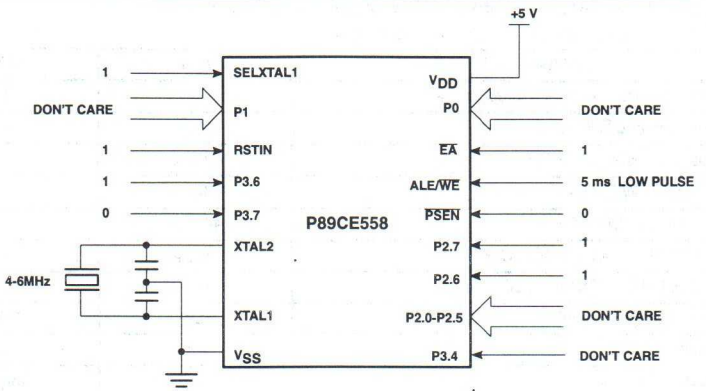



Figure 50. Erase Configuration

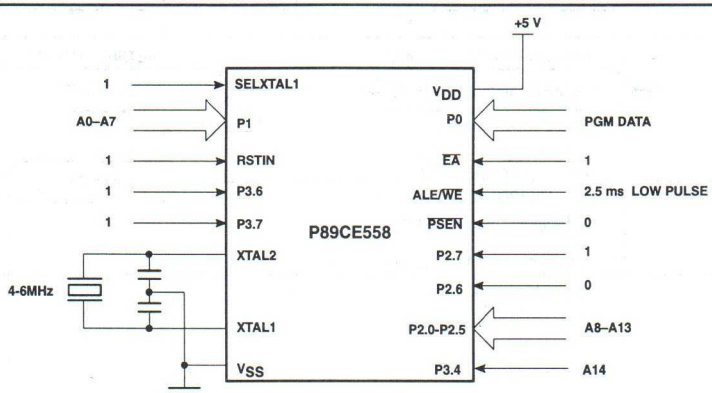



Figure 51. Programming Configuration

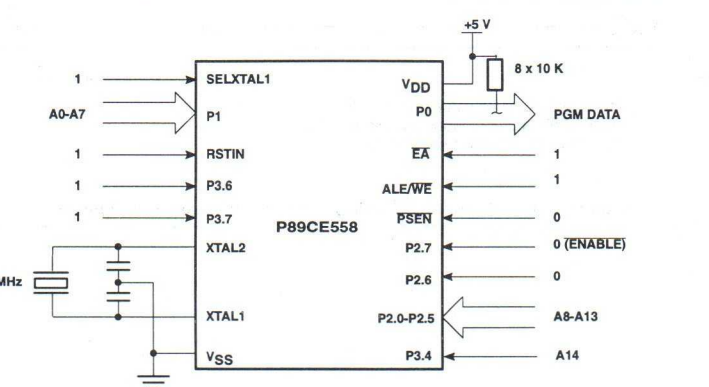
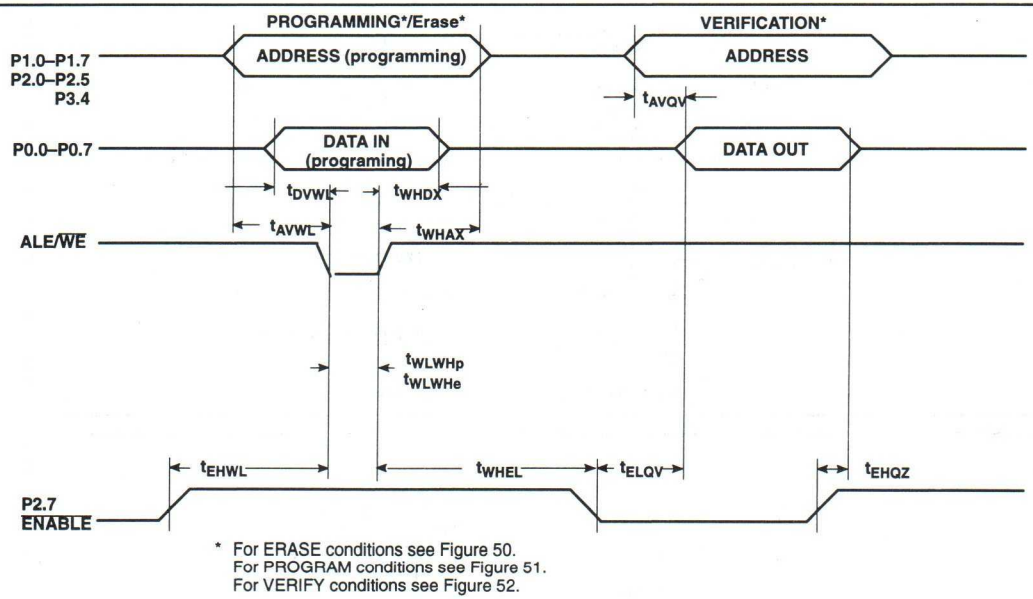



Figure 52. Program Verification

## FEEPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

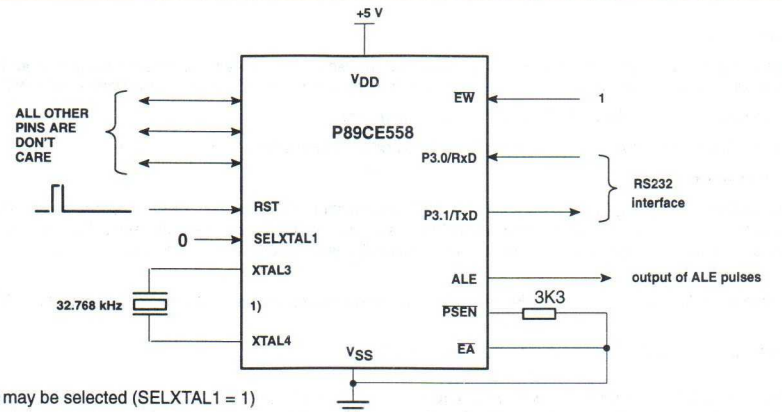
$T_{amb}$ = –40 °C to +85 °C, $V_{DD}$ = 5 V ± 10%, $V_{SS}$ = 0 V (see Figure 53)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $1/t_{CLK}$ | System clock frequency (standard oscillator) | 4 | 6 | MHz |
| $t_{AVWL}$ | Address setup to $\overline{WE}$ LOW | $48t_{CLK}$ | – | |
| $t_{WHAX}$ | Address hold after $\overline{WE}$ HIGH | $48t_{CLK}$ | – | |
| $t_{DVWL}$ | Data setup to $\overline{WE}$ LOW | $48t_{CLK}$ | – | |
| $t_{WHDX}$ | Data hold after $\overline{WE}$ HIGH | $48t_{CLK}$ | – | |
| $t_{EHWL}$ | P2.7 ($\overline{ENABLE}$) HIGH to $\overline{WE}$ LOW | $48t_{CLK}$ | – | |
| $t_{WHEL}$ | $\overline{WE}$ HIGH to P2.7 ($\overline{ENABLE}$) LOW | $48t_{CLK}$ | – | |
| $t_{WLWHp}$ | $\overline{WE}$ width (programming) | 2.25 | 2.75 | ms |
| $t_{WLWHe}$ | $\overline{WE}$ width (erase) | 4.5 | 5.5 | ms |
| $t_{AVQV}$ | Address to data valid | – | $48t_{CLK}$ | |
| $t_{ELQV}$ | P2.7 ($\overline{ENABLE}$) Low to data valid | – | $48t_{CLK}$ | |
| $t_{EHQZ}$ | Data float after P2.7 ($\overline{ENABLE}$) HIGH | 0 | $48t_{CLK}$ | |




* For ERASE conditions see Figure 50.
For PROGRAM conditions see Figure 51.
For VERIFY conditions see Figure 52.

**Figure 53. FEEPROM Programming/Erase and Verification Waveforms**




1) Alternative XTAL1, 2 may be selected (SELXTAL1 = 1)

**Figure 54. Serial programming (boot mode) Configuration**

## 8.6 Serial Programming of FEEPROM

Serial in-circuit programming (boot-mode) is entered if during and after RESET $\overline{PSEN}$ and $\overline{EA}$ are pulled down, $\overline{PSEN}$ via a resistor of 3.3 k Ohm to VSS. The two UBS bits are set to 1 by hardware and program execution starts at 0000H of the boot ROM. P3.0 (RXD) and P3.1 (TXD) form the serial RS232 interface. A baud rate of 4800 or 9600 Baud is possible, if the PLL oscillator is selected. The receive and transmit channel have the same baudrate. The format is: Startbit, 8 data bits (last bit always 0), no parity bit and at least one stopbit. The boot routine inputs the Intel Hex Object Format. The baud rate will be selected automatically after reception of the first character (:) of the object file. No other characters are allowed to preceed the first (:) character. Programming is only started if the first received record has the right type indication (TT). If the security feature is activated (contents of the security byte = 50H) then the programming starts with a Full Erase, otherwise only the addressed page(s) will be erased and the not altered bytes are rewritten. During the erase or write operation the next string of bytes can be received. Xon and Xoff handshake codes are used to control the serial transfer. At the end of the programming a message that indicates a successful or not successful programming, will be returned over the RS232 interface channel. If the programming was successful then the user program can be started up at 0000H in FEEPROM by a reset for user mode ($\overline{EA}$ = high, $\overline{PSEN}$ not affected). If the programming was not successful the boot program halts and a retry can be started by a reset for the boot mode.

## 8.7 Boot Routine

The boot routine transmits the next "one ASCII character" messages via the RS232 interface:

| | |
|---|---|
| " . " | After each record type TT = 00H indication in the HEX file. |
| " X " | Checksum error of a record in the HEX file detected. |
| " Y " | Wrong record type received |
| " Z " | Buffer overflow error (Check Xon/Xoff of terminal) |
| " R " | Verification error (of last written byte) |
| " V " | End record received and programming of FEEPROM was successful |

No messages are transmitted if the baud rate of the first character (:) can not be detected.

The boot routine can also be started by the internal or external user program (LJMP FC07H). FMCON must be loaded previously with 40H. Interrupt registers, stack pointer, Timer 0, UART, P3.0 and P3.1 must be in the reset state. $\overline{EA}$ and $\overline{PSEN}$ must not be affected. A reset is needed to restart the user program after programming.

The following baudrates will be detected automatically within the specified µC clock range in MHz.

| Baudrate | $f_{CLK}$ (min) | $f_{CLK}$ (max) |
|---|---|---|
| 1200 | 1 [1)] | 3.6 |
| 2400 | 2 [1)] | 7.3 |
| 4800 | 4 | 14.7 |
| 9600 | 7.9 | 29.5 [1)] |
| 19200 | 15.7 | 59 [1)] |

**NOTE:**
1. Value outside the specified clock range

Note that the boot routines can (re) program any number of bytes from 1 byte to 32 Kbytes, independent in which order or at which location, but if the security feature is activated, a full erase is performed and all not programmed bytes become FFH.

**Definitions:**

**:** – Record start character

**BC** – Byte Count. The hexadecimal number of data bytes in the record. This may theoretically be any number from 0 to 255, although many assemblers prefer to deal with 16 data bytes per record (as shown in the example below).

**AAAA** – Load address in hexadecimal of first data byte in this record.

**TT** – Record type. The record type is 00 for data records and 01 for the end record.

**HH** – One hexadecimal data byte.

**CC** – Record checksum. This is the 2's complement of the summation of all of the bytes in the record from the byte count through the last data byte. While the summation is calculated, it is always truncated to a one byte result. Thus, if all of the bytes in the record are summed, including the checksum itself, the result will always be 00 if the record is valid.

Construction of data records (using the notation defined above, each letter corresponds to one hexadecimal digit in ASCII representation) is as follows:

: BCAAAATTHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHCC

The last record in a file is the end record and contains no data. Usually the end record will appear as shown in the first example below. However, in some cases a 16 bit checksum of all of the data bytes in the entire file may be inserted in the address field of the end record. This checksum would correspond to one generated by an EPROM programmer during file load, and its inclusion does not violate the rules for this format. This is shown in the second example.

:00000001FF

:00B12C0122

Successive hex records need not appear in sequential address order . For instance, a record for address 0000H might appear after a record for address 7FE0H. All of the bytes in a single record, however, must be in sequence. Any characters that appear outside of a record (i.e. after a checksum, but before the next ":") will be ignored, if present.

An example of a valid hex file follows:

:10010000C2F0E53030E704F404D2F08531F030F786

:100110000763F0FF05F0B2F0A430F00A63F0FFF4DB

:0C0120002401500205F085F032F5332276

:00000001FF

## 9. ABSOLUTE MAXIMUM RATINGS

### ABSOLUTE MAXIMUM RATINGS [1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage on $V_{DD}$ to $V_{SS}$ and SCL, SDA to $V_{SS}$ | –0.5 to +6.5 | V |
| Input / output current on any I/O pin | 10 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.0 | W |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions are taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## 10. DC CHARACTERISTICS

### DC ELECTRICAL CHARACTERISTICS

$V_{DD}$ = 5V (± 10%), $V_{SS}$ = 0V, $T_{amb}$ = 0°C to +70°C (P8xCE558EBx). All voltages with respect to $V_{SS}$ unless otherwise specified.

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $V_{DD}$ | Supply voltage | | 4.5 | 5.5 | V |
| $I_{DD}$ | Supply current operating :<br>P89CE558<br>P83CE558 | See notes 1 and 2<br>$f_{CLK}$ = 16MHz<br>$V_{DD}$ = 5.5 V | | <br>40<br>40 | <br>mA<br>mA |
| $I_{ID}$ | Supply current Idle Mode :<br>P89CE558<br>P83CE558 | See notes 1 and 3<br>$f_{CLK}$ = 16MHz<br>$V_{DD}$ = 5.5 V | | <br>15<br>12 | <br>mA<br>mA |
| $I_{PD}$ | Supply current Power-down mode | See note 4<br>2 V < $V_{PD}$ < $V_{DDmax}$ | | 100 | µA |
| | Supply current Power-down mode:<br>32 kHz / PLL operation | See note 17<br>$V_{DD}$ = 5.5 V | | 100 | µA |
| **Inputs** | | | | | |
| $V_{IL}$ | Input LOW voltage, except $\overline{EA}$, SCL, SDA | | –0.5 | $0.2V_{DD}$–0.1 | V |
| $V_{IL1}$ | Input LOW voltage to $\overline{EA}$ | | –0.5 | $0.2V_{DD}$–0.3 | V |
| $V_{IL2}$ | Input LOW voltage to SCL, SDA [5] | | –0.5 | $0.3V_{DD}$ | V |
| $V_{IH}$ | Input HIGH voltage, except XTAL1, RSTIN, SCL, SDA, ADEXS | | $0.2V_{DD}$+0.9 | $V_{DD}$+0.5 | V |
| $V_{IH1}$ | Input HIGH voltage, XTAL1, RSTIN, ADEXS | | $0.7V_{DD}$ | $V_{DD}$+0.5 | V |
| $V_{IH2}$ | Input HIGH voltage, SCL, SDA [5] | | $0.7V_{DD}$ | 6.0 | V |
| $I_{IL}$ | Input current LOW level, Ports 1, 2, 3, 4 | $V_{IN}$ = 0.45 V | | –50 | µA |
| $I_{TL}$ | Transition current HIGH to LOW, Ports 1, 2, 3, 4 | See note 6 | | –650 | µA |
| $\pm I_{LI1}$ | Input leakage current, Port 0, $\overline{EA}$, ADEXS, $\overline{EW}$, SELXTAL1 | 0.45 V < $V_I$ < $V_{DD}$ | | 10 | µA |
| $\pm I_{LI2}$ | Input leakage current, SCL, SDA | 0 V < $V_I$ < 6 V<br>0 V < $V_{DD}$ < 5.5 V | | 10 | µA |
| $\pm I_{LI3}$ | Input leakage current, Port 5 | 0.45 V < $V_I$ < $V_{DD}$ | | 1 | µA |
| **Outputs** | | | | | |
| $V_{OL}$ | Output low voltage, Ports 1, 2, 3, 4 | $I_{OL}$ = 1.6mA[7] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, Port 0, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$, RSTOUT | $I_{OL}$ = 3.2mA[7] | | 0.45 | V |
| $V_{OL2}$ | Output low voltage, SCL, SDA | $I_{OL}$ = 3.0mA[7, 19] | | 0.4 | V |
| | | $I_{OL}$ = 6.0mA[7, 19] | | 0.6 | |
| $V_{OH}$ | Output high voltage, Ports 1, 2, 3, 4 | $V_{DD}$ = 5 V ± 10%<br>$-I_{OH}$ = 60µA<br>$-I_{OH}$ = 25µA<br>$-I_{OH}$ = 10µA | <br>2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | <br>V<br>V<br>V |
| $V_{OH1}$ | Output high voltage (Port 0 in external bus mode, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$, RSTOUT) [8] | $V_{DD}$ = 5 V ± 10%<br>$-I_{OH}$ = 800µA<br>$-I_{OH}$ = 300µA<br>$-I_{OH}$ = 80µA | <br>2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | <br>V<br>V<br>V |
| $V_{HYS}$ | Hysteresis of Schmitt Trigger inputs SCL, SDA (Fast-mode) | | $0.05V_{DD}$ [20] | | V |

**NOTES: See Page 3-631.**

## DC ELECTRICAL CHARACTERISTICS (Continued)

$V_{DD}$ = 5 V (± 10%), $V_{SS}$ = 0 V, $T_{amb}$ = –40°C to +85°C (P8xCE558EFx).
DC parameters not included here are the same as in the P8xCE558EBx, DC electrical characteristics
All voltages with respect to $V_{SS}$ unless otherwise specified.

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $R_{RST}$ | Internal reset pull-down resistor | | 50 | 150 | kΩ |
| $C_{IO}$ | Pin capacitance | Test freq = 1MHz, $T_{amb}$ = 25 °C | | 10 | pF |
| **Inputs** | | | | | |
| $V_{IL}$ | Input LOW voltage, except $\overline{EA}$, SCL, SDA | | –0.5 | $0.2V_{DD}$–0.15 | V |
| $V_{IL1}$ | Input LOW voltage to $\overline{EA}$ | | –0.5 | $0.2V_{DD}$–0.35 | V |
| $V_{IH}$ | Input HIGH voltage, except XTAL1, RSTIN, SCL, SDA, ADEXS | | $0.2V_{DD}$+1.0 | $V_{DD}$+0.5 | V |
| $V_{IH1}$ | Input HIGH voltage, XTAL1, RSTIN, ADEXS | | $0.7V_{DD}$+0.1 | $V_{DD}$+0.5 | V |
| $I_{IL}$ | Input current LOW level, Ports 1, 2, 3, 4 | $V_{IN}$ = 0.45 V | | –75 | µA |
| $I_{TL}$ | Transition current HIGH to LOW, Ports 1, 2, 3, 4 | See note 6 | | –750 | µA |

**NOTES: See Page 3-631.**

## DC ELECTRICAL CHARACTERISTICS ANALOG

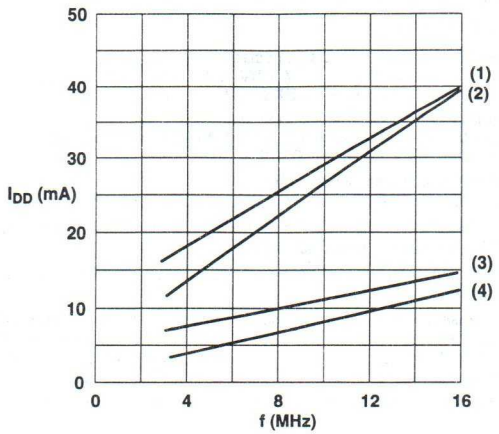
$AV_{DD}$ = 5 V (± 10%), $AV_{SS}$ = 0 V, Tamb = 0 °C to +70 °C (P8xCE558EBx).
$AV_{DD}$ = 5 V (± 10%), $AV_{SS}$ = 0 V, Tamb = –40 °C to +85 °C (P8xCE558EFx).
All voltages with respect to $V_{SS}$ unless otherwise specified.

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $AV_{DD}$ | Analog supply voltage | $AV_{DD} = V_{DD} \pm 0.2$ V | 4.5 | 5.5 | V |
| $AI_{DD}$ | Analog supply current operating | Port 5 = 0 to $AV_{DD}$ see notes 1 and 2 | | 1.2 | mA |
| | Analog supply current operating: 32 kHz/PLL operation | Port 5 = 0 to $AV_{DD}$ see note 17, 18 | | 7.2 | mA |
| $AI_{ID}$ | Analog supply current Idle Mode | see notes 1 and 3 | | 70 | µA |
| | Analog supply current Idle Mode: 32 kHz/PLL operation | see note 17 | | 6.0 | mA |
| $AI_{PD}$ | Supply current Power-down mode | 2 V < $V_{PD}$ < $V_{DDmax}$ see note 4 | | 50 | µA |
| | Supply current Power-down mode: 32 kHz / PLL operation | $V_{DD}$ = 5.5V see note 17 | | 200 | µA |
| **Analog Inputs** | | | | | |
| $AV_{IN}$ | Analog input voltage | | $AV_{SS}$–0.2 | $AV_{DD}$+0.2 | V |
| $AV_{REF}$ | Reference voltage: $AV_{REF-}$ $AV_{REF+}$ | | $AV_{SS}$–0.2 | $AV_{DD}$+0.2 | V V |
| $R_{REF}$ | Resistance between $AV_{REF+}$ and $AV_{REF-}$ | | 10 | 50 | kΩ |
| $C_{IA}$ | Analog input capacitance | | | 15 | pF |
| $DL_e$ | Differential non-linearity [9, 10, 11] | | | ±1 | LSB |
| $IL_e$ | Integral non-linearity [9, 12] | | | ±2 | LSB |
| $OS_e$ | Offset error [9, 13] | | | ±2 | LSB |
| $G_e$ | Gain error [9, 14] | | | ±0.4 | % |
| $A_e$ | Absolute voltage error [9, 15] | | | ±3 | LSB |
| $M_{CTC}$ | Channel to channel matching | | | ±1 | LSB |
| $C_t$ | Crosstalk between inputs of port 5 [16] | 0–100kHz | | –60 | dB |

**NOTES: See Page 3-631.**

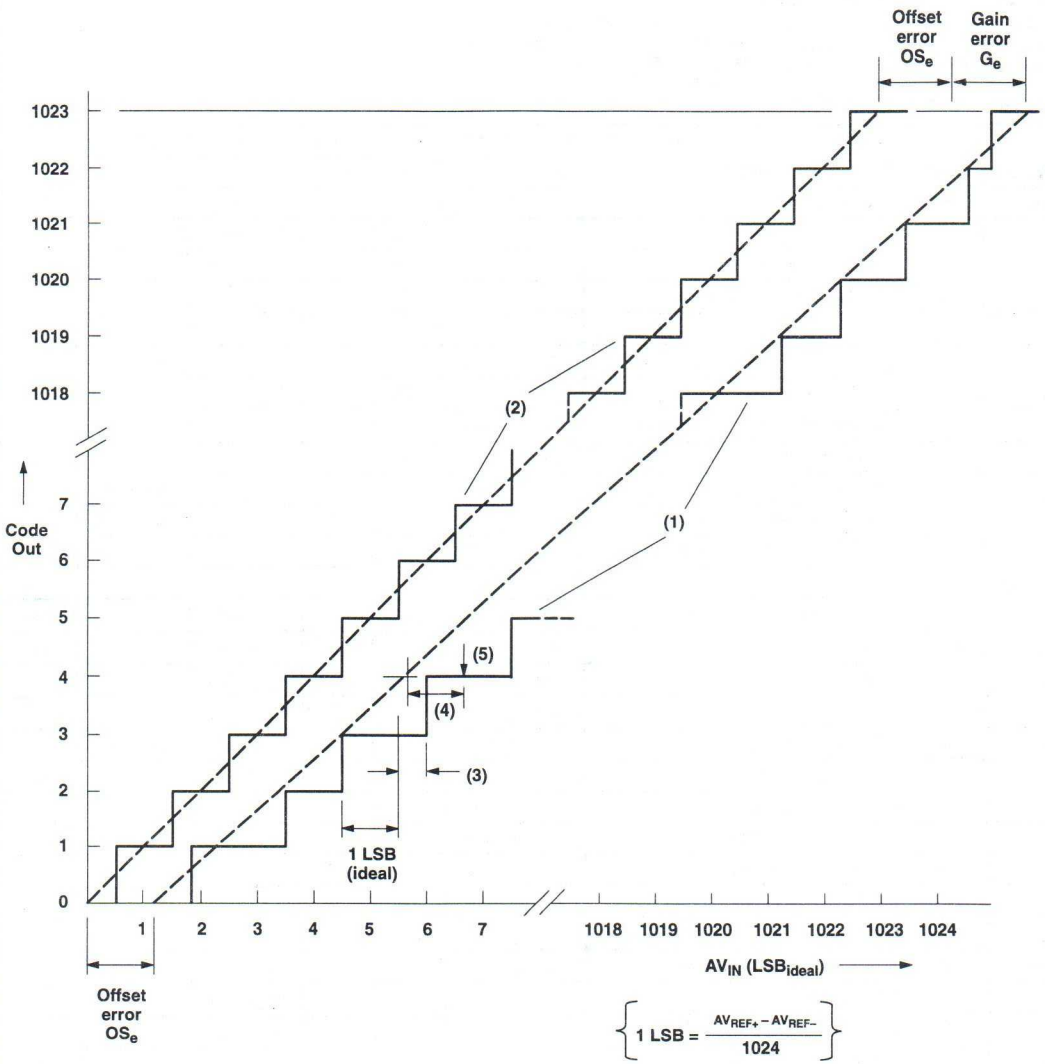
**NOTES FOR DC ELECTRICAL CHARACTERISTICS:**

1. See Figures 55 and 57 through 59 for $I_{DD}$ test conditions.
2. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5$ V; $V_{IH} = V_{DD} - 0.5$ V; XTAL2, XTAL3 not connected; $\overline{EA}$ = RSTIN = Port 0 = $\overline{EW}$ = SCL = SDA = SELXTAL1 = $V_{DD}$; ADEXS = XTAL4 = $V_{SS}$.
3. The Idle Mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5$ V; $V_{IH} = V_{DD} - 0.5$ V; XTAL2, XTAL3 not connected; Port 0 = $\overline{EW}$ = SCL = SDA = SELXTAL 1 = $V_{DD}$; $\overline{EA}$ = RSTIN = ADEXS = XTAL4 = $V_{SS}$.
4. The Power-down current is measured with all output pins disconnected; XTAL2 not connected; Port 0 = $\overline{EW}$ = SCL = SDA = SELXTAL 1 = $V_{DD}$; $\overline{EA}$ = RSTIN = ADEXS = XTAL1 = XTAL4 = $V_{SS}$.
5. The input threshold voltage of SCL and SDA (SIO1) meets the $I^2C$ specification, so an input voltage below 0.3 $V_{DD}$ will be recognized as a logic 0 while an input voltage above 0.7 $V_{DD}$ will be recognized as a logic 1.
6. Pins of ports 1, 2, 3, and 4 source a transition current when they are being externally driven from HIGH to LOW. The transition current reaches its maximum value when $V_{IN}$ is approximately 2 V.
7. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$ of ALE and ports 1, 3 and 4. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.
8. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{DD}$ specification when the address bits are stabilizing.
9. Conditions: $AV_{REF-} = 0$ V; $AV_{DD} = 5.0$ V, $AV_{REF+} = 5.12$ V. $V_{DD} = 5.0$ V, $V_{SS} = 0$ V, ADC is monotonic with no missing codes. Measurement by continuous conversion of $AV_{IN} = -20$mV to 5.12 V in steps of 0.5mV, derivating parameters from collected conversion results of ADC. ADC prescaler programmed according to the actual oscillator frequency, resulting in a conversion time within the specified range for $t_{conv}$ (15µs ... 50µs).
10. The differential non-linearity ($DL_e$) is the difference between the actual step width and the ideal step width.
11. The ADC is monotonic; there are no missing codes.
12. The integral non-linearity ($IL_e$) is the peak difference between the center of the steps of the actual and the ideal transfer curve after appropriate adjustment of gain and offset error.
13. The offset error ($OS_e$) is the absolute difference between the straight line which fits the actual transfer curve (after removing gain error), and a straight line which fits the ideal transfer curve. The offset error is constant at every point of the actual transfer curve.
14. The gain error ($G_e$) is the relative difference in percent between the straight line fitting the actual transfer curve (after removing offset error), and the straight line which fits the ideal transfer curve. Gain error is constant at every point on the transfer curve.
15. The absolute voltage error ($A_e$) is the maximum difference between the center of the steps of the actual transfer curve of the non-calibrated ADC and the ideal transfer curve.
16. This should be considered when both analog and digital signals are simultaneously input to port 5.
17. The supply current with 32 kHz oscillator running and PLL operation (SELXTAL1 = 0) is measured with all output pins disconnected; XTAL4 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5$ V; $V_{IH} = V_{DD} - 0.5$ V; XTAL2 not connected; Port 0 = $\overline{EW}$ = SCL = SDA = $V_{DD}$; $\overline{EA}$ = RSTIN = ADEXS = SELXTAL 1 = XTAL1 = $V_{SS}$.
18. Not 100% tested; sum of $A_{IID}$ (PLL) and $A_{IDD}$ (HF-Oscillator).
19. The parameter meets the $I^2C$ bus specification for standard-mode and fast-mode devices.
20. Not 100% tested.




(1) Maximum operating mode P89CE558 : $V_{DD}$ = 5.5 V
(2) Maximum operating mode P83CE558/P80CE558 : $V_{DD}$ = 5.5 V
(3) Maximum Idle Mode P89CE558 : $V_{DD}$ = 5.5 V
(4) Maximum Idle Mode P83CE558/P80CE558 : $V_{DD}$ = 5.5 V

**Figure 55. Supply Current ($I_{DD}$) as a Function of Frequency at XTAL1**




(1) Example of an actual transfer curve.
(2) The ideal transfer curve.
(3) Differential non-linearity ($DL_e$).
(4) Integral non-linearity ($IL_e$).
(5) Center of a step of the actual transfer curve.

**Figure 56. ADC Conversion Characteristic**

## 11. AC CHARACTERISTICS

### AC ELECTRICAL CHARACTERISTICS

$V_{DD}$ = 5 V ± 10% (EBx), $V_{SS}$ = 0 V, $t_{CLK}$ min = 1/fmax (maximum operating frequency)
$V_{DD}$ = 5 V ± 10% (EFx), $V_{SS}$ = 0 V, $t_{CLK}$ min = 1/fmax (maximum operating frequency)
$T_{amb}$ = 0 °C to +70 °C, $t_{CLK}$ min = 63 ns for P8xCE558EBx
$T_{amb}$ = –40 °C to +85 °C, $t_{CLK}$ min = 63 ns for P8xCE558EFx
C1 = 100 pF for Port 0, ALE and $\overline{PSEN}$ ; C1 = 80 pF for all other outputs unless otherwise specified.

| SYMBOL | FIGURE | PARAMETER | 12MHz CLOCK | | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | MIN | MAX | |
| $1/t_{CLK}$ | 60 | System clock frequency | | | | | 3.5 | 16 | MHz |
| $t_{LHLL}$ | 60 | ALE pulse width | 127 | | 85 | | $2t_{CLK}-40$ | | ns |
| $t_{AVLL}$ | 60 | Address valid to ALE LOW | 43 | | 23 | | $t_{CLK}-40$ | | ns |
| $t_{LLAX}$ | 60 | Address hold after ALE LOW | 53 | | 33 | | $t_{CLK}-30$ | | ns |
| $t_{LLIV}$ | 60 | ALE LOW to valid instruction in | | 234 | | 150 | | $4t_{CLK}-100$ | ns |
| $t_{LLPL}$ | 60 | ALE LOW to $\overline{PSEN}$ LOW | 53 | | 33 | | $t_{CLK}-30$ | | ns |
| $t_{PLPH}$ | 60 | $\overline{PSEN}$ pulse width | 205 | | 143 | | $3t_{CLK}-45$ | | ns |
| $t_{PLIV}$ | 60 | $\overline{PSEN}$ LOW to valid instruction in | | 145 | | 83 | | $3t_{CLK}-105$ | ns |
| $t_{PXIX}$ | 60 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 60 | Input instruction float after $\overline{PSEN}$ | | 59 | | 38 | | $t_{CLK}-25$ | ns |
| $t_{AVIV}$ | 60 | Address to valid instruction in | | 312 | | 208 | | $5t_{CLK}-105$ | ns |
| $t_{PLAZ}$ | 60 | $\overline{PSEN}$ LOW to address float | | 10 | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | | | |
| $t_{AVLL}$ | 61, 62 | Address valid to ALE LOW | 43 | | 23 | | $t_{CLK}-40$ | | ns |
| $t_{LLAX}$ | 61, 62 | Address hold after ALE LOW | 48 | | 28 | | $t_{CLK}-35$ | | ns |
| $t_{RLRH}$ | 61 | $\overline{RD}$ pulse width | 400 | | 275 | | $6t_{CLK}-100$ | | ns |
| $t_{WLWH}$ | 62 | $\overline{WR}$ pulse width | 400 | | 275 | | $6t_{CLK}-100$ | | ns |
| $t_{RLDV}$ | 61 | $\overline{RD}$ LOW to valid data in | | 252 | | 148 | | $5t_{CLK}-165$ | ns |
| $t_{RHDX}$ | 61 | Data hold after $\overline{RD}$ | 0 | | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 61 | Data float after $\overline{RD}$ | | 97 | | 55 | | $2t_{CLK}-70$ | ns |
| $t_{LLDV}$ | 61 | ALE LOW to valid data in | | 517 | | 350 | | $8t_{CLK}-150$ | ns |
| $t_{AVDV}$ | 61 | Address to valid data in | | 585 | | 398 | | $9t_{CLK}-165$ | ns |
| $t_{LLWL}$ | 61, 62 | ALE LOW to $\overline{RD}$ or $\overline{WR}$ LOW | 200 | 300 | 138 | 238 | $3t_{CLK}-50$ | $3t_{CLK}+50$ | ns |
| $t_{AVWL}$ | 61, 62 | Address valid to $\overline{WR}$ LOW or $\overline{RD}$ LOW | 203 | | 120 | | $4t_{CLK}-130$ | | ns |
| $t_{QVWX}$ | 62 | Data valid to $\overline{WR}$ transition | 33 | | 13 | | $t_{CLK}-50$ | | ns |
| $t_{QVWH}$ | 62 | Data before $\overline{WR}$ | 433 | | 288 | | $7t_{CLK}-150$ | | ns |
| $t_{WHQX}$ | 62 | Data hold after $\overline{WR}$ | 33 | | 13 | | $t_{CLK}-50$ | | ns |
| $t_{RLAZ}$ | 61 | $\overline{RD}$ low to address float | | 0 | | 0 | | 0 | ns |
| $t_{WHLH}$ | 61, 62 | $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH | 43 | 123 | 23 | 103 | $t_{CLK}-40$ | $t_{CLK}+40$ | ns |
| **UART Timing – Shift Register Mode** (Test Conditions: $T_{amb}$ = 0 °C to +70 °C; $V_{SS}$ = 0 V; Load Capacitance = 80pF) | | | | | | | | | |
| $t_{XLXL}$ | 64 | Serial port clock cycle time | 1.0 | | 0.75 | | $12t_{CLK}$ | | µs |
| $t_{QVXH}$ | 64 | Output data setup to clock rising edge | 700 | | 492 | | $10t_{CLK}-133$ | | ns |
| $t_{XHQX}$ | 64 | Output data hold after clock rising edge | 50 | | 8 | | $2t_{CLK}-117$ | | ns |
| $t_{XHDX}$ | 64 | Input data hold after clock rising edge | 0 | | 0 | | 0 | | ns |
| $t_{XHDV}$ | 64 | Clock rising edge to input data valid | | 700 | | 492 | | $10t_{CLK}-133$ | ns |

## AC ELECTRICAL CHARACTERISTICS (Continued)

| SYMBOL | PARAMETER | Standard-mode I²C-bus | | Fast-mode I²C-bus | | UNIT |
|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | |
| **I²C Interface timing (refer to Figure 63)** | | | | | | |
| $f_{SCL}$ | SCL clock frequency | 0 | 100 | 0 | 400 | kHz |
| $t_{BUF}$ | Bus free time between a STOP and START condition | 4.7 | – | 1.3 | – | µs |
| $t_{HD;\ STA}$ | Hold time (repeated) START condition. After this period, the first clock pulse is generated | 4.0 | – | 0.6 | – | µs |
| $t_{LOW}$ | LOW period of the SCL clock | 4.7 | – | 1.3 | – | µs |
| $t_{HIGH}$ | High period of the SCL clock | 4.0 | – | 0.6 | – | µs |
| $t_{SU;\ STA}$ | Set-up time for a repeated START condition | 4.7 | – | 0.6 | – | µs |
| $t_{HD;\ DAT}$ | Data hold time:<br>for CBUS competible masters (see Section 9, Notes 1, 3)<br>for I²C-bus devices | <br>5.0<br>0 [1] | | <br>–<br>0[1] | <br>–<br>0.9[2] | µs |
| $t_{SU;\ DAT}$ | Data set-up time | 250 | – | 100[3] | – | ns |
| $t_{FD}$, $t_{FC}$ | Rise time of both SDA and SCL signals | – | 1000 | 20 + 0.1$C_b$[4] | 300 | ns |
| $t_{FD}$, $t_{FC}$ | Fall time of both SDA and SCL signals | – | 300 | 20 + 0.1$C_b$[4] | 300 | ns |
| $t_{SU;\ STO}$ | Set-up time for STOP condition | 4.0 | – | 0.6 | – | µs |
| $C_b$ | Capacitive load for each bus line | – | 400 | – | 400 | pF |
| $t_{SP}$ | Pulse width of spikes which must be suppressed by the input filter | – | – | 0 | 50 | ns |

All values referred to $V_{IH}$ and $V_{IL\ max}$ levels.

**NOTES:**

1. A device must internally provide a hold time of at least 300 ns from the SDA signal (referred to the $V_{IH\ min}$ of the SCL signal) in order to bridge the undefined region of the falling edge of SCL.
2. The maximum $t_{HD,DAT}$ has only to be met if the device does not stretch the LOW period ($t_{LOW}$) of the SCL signal.
3. A fast-mode I²C-bus device can be used in a standard-mode I²C-bus system, but the requirement $t_{SU,DAT} \geq 250$ ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line $t_{Rmax} + t_{SU,DAT} = 1000 + 250 = 1250$ ns (according to the standard-mode I²C-bus specification) before the SCL line is released.
4. $C_b$ = total capacitance of one bus line in pF.

### Table 46. External clock drive XTAL1 (refer to Figure 57)

| SYMBOL | PARAMETER | VARIABLE CLOCK $f_{CLK}$ = 3.5 to 16 MHz | | UNIT |
|---|---|---|---|---|
| | | MIN | MAX | |
| $t_{CLK}$ | XTAL1 Period | 63 | 286 | ns |
| $t_{CLKH}$ | XTAL1 HIGH time | 20 | – | ns |
| $t_{CLKL}$ | XTAL1 LOW time | 20 | – | ns |
| $t_{CLKR}$ | XTAL1 rise time | – | 20 | ns |
| $t_{CLKF}$ | XTAL1 fall time | – | 20 | ns |
| $t_{CYC}$ [1)] | Controller cycle time | 0.75 | 3.4 | µs |

**NOTE:**

1. $t_{CYC} = 12\ f_{CLK}$

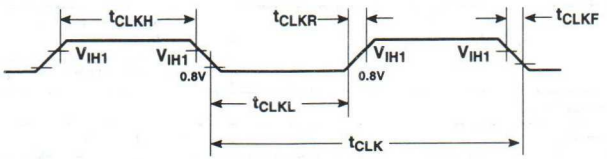
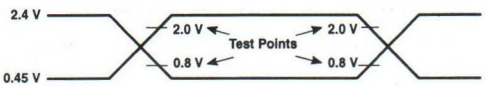
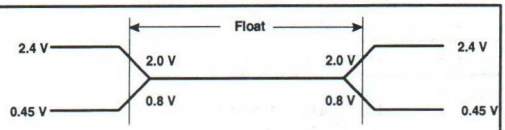



Figure 57. External Clock Drive waveform




NOTE:
AC inputs during testing are driven at 2.4V for a logic 'HIGH' and 0.45V for a logic 'LOW'. Timing measurements are made at 2.0 V for a logic 'HIGH' and 0.8 V for a logic 'LOW'.

Figure 58. AC Testing Input/Output




NOTE:
The float state is defined as the point at which a port 0 pins sinks 3.2 mA or sources 400µA at the voltage test levels.

Figure 59. AC Testing, Float Waveform

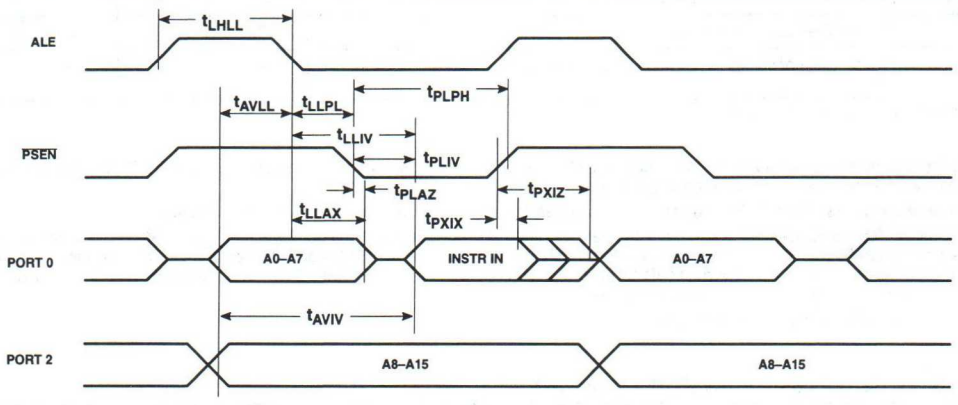



Figure 60. External Program Memory Read Cycle

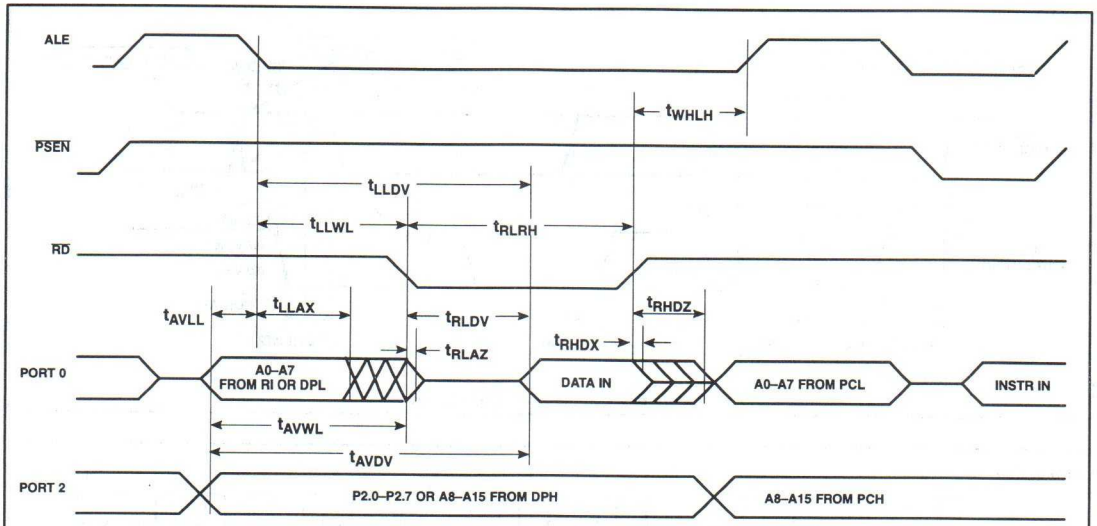



Figure 61. External Data Memory Read Cycle

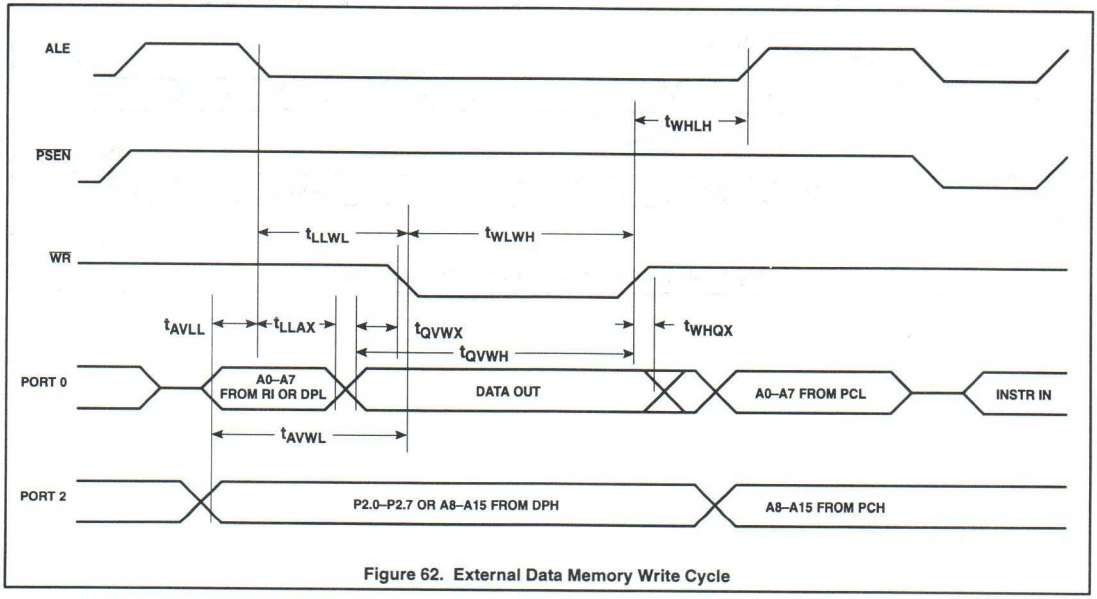



Figure 62. External Data Memory Write Cycle

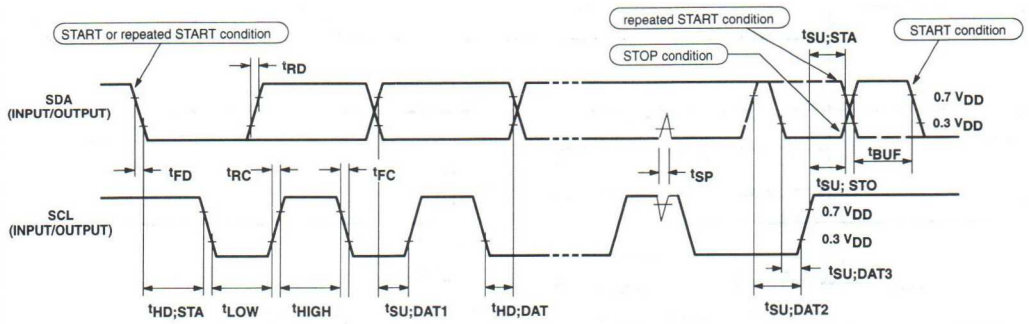



Figure 63. Timing SIO1 ($I^2C$) Interface

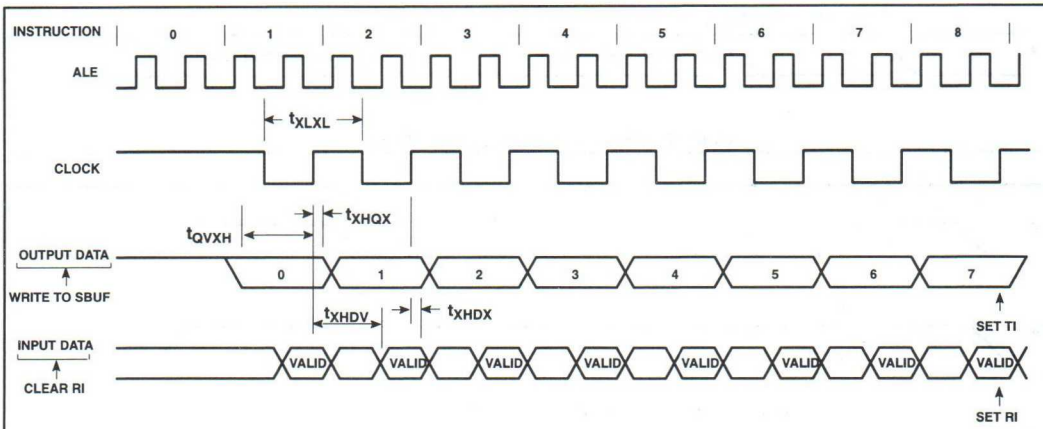



Figure 64. UART waveforms in Shift Register Mode

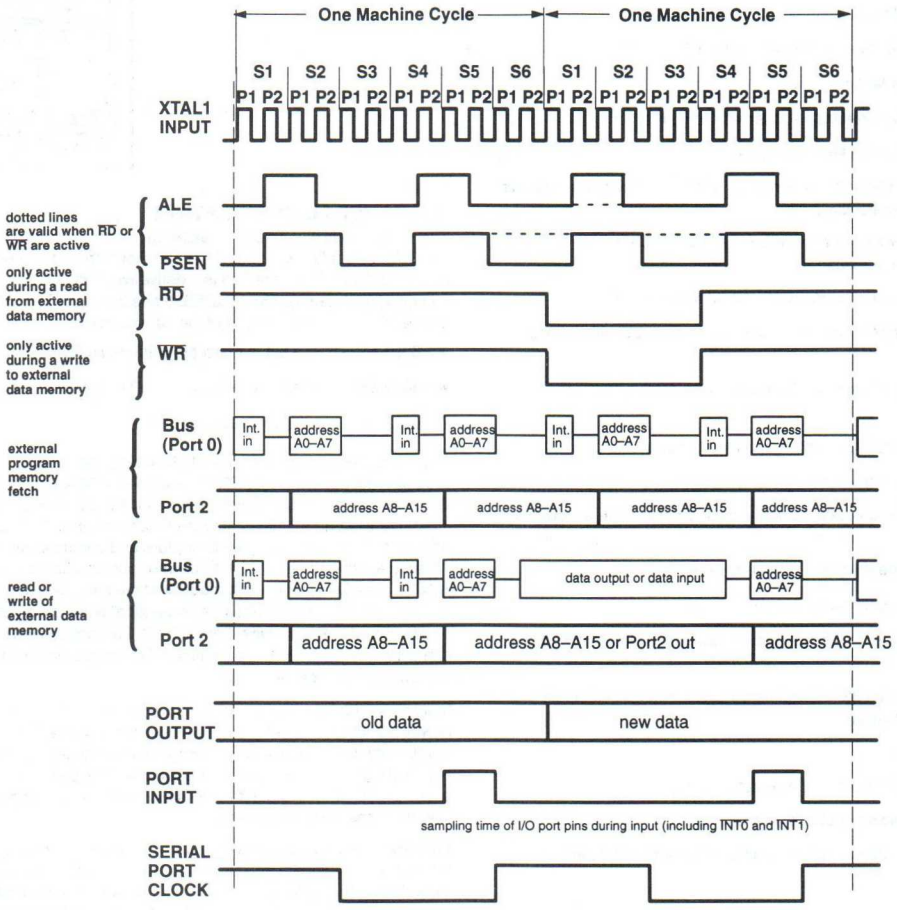



**Figure 65. Instruction cycle timing**

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# Single-chip 8-bit microcontroller P83CE559/P80CE559

## 1. FEATURES

- 80C51 central processing unit
- 48 K × 8 ROM, expandable externally to 64 Kbytes
- ROM Code protection
- 1536 × 8 RAM, expandable externally to 64 Kbytes
- Two standard 16-bit timer/counters
- An additional 16-bit timer/counter coupled to four capture registers and three compare registers
- A 10-bit ADC with eight multiplexed analog inputs and programmable autoscan
- Two 8-bit resolution, pulse width modulation outputs
- Five 8-bit I/O ports plus one 8-bit input port shared with analog inputs
- I²C-bus serial I/O port with byte oriented master and slave functions
- Full-duplex UART compatible with the standard 80C51
- On-chip watchdog timer
- 15 interrupt sources with 2 priority levels (2 to 6 external sources possible)
- Extended temperature range (–40 to +85 °C)
- 4.5 to 5.5 V supply voltage range
- Frequency range for 80C51–family standard oscillator: 3.5 MHz to 16 MHz
- PLL oscillator with 32 kHz reference and software–selectable system clock frequency
- Seconds Timer
- Software enable/disable of ALE output pulse
- Electromagnetic compatibility improvements
- Wake–up from Power–down by external or seconds interrupt

## 2. GENERAL DESCRIPTION

The P80CE559/P83CE559 (hereafter generically referred to as P8xCE559) single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The P8xCE559 has the same instruction set as the 80C51. Three versions of the derivative exist:

- P83CE559 — 48 Kbytes mask programmable ROM
- P80CE559 — ROMless version of the P83CE559
- P89CE559 — not planned any longer

The P8xCE559 contains a non-volatile 48 Kbytes mask programmable ROM (P83CE559), a volatile 1536 × 8 read/write data memory, five 8-bit I/O ports, one 8-bit input port, two 16-bit timer/event counters (identical to the timers of the 80C51), an additional 16-bit timer coupled to capture and compare latches, a 15-source, two-priority-level, nested interrupt structure, an 8-input ADC, a dual DAC pulse width modulated interface, two serial interfaces (UART and I²C-bus), a "watchdog" timer, an on-chip oscillator and timing circuits. For systems that require extra capability the P8xCE559 can be expanded using standard TTL compatible memories and logic.

In addition, the P8xCE559 has two software selectable modes of power reduction — Idle Mode and power-down mode. The Idle Mode freezes the CPU while allowing the RAM, timers, serial ports, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic as well as bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte, and 17 three- byte. With a 16 MHz system clock, 58% of the instructions are executed in 0.75 µs and 40% in 1.5 µs. Multiply and divide instructions require 3 µs.

## 3. ORDERING INFORMATION

| EXTENDED TYPE NUMBER | PACKAGE | | | FREQUENCY RANGE (MHz) | TEMPERATURE RANGE (°C) |
|---|---|---|---|---|---|
| | NAME | DESCRIPTION | CODE | | |
| **ROMless** | | | | | |
| P80CE559EBB | QFP80 | Plastic Quad Flat Pack; 80 leads | SOT318-1 | 3.5 to 16 | 0 to +70 |
| P80CE559EFB | QFP80 | Plastic Quad Flat Pack; 80 leads | SOT318-1 | 3.5 to 16 | –40 to +85 |
| **ROM coded** | | | | | |
| P83CE559EBB/YYY[1] | QFP80 | Plastic Quad Flat Pack; 80 leads | SOT318-1 | 3.5 to 16 | 0 to +70 |
| P83CE559EFB/YYY[1] | QFP80 | Plastic Quad Flat Pack; 80 leads | SOT318-1 | 3.5 to 16 | –40 to +85 |

**NOTE:**
1. YYY denotes the ROM code number.

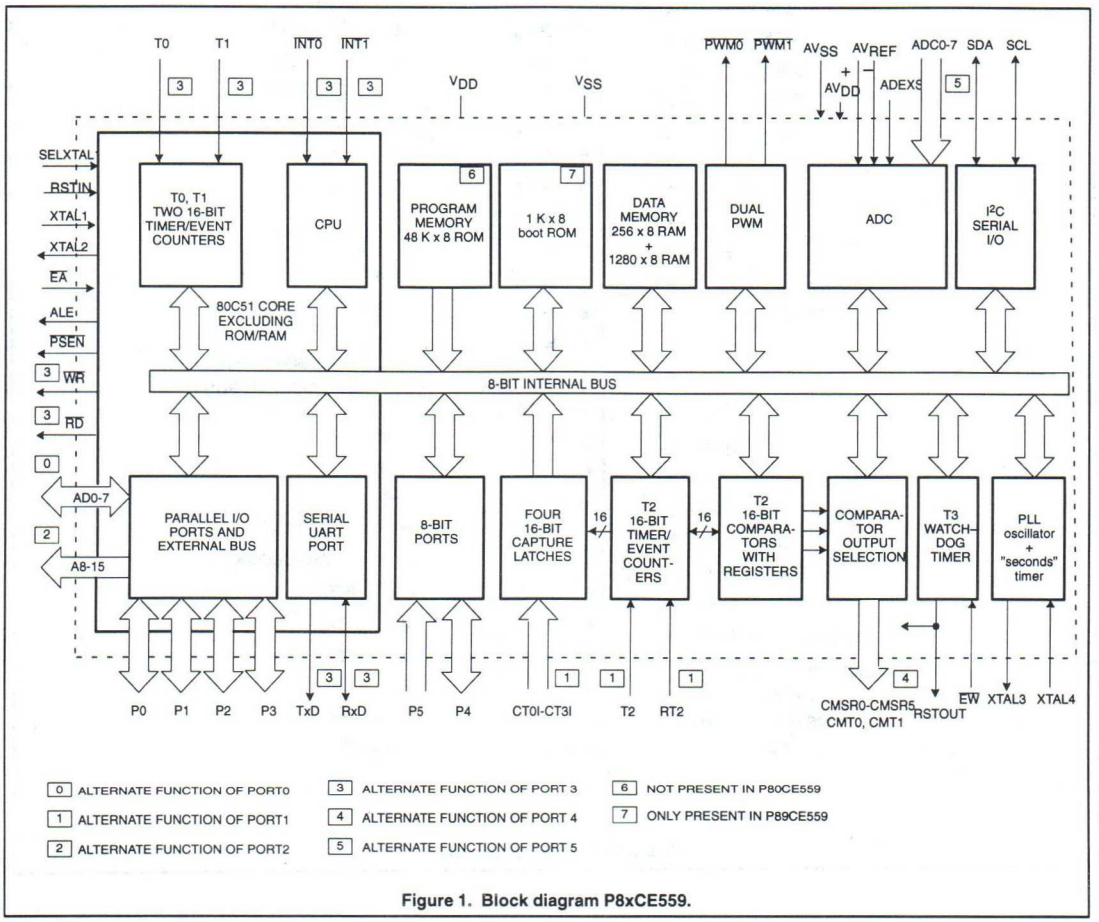



**Figure 1. Block diagram P8xCE559.**

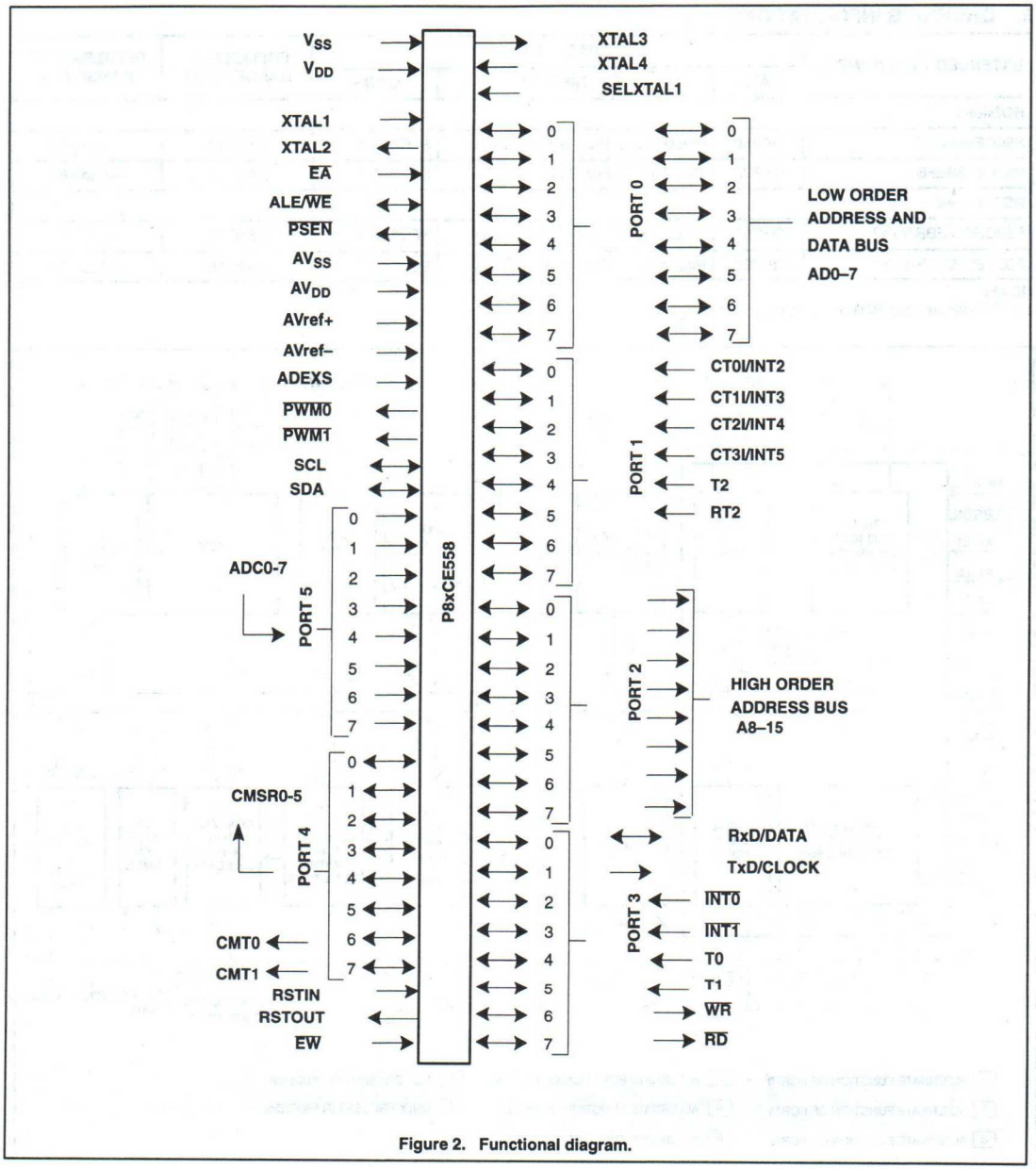



**Figure 2. Functional diagram.**

## 4. PINNING

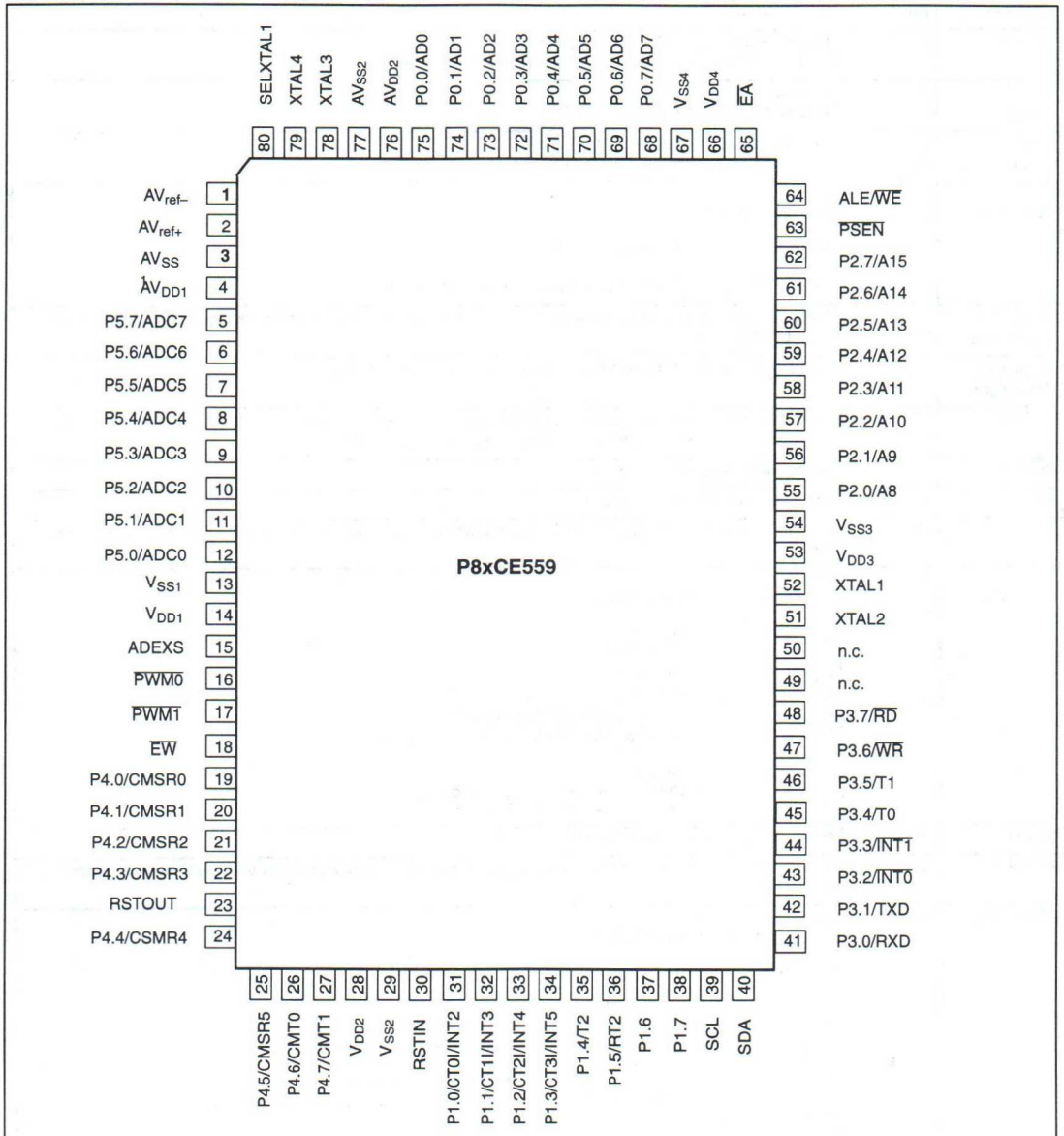



**Figure 3. Pinning diagram for QFP80 (SOT318).**

## 4.1 PIN DESCRIPTIONS

| SYMBOL | PIN | DESCRIPTION |
|---|---|---|
| $AV_{ref-}$<br>$AV_{ref+}$ | 1<br>2 | **Low end** of analog to digital conversion reference resistor<br>**High end** of analog to digital conversion reference resistor. |
| $AV_{SS1}$<br>$AV_{DD1}$ | 3<br>4 | **Analog** ground for ADC<br>**Analog** power supply (+5 V) for ADC |
| $AV_{SS2}$<br>$AV_{DD2}$ | 77<br>76 | **Analog** ground; for PLL oscillator<br>**Analog** power supply; (+5 V) for PLL oscillator |
| P5.7 – P5.0 | 5 – 12 | **Port 5**<br>8–bit input port<br>Port pin: Alternative function<br>P5.0–P5.7: Eight input channels to ADC (ADC0–ADC7) |
| $V_{DD1}$, $V_{DD2}$,<br>$V_{DD3}$, $V_{DD4}$ | 14, 28,<br>53, 66 | **Digital power supply:** +5 V power supply pins during normal operation and power reduction modes. All pins must be connected. |
| $V_{SS1}$, $V_{SS2}$<br>$V_{SS3}$, $V_{SS4}$ | 13, 29,<br>54, 67 | **Digital ground:** circuit ground potential. All pins must be connected. |
| ADEXS | 15 | **Start ADC operation:** Input starting analog to digital conversion triggered by a programmable edge (ADC operation can also be started by software). This pin must not float |
| $\overline{PWM0}$ | 16 | **Pulse width modulation** output 0 |
| $\overline{PWM1}$ | 17 | **Pulse width modulation** output 1 |
| $\overline{EW}$ | 18 | **Enable watchdog timer:** Enable for T3 watchdog timer and disable Power–down Mode.This pin must not float. |
| P4.0 – P4.7 | 19 – 22<br>24 – 27 | **Port 4**<br>8–bit quasi–bidirectional I/O port<br>P4.0: CMSR0 } compare and set/reset<br>P4.1: CMSR1 }<br>P4.2: CMSR2 }<br>P4.3: CMSR3 } outputs on a match with timer T2<br>P4.4: CMSR4 }<br>P4.5: CMSR5 }<br>P4.6: CMT0 } compare and toggle outputs<br>P4.7: CMT1 } on a match with timer T2 |
| RSTIN | 30 | **Reset:** Input to reset the P8xCE559. |
| RSTOUT | 23 | **Reset:** Output of the P8xCE559 for resetting peripheral devices during initialization and Watchdog Timer overflow. |
| P1.0 – P1.7 | 31 – 38 | **Port 1**<br>8–bit quasi–bidirectional I/O port<br>P1.0: CT0I/INT2}<br>P1.1: CT1I/INT3} : Capture timer inputs for<br>P1.2: CT2I/INT4} timer T2 or external interrupt inputs<br>P1.3: CT3I/INT5}<br>P1.4: T2 : T2 event input, rising edge triggered<br>P1.5: RT2 : T2 timer reset input, rising edge triggered<br>P1.6<br>P1.7 |
| SCL | 39 | **I²C–bus serial clock** I/O port |
| SDA | 40 | **I²C–bus serial data** I/O port<br>If SCL and SDA are not used, they must be connected to $V_{SS}$. |

| SYMBOL | PIN | DESCRIPTION |
|---|---|---|
| P3.0 – P3.7 | 41 – 48 | 8–bit quasi–bidirectional I/O port<br>Port pin — Alternative function<br>P3.0 RXD : Serial input port<br>P3.1 TXD : Serial output port<br>P3.2 $\overline{INT0}$ : External interrupt<br>P3.3 $\overline{INT1}$ : External interrupt<br>P3.4 T0 : Timer 0 external input<br>P3.5 T1 : Timer 1 external input<br>P3.6 $\overline{WR}$ : External data memory write strobe<br>P3.7 $\overline{RD}$ : External data memory read strobe |
| N.C. | 49 – 50 | Not connected pins. |
| XTAL2 | 51 | **Crystal pin 2**: output of the inverting amplifier that forms the oscillator. Left open–circuit when an external oscillator clock is used. |
| XTAL1 | 52 | **Crystal pin 1**: input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external oscillator clock signal when an external oscillator is used. Must be connected to logic HIGH if the PLL oscillator is selected (SELXTAL1 = LOW) |
| P2.0 – P2.7 | 55 – 62 | **Port2:** 8–bit quasi–bidirectional I/O port with internal pull–ups.During access to external memories (RAM/ROM) that use 16–bit addresses (MOVX@DPTR) Port 2 emits the high order address byte.<br>The alternative function of P2.7 for the P89CE559 is the output enable signal for verify/read modes (active low).<br><br>Port 2 can sink/source one TTL (=4 LSTTL) input. It can drive CMOS inputs without external pull–ups. |
| $\overline{PSEN}$ | 63 | **Program Store Enable output:** read strobe to the external program memory via Port 0 and 2. Is activated twice each machine cycle during fetches from external program memory. When executing out of external program memory two activations of $\overline{PSEN}$ are skipped during each access to external data memory. $\overline{PSEN}$ is not activated (remains HIGH) during no fetches from external program memory. $\overline{PSEN}$ can sink/source 8 LSTTL inputs. It can drive CMOS inputs without external pull–ups. |
| ALE/$\overline{WE}$ | 64 | **Address Latch Enable output**: latches the low byte of the address during access of external memory in normal operation. It is activated every six oscillator periods except during an external data memory access. ALE/$\overline{WE}$ can sink/–source 8 LSTTL inputs.<br>It can drive CMOS inputs without an external pull–up.<br>The alternative function for the P89CE559 is the programming pulse input $\overline{WE}$.<br><br>To prohibit the toggling of ALE pin (RFI noise reduction) the bit RFI in the PCON Register (PCON.5) must be set by software. This bit is cleared on RESET and can be set and cleared by software. When set, ALE pin will be pulled down internally, switching an external address latch to a quiet state. The MOVX instruction will still toggle ALE if external memory is accessed.<br><br>ALE will retain its normal high value during Idle Mode and a low value during Power–down Mode while in the "RFI" mode. Additionally during internal access ($\overline{EA}$ = 1) ALE will toggle normally when the address exceeds the internal program memory size. During external access ($\overline{EA}$ = 0) ALE will always toggle normally, whether the flag "RFI" is set or not. |
| $\overline{EA}$ | 65 | **External Access Input:** If, during RESET, $\overline{EA}$ is held at a TTL level HIGH the CPU executes out of the internal program memory, provided the program counter is less than 49152. If, during RESET, $\overline{EA}$ is held at a TTL level LOW the CPU executes out of external program memory via Port 0 and Port 2. $\overline{EA}$ is not allowed to float. $\overline{EA}$ is latched during RESET and don't care after RESET. |
| P0.7–P0.0 | 68 –75 | **Port 0:** 8–bit open drain bidirectional I/O port. It is also the multiplexed low–order address and data bus during accesses to external memory (during theses accesses internal pull–ups are activated). Port 0 can sink/source 8 LSTTL inputs. |
| XTAL3 | 78 | **Crystal pin**, output of the inverting amplifier that forms the 32 kHz oscillator |
| XTAL4 | 79 | **Crystal pin**, input to the inverting amplifier that forms the 32 kHz oscillator. XTAL3 and XTAL4 are pulled LOW if the PLL oscillator is not selected (SELXTAL1 = HIGH) or if Reset is active. |
| SELXTAL1 | 80 | Must be connected to logic HIGH level to select the HF oscillator, using the XTAL1/XTAL2 crystal. If pulled low the PLL is selected for clocking of the controller, using the XTAL3/ XTAL4 crystal. |

**NOTE:**

1. To avoid a 'latch–up' effect at Power–on, the voltage at any pin at any time must not be higher or lower than $V_{DD}$+ 0.5 V or $V_{SS}$– 0.5 V respectively.

## 5. ELECTROMAGNETICS COMPATIBILITY (EMC) IMPROVEMENTS

Primary attention was paid on the reduction of electromag– netic emission of the microcontroller P8xCE559.

The following features effect in reducing the electromag– netic emission and additionally improve the electromagnetic susceptibility:

- Four supply voltage pins ($V_{DD}$) and four ground pins ($V_{SS}$) with pairs of $V_{DD}$ and $V_{SS}$ at two adjacent pins at each side of the package.
- Separated $V_{DD}$ pins for the internal logic and the port buffers
- Internal decoupling capacitance improves the EMC radiation behavior and the EMC immunity
- External capacitors are to be located as close as possible between pins $V_{DD1}$ and $V_{SS1}$, $V_{DD2}$ and $V_{SS2}$, $V_{DD3}$ and $V_{SS3}$ as well as $V_{DD4}$ and $V_{SS4}$ ; ceramic chip capacitors are recommended (100nF).
- The ALE output signal (pulses at a frequency off $_{CLK}$/6) can be disabled under software control (bit 5 in the SFR PCON: "RFI"); if disabled, no ALE pulse will occur. ALE pin will be pulled down internally, switching an external address latch to a quiet state. The MOVX instruction will still toggle ALE (external data memory is accessed). ALE will retain its normal HIGH value during Idle Mode and a LOW value during Power-down mode while in the "RFI" reduction mode. Additionally during internal access ($\overline{EA}$ = 1) ALE will toggle normally when the address exceeds the internal program memory size. During external access ($\overline{EA}$ = 0) ALE will always toggle normally, whether the flag "RFI" is set or not.

## 6. FUNCTIONAL DESCRIPTION

### 6.1 General

The P8xCE559 is a stand–alone high–performance microcontroller designed for use in real time applications such as instrumentation, industrial control, medium to high–end consumer applications and specific automotive control applications.

In addition to the 80C51 standard functions, the device provides a number of dedicated hardware functions for these applications.

The P8xCE559 is a control–oriented CPU with on–chip program and data memory. It can be extended with external program memory up to 64 Kbytes. It can also access up to 64 Kbytes of external data memory. For systems requiring extra capability, the P8xCE559 can be expanded using standard memories and peripherals.

The P8xCE559 has two software selectable modes of reduced activity for further power reduction – Idle and Power–down. The Idle Mode freezes the CPU while allowing the RAM, timers, serial ports and interrupt system to continue functioning. The Power–down Mode saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative.The Power–down Mode can be terminated by an external Reset, by the seconds interrupt and by any one of the two external interrupts. (see description Wake–up from Power–down Mode).

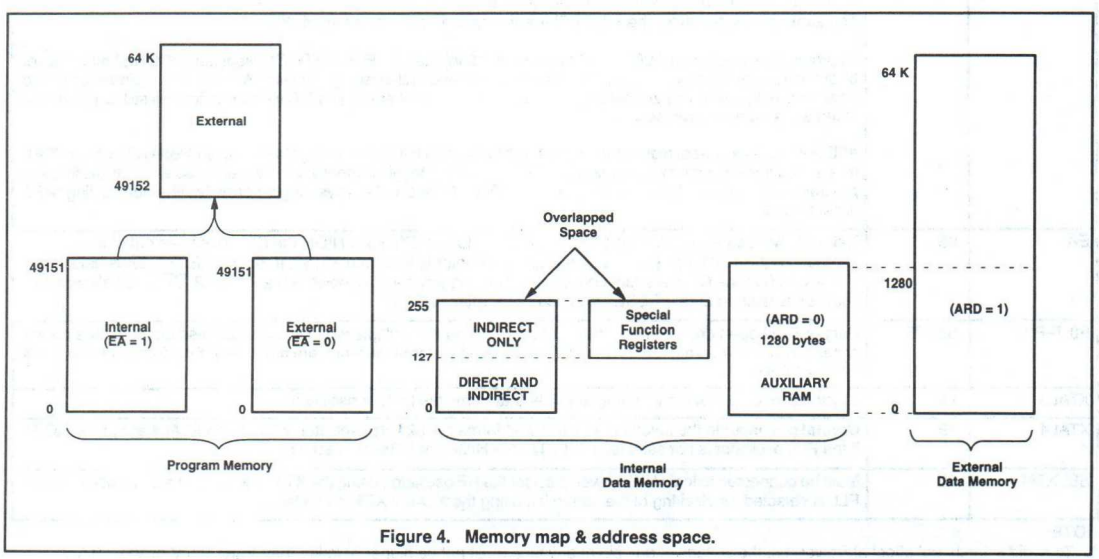



**Figure 4. Memory map & address space.**

## 6.2 Memory Organization

The central processing unit (CPU) manipulates operands in three memory spaces; these are the 64 Kbytes external data memory, 1536 bytes internal data memory (consisting of 256 bytes standard RAM and 1280 bytes AUX–RAM) and the 48 Kbytes internal and/or 64 Kbytes external program memory (see Figure 4).

### 6.2.1 Program Memory

The program memory of the P8xCE559 consists of 48 Kbytes ROM respectively FEEPROM ("Flash Memory") on–chip, externally expandable up to 64 Kbytes. If, during RESET, the $\overline{EA}$ pin was held HIGH, the P8xCE559 executes out of the internal program memory unless the address exceeds 0BFFFH. Locations 0C000H through 0FFFFH are then fetched from the external program memory. If the $\overline{EA}$ pin was held LOW during RESET the P8xCE559 fetches all instructions from the external program memory. The $\overline{EA}$ input is latched during RESET and is don't care after RESET.

The internal program memory content is protected, by setting a mask programmable security bit (ROM) or by the software programmable security byte (FEEPROM) respectively, i.e., it cannot be read out at any time by any test mode or by any instruction in the external program memory space. The MOVC instructions are the only ones which have access to program code in the internal or external program memory. The $\overline{EA}$ input is latched during RESET and is 'don't care' after RESET. This implementation prevents from reading internal program code by switching from external program memory to internal program memory during MOVC instruction or an instruction that handles immediate data. Table 1 lists the access to the internal and external program memory with MOVC instructions when the security feature has been activated.

### 6.2.2 Internal Data Memory

The internal data memory is divided into three physically separated parts:

256 bytes of RAM, 1280 bytes of AUX–RAM, and a 128 bytes special function area. These can be addressed each in a different way (see also Table 2).

- RAM 0 to 127 can be addressed directly and indirectly as in the 80C51.

  Address pointers are R0 and R1 of the selected registerbank.
- RAM 128 to 255 can only be addressed indirectly.

  Address pointers are R0 and R1 of the selected registerbank.
- AUX–RAM 0 to 1279 is also indirectly addressable as external DATA MEMORY locations 0 to 1279 via MOVX–Datapointer instruction, unless it is disabled by setting ARD = 1.

  AUX–RAM 0 to 1279 is indirectly addressable via pageregister (XRAMP) and MOVX–Ri instructions, unless it is disabled by setting ARD = 1 (see Figure 5).

  When executing from internal program memory, an access to AUX–RAM 0 to 1279 will not affect the ports P0, P2, P3.6 and P3.7.

  An access to external DATA MEMORY locations higher than 1279 will be performed with the MOVX @ DPTR instructions in the same way as in the 80C51 structure, so with P0 and P2 as data/address bus and P3.6 and P3.7 as write and read timing signals. Note that the external DATA MEMORY cannot be accessed with R0 and R1 as address pointer if the AUX–RAM is enabled (ARD = 0, default).
- The Special Function Registers (SFR) can only be addressed directly in the address range from 128 to 255 (see Table 5).
- Four register banks, each 8 registers wide, occupy locations 0 through 31 in the lower RAM area. Only one of these banks may be enabled at a time. The next 16 bytes, locations 32 through 47, contain 128 directly addressable bit locations.The stack can be located anywhere in the internal 256 bytes RAM.The stack depth is only limited by the available internal RAM space of 256 bytes (see Figure 7).

  All registers except the program counter and the four register banks reside in the Special Function Register address space.

**Table 1. Memory access by the MOVC instruction for protected ROMs**

| MOVC LOCATION | ACCESS TO INTERNAL PROGRAM MEMORY | ACCESS TO EXTERNAL PROGRAM MEMORY |
|---|---|---|
| MOVC in internal program memory | YES | YES |
| MOVC in external program memory | NO | YES |

**NOTE:**
1. If the security feature has not been activated, there are no restrictions for MOVC instructions.

**Table 2. Internal data memory map**

| LOCATION | | ADDRESSED |
|---|---|---|
| RAM | 0 to 127 | Direct and indirect |
| AUX–RAM | 0 to 1279 | Indirect only with MOVX |
| RAM | 128 to 255 | Indirect only |
| SFR | 128 to 255 | Direct only |

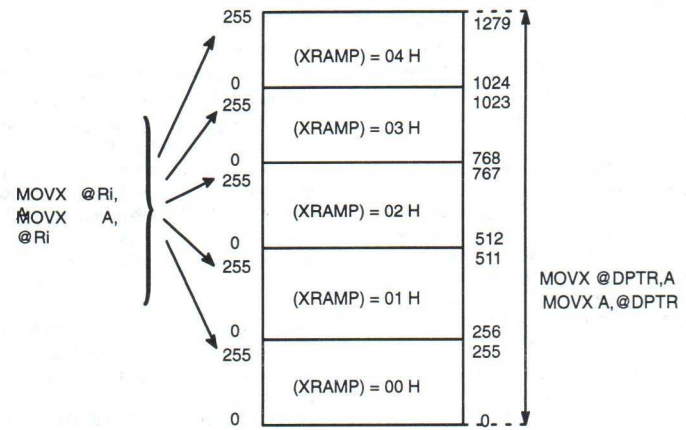



**Figure 5. Indirect addressing of AUX–RAM (1280 Bytes), ARD bit in PCON = 0.**

#### 6.2.2.1 AUX–RAM Page Register XRAMP

The AUX–RAM Page Register is used to select one of five 256 bytes pages of the internal 1280 bytes AUX–RAM for MOVX–accesses via R0 or R1. Its reset value is (XXXXX000).

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **XRAMP (FAH)** | x | x | x | x | x | XRAMP2 | XRAMP1 | XRAMP0 |

x: undefined during read, a write operation must write "0" to these locations

**Figure 6. AUX–RAM page register.**

**Table 3. Description of XRAMP bits**

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| XRAMP.3–7 | XRAMPx | reserved for future use |
| XRAMP.2 | XRAMP2 | AUX–RAM page select bit 2 |
| XRAMP.1 | XRAMP1 | AUX–RAM page select bit 1 |
| XRAMP.0 | XRAMP0 | AUX–RAM page select bit 0 |

**Table 4. Memory locations for all possible MOVX–accesses**

| ARD[1] | XRAMP2 | XRAMP1 | XRAMP0 | MOVX @Ri,A and MOVX A,@Ri instructions access: |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | AUX–RAM locations 0 .. 255 (reset condition) |
| 0 | 0 | 0 | 1 | AUX–RAM locations 256 .. 511 |
| 0 | 0 | 1 | 0 | AUX–RAM locations 512 .. 767 |
| 0 | 0 | 1 | 1 | AUX–RAM locations 768 ... 1023 |
| 0 | 1 | 0 | 0 | AUX–RAM locations 1024 .. 1279 |
| 0 | 1 | 0 | 1 | no valid memory access; reserved for future use |
| 0 | 1 | 1 | X | no valid memory access; reserved for future use |
| 1 | X | X | X | External RAM locations 0 .. 255 |
| | | | | **MOVX @DPTR,A and MOVX A,@DPTR instructions access:** |
| 0 | X | X | X | AUX–RAM locations 0 .. 1279 (reset condition)<br>External RAM locations 1280 .. 65535 |
| 1 | X | X | X | External RAM locations 0 .. 65535 |

**NOTE:**
1. ARD (AUX–RAM Disable) is a bit in the Special Function Register PCON

**Table 5. Special Function Register Memory Map and Reset Values**

| | High Nibble of SFR Address | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| LOW | 8 | 9 | A | B | C | D | E | F |
| 0 | P0 %<br>11111111 | P1 %<br>11111111 | P2 %<br>11111111 | P3 %<br>11111111 | P4 %<br>11111111 | PSW %<br>00000000 | ACC %<br>00000000 | B %<br>00000000 |
| 1 | SP 00000111 | | | | | | | |
| 2 | DPL<br>00000000 | | | | | | | |
| 3 | DPH<br>00000000 | | | | | | | |
| 4 | | | | | | | | |
| 5 | | | | | | | | |
| 6 | ADRSL0 #<br>XXXXXXXX | ADRSL1 #<br>XXXXXXXX | ADRSL2 #<br>XXXXXXXX | ADRSL3 #<br>XXXXXXXX | ADRSL4 #<br>XXXXXXXX | ADRSL5 #<br>XXXXXXXX | ADRSL6 #<br>XXXXXXXX | ADRSL7 #<br>XXXXXXXX |
| 7 | PCON<br>00000000 | | | | P5 #<br>XXXXXXXX | ADCON<br>00000000 | ADPSS<br>00000000 | ADRSH #<br>000000XX |
| 8 | TCON %<br>00000000 | S0CON %<br>00000000 | IEN0 %<br>00000000 | iP0 %<br>X0000000 | TM2IR %<br>00000000 | S1CON %<br>00000000 | IEN1 %<br>00000000 | IP1 %<br>00000000 |
| 9 | TMOD<br>00000000 | S0BUF<br>XXXXXXXX | CML0<br>00000000 | | CMH0<br>00000000 | S1STA #<br>11111000 | | PLLCON<br>00001101 |
| A | TL0<br>00000000 | | CML1<br>00000000 | | CMH1<br>00000000 | S1DAT<br>00000000 | TM2CON<br>00000000 | XRAMP<br>XXXXX000 |
| B | TL1<br>00000000 | | CML2<br>00000000 | | CMH2<br>00000000 | S1ADR<br>00000000 | CTCON<br>00000000 | FMCON *<br>000X0000 |
| C | TH0<br>00000000 | | CTL0 #<br>XXXXXXXX | | CTH0 #<br>XXXXXXXX | | TML2 #<br>00000000 | PWM0<br>00000000 |
| D | TH1<br>00000000 | | CTL1 #<br>XXXXXXXX | | CTH1 #<br>XXXXXXXX | | TMH2 #<br>00000000 | PWM1<br>00000000 |
| E | | | CTL2 #<br>XXXXXXXX | | CTH2 #<br>XXXXXXXX | | STE<br>11000000 | PWMP<br>00000000 |
| F | | | CTL3 #<br>XXXXXXXX | | CTH3 #<br>XXXXXXXX | | RTE<br>00000000 | T3<br>00000000 |

% = Bit addressable register
# = Read only register
X = Undefined
* = FMCON only in P89CE559

## 6.3 Addressing

The P8xCE559 has five methods for addressing:

- Register
- Direct
- Register–Indirect
- Immediate
- Base–Register plus Index–Register–Indirect

The first three methods can be used for addressing destination operands. Most instructions have a "destination/source" field that specifies the data type, addressing methods and operands involved. For operations other than MOVs, the destination operand is also a source operand.

Access to memory addresses is as follows:

- Register in one of the four register banks through Register, Direct or Register–Indirect addressing
- 1536 bytes of internal RAM through Direct or Register–Indirect addressing. Bytes 0–127 of internal RAM may be addressed directly/indirectly. Bytes 128–255 of internal RAM share their address location with the SFRs and so may only be addressed indirectly as data RAM. Bytes 0–1279 of AUX–RAM can only be addressed indirectly via MOVX.
- Special Function Register through direct addressing at address locations 128–255 (see Figure 8).
- External data memory through Register–Indirect addressing
- Program memory look–up tables through Base– Register plus Index–Register–Indirect addressing

| BYTE ADDRESS (HEX) | BIT ADDRESS (HEX) (MSB) | | | | | | | (LSB) | BYTE ADDRESS (DECIMAL) |
|---|---|---|---|---|---|---|---|---|---|
| FFH | | | | | | | | | 255 |
| 2FH | 7F | 7E | 7D | 7C | 7B | 7A | 79 | 78 | 47 |
| 2EH | 77 | 76 | 75 | 74 | 73 | 72 | 71 | 70 | 46 |
| 2DH | 6F | 6E | 6D | 6C | 6B | 6A | 69 | 68 | 45 |
| 2CH | 67 | 66 | 65 | 64 | 63 | 62 | 61 | 60 | 44 |
| 2BH | 5F | 5E | 5D | 5C | 5B | 5A | 59 | 58 | 43 |
| 2AH | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 42 |
| 29H | 4F | 4E | 4D | 4C | 4B | 4A | 49 | 48 | 41 |
| 28H | 47 | 46 | 45 | 44 | 43 | 42 | 41 | 40 | 40 |
| 27H | 3F | 3E | 3D | 3C | 3B | 3A | 39 | 38 | 39 |
| 26H | 37 | 36 | 35 | 34 | 33 | 32 | 31 | 30 | 38 |
| 25H | 2F | 2E | 2D | 2C | 2B | 2A | 29 | 28 | 37 |
| 24H | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 36 |
| 23H | 1F | 1E | 1D | 1C | 1B | 1A | 19 | 18 | 35 |
| 22H | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 34 |
| 21H | 0F | 0E | 0D | 0C | 0B | 0A | 09 | 08 | 33 |
| 20H | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 | 32 |
| 1FH | Bank 3 | | | | | | | | 31 |
| 18H | | | | | | | | | 24 |
| 17H | Bank 2 | | | | | | | | 23 |
| 10H | | | | | | | | | 16 |
| 0FH | Bank 1 | | | | | | | | 15 |
| 08H | | | | | | | | | 8 |
| 07H | Bank 0 | | | | | | | | 7 |
| 00H | | | | | | | | | 0 |

**Figure 7. RAM bit addresses.**

| DIRECT BYTE ADDRESS (HEX) | BIT ADDRESS (HEX) (MSB) | | | | | | | (LSB) | REGISTER MNEMONIC |
|---|---|---|---|---|---|---|---|---|---|
| FFH | PT2 | PCM2 | PCM1 | PCM0 | PCT3 | PCT2 | PCT1 | PCT0 | |
| F8H | FF | FE | FD | FC | FB | FA | F9 | F8 | IP1 |
| F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | B |
| E8H | ET2 | ECM2 | ECM1 | ECM0 | ECT3 | ECT2 | ECT1 | ECT0 | IEN1 |
| E0H | EF | EE | ED | EC | EB | EA | E9 | E8 | ACC |
| D8H | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 | S1CON |
| D0H | DF | DE | DD | DC | DB | DA | D9 | D8 | PSW |
| C8H | CY | AC | F0 | RS1 | RS0 | OV | F1 | P | TM2IR |
| C0H | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | P4 |
| B8H | T2OV | CMI2 | CMI1 | CMI0 | CTI3 | CTI2 | CTI1 | CTI0 | IP0 |
| B0H | CF | CE | CD | CC | CB | CA | C9 | C8 | P3 |
| A8H | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 | IEN0 |
| A0H | – | PAD | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 | P2 |
| 98H | BF | BE | BD | BC | BB | BA | B9 | B8 | S0CON |
| 90H | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | P1 |
| 88H | EA | EAD | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 | TCON |
| 80H | AF | AE | AD | AC | AB | AA | A9 | A8 | P0 |
| | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | |
| | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | |
| | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |

**Figure 8. Special Function Register bit addresses.**

## 6.4 I/O Facilities

The P8xCE559 has six 8–bit ports. Ports 0 to 3 are the same as in the 80C51, with the exception of the additional functions of Port 1. The parallel I/O function of Port 4 is equal to that of Ports 1, 2 and 3. Port 5 has a parallel input port function, but has no function as an output port.

The SDA and SCL lines serve the serial port SI01 ($I^2C$). Because the $I^2C$–bus may be active while the device is disconnected from $V_{DD}$, these pins, are provided with open drain drivers.

Ports 0, 1, 2, 3, 4 and 5 perform the following alternative functions:

Port 0 : provides the multiplexed low–order address and data bus used for expanding the P8xCE559 with standard memories and peripherals.

Port 1 : Port 1 is used for a number of special functions:

4 capture inputs (or external interrupt request inputs if capture information is not utilized)
- external counter input
- external counter reset input

Port 2 : provides the high–order address bus when the P8xCE559 is expanded with external Program Memory and/or external Data Memory.

Port 3 : pins can be configured individually to provide:
- external interrupt request inputs
- counter inputs
- receiver input and transmitter output of seri port SIO 0 (UART)
- control signals to read and write external Data Memory

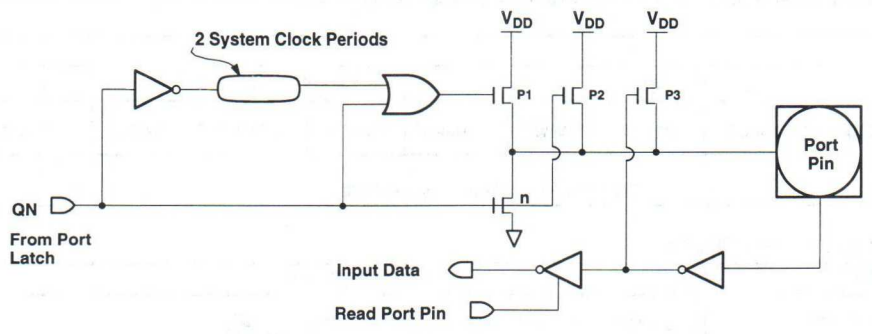
Port 4 : can be configured to provide signals indicating a match between timer counter T2 and its compare registers.

Port 5 : may be used in conjunction with the ADC interface.Unused analog inputs can be used as digital inputs. As Port 5 lines may be used as inputs to the ADC, these digital inputs have an inherent hysteresis to prevent the input logic from drawing too much current from the power lines when driven by analog signals. Channel to channel crosstalk should be taken into consideration when both digital and analog signals are simultaneously input to Port 5 (see DC characteristics).

All ports are bidirectional with the exception of Port 5 which is an input port.

Pins of which the alternative function is not used may be used as normal bidirectional I/Os.

The generation or use of a Port 1, Port 3 or Port 4 pin as an alternative function is carried out automatically by the P8xCE559 provided the associated Special Function Register bit is set HIGH. The pull–up arrangements of Ports 1 – 4 are shown in Figure 9.




P1 is turned on for 2 system clock periods after QN makes a 1–to–0 transition.
During this time, P1 also turns on P3 through the inverter to form an additional pull up.

**Figure 9. I/O buffers in the P8xCE559 (Ports 1, 2, 3 and 4).**

## 6.5 Pulse Width Modulated Outputs

The P8xCE559 contains two pulse width modulated output channels (see Figure 13). These channels generate pulses of programmable length and interval. The repetition frequency is defined by an 8-bit prescaler PWMP, which supplies the clock for the counter. The prescaler and counter are common to both PWM channels. The 8-bit counter counts module 255, i.e., from 0 to 254 inclusive. The value of the 8-bit counter is compared to the contents of two registers: PWM0 and PWM1. Provided the contents of either of these registers is greater than the counter value, the corresponding $\overline{PWM0}$ or $\overline{PWM1}$ output is set LOW. If the contents of these registers are equal to, or less than the counter value, the output will be HIGH. The pulse-width-ratio is therefore defined by the contents of the registers PWM0 and PWM1. The pulse-width-ratio is in the range of 0/255 to 255/255 and may be programmed in increments of 1/255.

Buffered PWM outputs may be used to drive DC motors. The rotation speed of the motor would be proportional to the contents of PWMn. The PWM outputs may also be configured as a dual DAC. In this application, the PWM outputs must be integrated using conventional operational amplifier circuitry. If the resulting output voltages have to be accurate, external buffers with their own analog supply should be used to buffer the PWM outputs before they are integrated. The repetition frequency fpwm, at the PWMn outputs is give by:

$$f_{PWM} = \frac{f_{CLK}}{2 \times (1 + PWMP) \times 255}$$

This gives a repetition frequency range of 123 Hz to 31.4 kHz ($f_{CLK}$ = 16 MHz). By loading the PWM registers with either 00H or FFH, the PWM channels will output a constant HIGH or LOW level, respectively. Since the 8-bit counter counts modulo 255, it can never actually reach the value of the PWM registers when they are loaded with FFH.

When a compare register (PWM0 or PWM1) is loaded with a new value, the associated output is updated immediately. It does not have to wait until the end of the current counter period. Both $\overline{PWMn}$ output pins are driven by push-pull drivers. These pins are not used for any other purpose.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **PWMP (FEH)** | PWMP.7 | PWMP.6 | PWMP.5 | PWMP.4 | PWMP.3 | PWMP.2 | PWMP.1 | PWMP.0 |

**Figure 10. Prescaler frequency control register PWMP.**

**Table 6. Description of PWMP bits**

| BIT | FUNCTION |
|---|---|
| PWMP.0 to 7 | Prescaler division factor = (PWMP) + 1 |

Reading PWMP gives the current reload value. The actual count of the prescaler cannot be read.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM0 (FCH)** | PWM0.7 | PWM0.6 | PWM0.5 | PWM0.4 | PWM0.3 | PWM0.2 | PWM0.1 | PWM0.0 |

**Figure 11. Pulse width register PWM0.**

**Table 7. Description of PWM0 bits**

| BIT | FUNCTION |
|---|---|
| PWM0.0 to 7 | LOW/HIGH ration of $\overline{PWM0}$ signal = $\frac{(PWM0)}{255 - (PWM0)}$ |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM1 (FDH)** | PWM1.7 | PWM1.6 | PWM1.5 | PWM1.4 | PWM1.3 | PWM1.2 | PWM1.1 | PWM1.0 |

**Figure 12. Pulse width register PWM1.**

**Table 8. Description of PWM1 bits**

| BIT | FUNCTION |
|---|---|
| PWM1.0 to 7 | LOW/HIGH ration of $\overline{PWM1}$ signal = $\frac{(PWM1)}{255 - (PWM1)}$ |

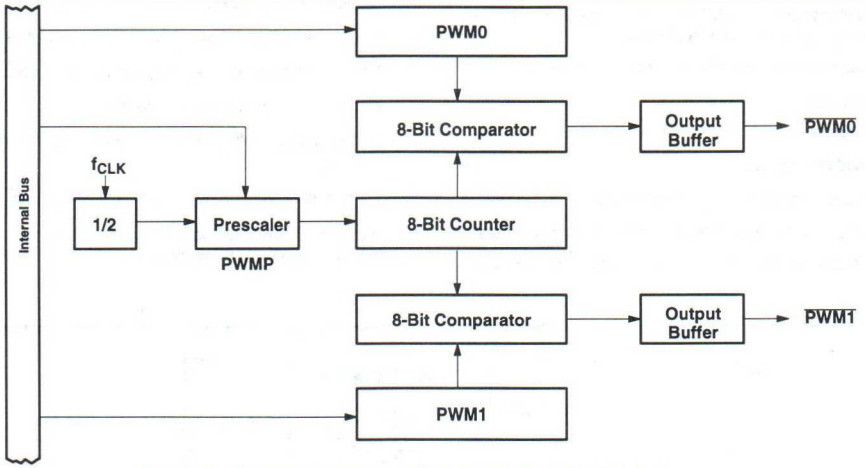



**Figure 13. Functional Diagram of Pulse Width Modulated Outputs.**

## 6.6 Analog/Digital Converter (ADC)

The P8xCE559 A/D Converter is a 10–bit, successive approximation ADC with 8 multiplexed analog input channels. It additionally contains a high input impedance comparator, a DAC built with 1024 series resistors and analog switches, registers and control logic.

Input voltage range is from $AV_{ref-}$ (typical 0V) to $AV_{ref+}$ (typical +5V). A set of 8 buffer registers (10–bit) store the conversion results of the proper analog input channel each.

11 Special Function Registers (SFR) perform the user software interface to the ADC: a control SFR (ADCON), an analog port scan–select SFR (ADPSS), 8 input channel related conversion result SFR with the 8 lower result bits (ADRSL0...ADRSL7), one common result SFR for the upper 2 result bits (ADRSH). An extra SFR (P5) allows for reading digital input port data as an alternative function of the 8 analog input pins.

In order to have a minimum of ADC service overhead in the microcontroller program, the ADC is able to operate autonomously within its user configurable autoscan function.

The functional diagram of the ADC is shown in Figure 14.

**Feature Overview:**

- 10–bit resolution.
- 8 multiplexed analog inputs.
- Programmable autoscan of the analog inputs.
- Bit oriented 8–bit scan–select register to select analog inputs.
- Continuous scan or one time scan configurable from 1 to 8 analog inputs.
- Start of a conversion by software or with an external signal.
- Eight 10–bit buffer registers, one register for each analog input channel.
- Interrupt request after one channel scan loop.
- Programmable prescaler (dividing by 2, 4, 6, 8) to adapt to different system clock frequencies.
- Conversion time for one A/D conversion: 15 µs ... 50 µs
- Differential non–linearity : DLe ±1 LSB.
- Integral non–linearity : ILe ±2 LSB.
- Offset error : OSe ±2LSB.
- Gain error : Ge ±0.4 %.
- Absolute voltage error : Ae ±3 LSB.
- Channel to channel matching : Mctc ±1LSB.
- Crosstalk between analog inputs : Ct < –60dB. @100 kHz.
- Monotonic and no missing codes.
- Separated analog ($AV_{DD}$, $AV_{SS}$) and digital ($V_{DD}$, $V_{SS}$) supply voltages.
- Reference voltage at two special pins : $AV_{REF-}$ and $AV_{REF+}$.

For further information on the ADC characteristics, refer to the "DC CHARACTERISTICS" section.

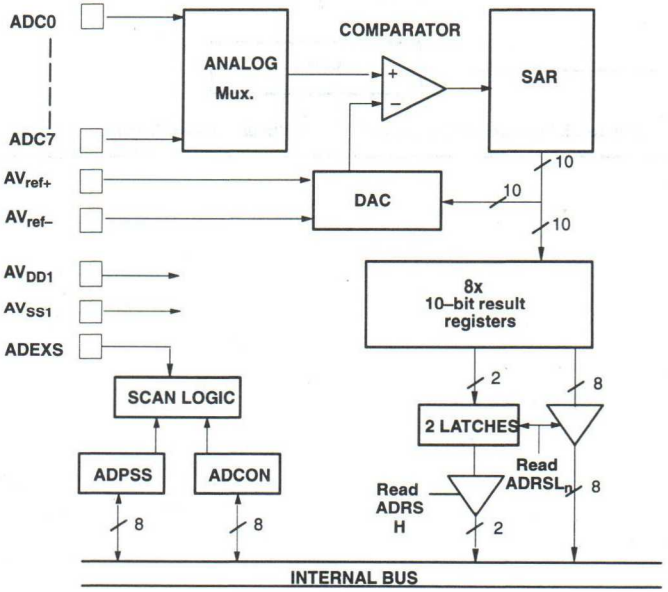



**Figure 14. Functional diagram of AD converter.**

### 6.6.1 Functional Description

**Table 9. A/D Special Function Registers**

| SYMBOL | NAME | ACCESS |
|---|---|---|
| ADCON | A/D control register | read/write |
| ADPSS | Analog port scan–select register | read/write |
| ADRSLn | 8 A/D result registers, the 8 lower bits (n: 0...7) | read only |
| ADRSH | A/D result register, the 2 higher bits | read only |
| P5 | Digital input port (shared with analog inputs) | read only |

**A/D Control Register ADCON**

The Special Function Register ADCON contains control and status bits for the A/D Converter peripheral block. The reset value of ADCON is (00000000). Its hardware address is D7H. ADCON is not bit addressable.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **ADCON (D7H)** | ADPR1 | ADPR0 | ADPOS | ADINT | ADSST | ADCSA | ADSRE | ADSFE |

**Figure 15. ADC control register.**

**Table 10. Description of ADCON bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| ADCON.7<br>ADCON.6 | ADPR1<br>ADPR0 | Control bit for the prescaler.<br>Control bit for the prescaler.<br>ADPR1=0 ADPR0=0 Prescaler divides by 2 (default by reset)<br>ADPR1=0 ADPR0=1 Prescaler divides by 4<br>ADPR1=1 ADPR0=0 Prescaler divides by 6<br>ADPR1=1 ADPR0=1 Prescaler divides by 8 |
| ADCON.5 | ADPOS | ADPOS is reserved for future use. Must be '0' if ADCON is written. |
| ADCON.4 | ADINT | ADC interrupt flag. This flag is set when all selected analog inputs are converted, as well in continuous scan as in one–time scan mode. An interrupt is invoked if this interrupt is enabled. ADINT must be cleared by software. It cannot be set by software. |
| ADCON.3 | ADSST | ADC start and status. Setting this bit by software or by hardware (via ADEXS input) starts the A/D conversion of the selected analog inputs. ADSST stays a 'one' in continuous scan mode. In one–time scan mode, ADSST is cleared by hardware when the last selected analog input channel has been converted. As long as ADSST is '1', new start commands to the ADC–block are ignored.<br>An A/D conversion in progress is aborted if ADSST is cleared by software. |
| ADCON.2 | ADCSA | 1 = Continuous scan of the selected analog inputs after a start of an A/D conversion.<br>0 = One–time scan of the selected analog inputs after a start of an A/D conversion. |
| ADCON.1 | ADSRE | 1 = A rising edge at input ADEXS will start the A/D conversion and generate a capture signal.<br>0 = A rising edge at input ADEXS has no effect. |
| ADCON.0 | ADSFE | 1 = A falling edge at input ADEXS will start the A/D conversion and generate a capture signal.<br>0 = A falling edge at input ADEXS has no effect. |

**A/D Input Port Scan-Select Register ADPSS**
The Special Function Register ADPSS contains control bits to select the analog input channel(s) to be scanned for A/D conversion. The reset value of ADPSS is (00000000). Its hardware address is E7H. ADPSS is not bit addressable.

If all bits are '0' then no A/D conversion can be started. If ADPSS is written while an A/D conversion is in progress (ADSST in the ADCON register is '1') then the autoscan loop with the previous selected analog inputs is completed first. The next autoscan loop is performed with the new selected analog inputs.

**A/D Result Registers ADRSLn and ADRSH:**
The binary result code of A/D conversions is accessed by these Special Function Registers. The result SFR are read only registers. The read value after reset is indeterminate. Their data are not affected by chip reset. They are not bit addressable.

There are 8 Special Function Registers ADRSLn (ADRSL0...ADRSL7) – A/D Result Low byte - and one general SFR ADRSH - A/D Result High byte - . Each of ADRSLn is associated with the coincidently indexed analog input channel ADCn (ADC0/P5.0...ADC7/P5.7). Reading an ADRSLn register by software copies at the same time the two highest bits of the 10-bit conversion result into two latches, thus preserving them until the next read of any ADRSLn register. These two latches form bit positions 0 and 1 of SFR ADRSH, the upper 6 bits of ADRSH are always read as '0'.

Thus it is ensured to get the 10-bit result of the same single A/D conversion by reading any register ADRSLn first and after it the register ADRSH.

**Digital Input Port Register P5**
Port 5 Special Function Register P5 always represents the binary value of the logic level at input pins P5.0/ADC0...P5.7/ADC7. P5 is not affected by chip reset. P5 is a read only register. Its hardware address is C7H. P5 is not bit addressable.

Reading Special Function Register P5 does not affect A/D conversions. But it is recommended to use the digital input port function of the hardware Port 5 only as an alternative to analog input voltage conversions. Simultaneous mixed operation is discouraged for the sake of A/D conversion result reliability and accuracy.

For further information on Port 5, refer to the "I/O facilities" section.

For further information on A/D Special Function Registers, refer to the "Internal Data Memory" section.

**Reset**
After a RESET of the microcontroller the ADCON and ADPSS register bits are initialized to zero. Registers ADRSLn and ADRSH are not initialized by a RESET.

**Idle and Power-down Mode**
The A/D Converter is active only when the microcontroller is in normal operating mode. If the Idle or Power-down Mode is activated, then the ADC is switched off and put into a power saving idle state - a conversion in progress is aborted, a previously set ADSST flag is cleared and the internal clock is halted. The conversion result registers are not affected.

The interrupt flag ADINT will not be set by activation of Idle or Power-down Mode. A previously set flag ADINT will not be cleared by the hardware. (Note: ADINT cannot be cleared by hardware at all, except for a RESET - it must be cleared by the user software.)

After a wakeup from Idle or Power-down Mode a set flag ADINT indicates that at least one autoscan loop was finished completely before the microcontroller was put into the respective power reduction mode and it indicates that the stored result data may be fetched now - if desired.

For further information on Idle and Power-down Mode, refer to the "Power reduction modes" section.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **ADPSS (E7H)** | ADPSS7 | ADPSS6 | ADPSS5 | ADPSS4 | ADPSS3 | ADPSS2 | ADPSS1 | ADPSS0 |

ADPSS7–0 For each individual bit position: 0 = The corresponding analog input is skipped in the auto-scan loop.
1 = The corresponding analog input is included in the auto-scan loop.

**Figure 16. A/D input port scan-select register.**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **ADRSLn** | ADRSn.7 | ADRSn.6 | ADRSn.5 | ADRSn.4 | ADRSn.3 | ADRSn.2 | ADRSn.1 | ADRSn.0 |

(n: 0...7)

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **ADRSH** | 0 | 0 | 0 | 0 | 0 | 0 | ADRSn.9 | ADRSn.8 |

**Figure 17. A/D Result Registers.**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **P5 (C7H)** | P5.7 | P5.6 | P5.5 | P5.4 | P5.3 | P5.2 | P5.1 | P5.0 |

**Figure 18. Digital input port register P5.**

**Timing**
A programmable prescaler is controlled by the bits ADPR1 and ADPR0 in register ADCON to adapt the conversion time for different microcontroller clock frequencies.

Table 11 shows conversion times (tconv) for one A/D conversion at some convenient system clock frequencies (fclk) and ADC prescaler divisors (m), which are user selectable by the bits ADCON.7/ADPR1 and ADCON.6/ADPR0.

For conversion times outside the limits for tconv the specified ADC characteristics are not guaranteed; (prohibited conversion times are put in brackets):

**Table 11. Conversion time configuration examples (tconv/μs)**

| m | $f_{CLK}$ | | | |
|---|---|---|---|---|
| | 6 MHz | 8 MHz | 12 MHz | 16 MHz |
| 2 | 26 | 19.5 | [13] | [9.75] |
| 4 | 50 | 37.5 | 25 | 18.75 |
| 6 | [74] | [55.5] | 37 | 27.75 |
| 8 | [98] | [73.5] | 49 | 36.75 |

Conversion time tconv = (6 m + 1) machine cycles

A conversion time tconv consists of one sample time period (which equals two bit conversion times), 10 bit conversion time periods and one machine cycle to store the result.

After result storage an extra initializing time period follows to select the next analog input channel (according to the contents of SFR ADPSS), before the input signal is sampled.

Thus the time period between two adjacent conversions within an autoscan loop is larger than the pure time tconv. This autoscan cycle time is (7 m) machine cycles.

At the start of an autoscan conversion the time between writing to SFR ADCON and the first analog input signal sampling depends on the current prescaler value (m) and the relative time offset between this write operation and the internal (divided) ADC clock. This gives a variation range for the A/D conversion start time of ( m / 2 ) machine cycles.

### 6.6.2 Configuration and Operation

Every A/D conversion is an autoscan conversion. The two user selectable general operation modes are continuous scan and one–time scan mode.

The desired analog input port channel/s for conversion is/are selected by programming A/D input port scan–select bits in SFR ADPSS. An analog input channel is included in the autoscan loop if the corresponding bit in ADPSS is 1, a channel is skipped if the corresponding bit in ADPSS is 0.

An autoscan is always started according to the lowest bit position of ADPSS that contains a 1.

An autoscan conversion is started by setting the flag ADSST in register ADCON either by software or by an external start signal at input pin ADEXS, if enabled. Either no edge (external start totally disabled), a rising edge or/and a falling edge of ADEXS is selectable for external conversion start by the bits ADSRE and ADSFE in register ADCON.

After completion of an A/D conversion the 10–bit result is stored in the corresponding 10–bit buffer register. Then the next analog input is selected according to the next higher set bit position in ADPSS, converted and stored, and so on. When the result of the last conversion of this autoscan loop is stored, flag ADCON.4/ADINT, the ADC interrupt flag, is set. It is not cleared by interrupt hardware – it must be cleared by software.

In continuous scan mode (ADCON.2/ADCSA=1) the ADC start and status flag ADCON.3/ADSST retains the set state and the autoscan loop restarts from the beginning. In one–time scan mode (ADCSA=0) conversions stop after the last selected analog input was converted, ADINT is set and ADSST is cleared automatically.

ADSST cannot be set (neither externally nor by software) as long as ADINT=1, i.e. as long as ADINT is set, a new conversion start – by setting flag ADSST – is inhibited; actually it is only delayed until ADINT is cleared.

(If a '1' is written to ADSST while ADINT=1, this new value is internally latched and preserved, not setting ADSST until ADCON.4/ADINT=0. In this state, a read of SFR ADCON will display ADCON.3/ADSST=0, because always the effective ADC status is read.)

Note that under software control the analog inputs can also be converted in arbitrary order, when one–time scan mode is selected and in SFR ADPSS only one bit is set at a time. In this case ADINT is set and ADSST is cleared after every conversion.

### 6.6.3 Resolution and Characteristics

The ADC system has its own analog supply pins $AV_{DD}$ and $AV_{SS}$. It is referenced by two special reference voltage input pins sourcing the resistance ladder of the DAC: $AV_{ref+}$ and $AV_{ref-}$. The voltage between $AV_{REF+}$ and $AV_{REF-}$ defines the full–scale range. Due to the 10–bit resolution the full scale range is divided into 1024 unit steps. The unit step voltage is 1 LSB, which is typically 5 mV ($AV_{ref+}$ = 5.12 V, $AV_{ref-}$ = 0 V = $AV_{SS}$).

The DAC's resistance ladder has 1023 equally spaced taps, separated by a unit resistance 'R'. The first tap is located 0.5 x R above $AV_{ref-}$, the last tap is located 1.5 x R below $AV_{ref+}$. This results in a total ladder resistance of 1024 x R. This structure ensures that the DAC is monotonic and results in a symmetrical quantization error. For input voltages between $AV_{ref-}$ and ($AV_{ref-}$ + 1/2 LSB) the 10–bit conversion result code will be 00 0000 0000 B = 000H = 0D. For input voltages between ($AV_{ref+}$ – 3/2 LSB) and $AV_{ref+}$ the 10–bit conversion result code will be 11 1111 1111 B = 3FFH = 1023D.

The result code corresponding to an analog input voltage ($AV_{in}$) can be calculated from the formula:

$$\text{Result Code} = 1024 \times \frac{AV_{IN} - AV_{REF-}}{AV_{REF+} - AV_{REF-}}$$

The analog input voltage should be stable when it is sampled for conversion. At any times the input voltage slew rate must be less than 10 V/ms (5 V conversion range) in order to prevent an undefined result.

This maximum input voltage slew rate can be ensured by an RC low pass filter with R = 2k2 and C = 100 nF. The capacitor between analog input pin and analog ground pin shall be placed close to the pins in order to have maximum effect in minimizing input noise coupling.

## 6.7 Timer / Counters

The P8xCE559 contains three 16-bit timer/event counters: Timer 0, Timer 1 and Timer T2 and one 8-bit timer, T3. Timer 0 and Timer 1 may be programmed to carry out the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests

### 6.7.1 Timer 0 and Timer 1

Timers 0 and 1 each have a control bit in SFR TMOD that selects the timer or counter function of the corresponding timer.

In the timer function, the register is incremented every machine cycle. Thus, one can think of it as counting machine cycles. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

In the counter function, the register is incremented in response to a 1-to-0 transition at the corresponding external input pin, T0 or T1. In this function, the external input is sampled during S5P2 of every machine cycle. When the samples show a HIGH in one cycle and a LOW in the next cycle, the counter is incremented. Thus, it takes two machine cycles (24 oscillator periods) to recognize a 1-to-0 transition. There are no restrictions on the duty cycle of the external input signal, but to insure that a given level is sampled at least once before it changes, it should be held for at least one full machine cycle.

Timer 0 and Timer 1 can be programmed independently to operate in one of four modes:

- **Mode 0:** 8-bit timer or 8-bit counter each with divide-by-32 prescaler
- **Mode 1:** 16-bit time-interval or event counter
- **Mode 2:** 8-bit time-interval or event counter with automatic reload upon overflow
- **Mode 3:** -Timer 0: one 8-bit time-interval or event counter and one 8-bit time-interval counter
  -Timer 1: stopped

When Timer 0 is in Mode 3, Timer 1 can be programmed to operate in Modes 0, 1 or 2 but cannot set an interrupt request flag or generate an interrupt. However the overflow from Timer 1 can be used to pulse the serial port baud-rate generator.

With a 16 MHz crystal, the counting frequency of these timer/counters is as follows:

- In the timer function, the timer is incremented at a frequency of 1.33 MHz - a division by 12 of the system clock frequency
- 0 Hz to an upper limit of 0.66 MHz (1/24 of the system clock frequency) when programmed for external inputs

Both internal and external inputs can be gated to the counter by a second external source for directly measuring pulse durations.

When configured as a counter, the register is incremented on every falling edge on the corresponding input pin, T0 or T1. The incremented register value can be read earliest during the second machine cycle after that one, during which the incrementing pulse occurred.

The counters are started and stopped under software control. Each one sets its interrupt request flag when it overflows from all HIGHs to all LOWs (or automatic reload value), with the exception of mode 3 as previously described.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **TMOD (89H)** | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 |
| | | | Timer 1 | | | | Timer 0 | |

**Figure 19. Timer/Counter mode control (TMOD) register.**

**Table 12. Description of TMOD bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| Gate | TMOD.7<br>TMOD.3 | Gating control when set. Timer/Counter "x" is enabled only while "$\overline{INTx}$" pin is high and "TRx" control pin is set. When cleared Timer "x" is enabled whenever "TRx" control bit is set. |
| C/T | TMOD.6<br>TMOD.2 | Timer or Counter Selector cleared for Timer operation (input from internal system clock). Set for Counter operation (input from "Tx" input pin). |
| M1<br><br>M0 | TMOD.5<br>TMOD.1<br>TMOD.4<br>TMOD.0 | Timer 0, Timer 1 mode select see Table 13. |

**Table 13. Timer 0 / Timer 1 operation select**

| M1 | M0 | OPERATING |
|---|---|---|
| 0 | 0 | 8048 Timer "TLx" serves as 5-bit prescaler. |
| 0 | 1 | 16-bit Timer/Counter "THx" and "TLx" are cascaded; there is no prescaler. |
| 1 | 0 | 8-bit auto-reload Timer/Counter "THx" holds a value which is to be reloaded into "TLx" each time it overflows. |
| 1 | 1 | (Timer 0) TL0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits. TH0 is an 8-bit timer only controlled by Timer 1 control bits. |
| 1 | 1 | (Timer 1) Timer/Counter 1 stopped. |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **TCON (88H)** | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |

**Figure 20. Timer/Counter mode control (TCON) register.**

**Table 14. Description of TCON bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| TF1 | TCON.7 | Timer 1 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine. |
| TR1 | TCON.6 | Timer 1 run control bit. Set/cleared by software to turn Timer/Counter on/off. |
| TF0 | TCON.5 | Timer 0 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine. |
| TR0 | TCON.4 | Timer 0 run control bit. Set/cleared by software to turn Timer/Counter on/off. |
| IE1 | TCON.3 | Interrupt 1 edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed. |
| IT1 | TCON.2 | Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts. |
| IE0 | TCON.1 | Interrupt 0 edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed. |
| IT0 | TCON.0 | Interrupt 0 type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts. |

### 6.7.2 Timer T2

Timer T2 is a 16 bit timer/counter which has capture and compare facilities. The operational diagram is shown in Figure 21.

The 16 bit timer/counter is clocked via a prescaler with a programmable division factor of 1, 2, 4 or 8. The input of the prescaler is clocked with 1/12 of the clock frequency, or by an external source connected to the T2 input, or it is switched off. The maximum repetition rate of the external clock source is $f_{CLK}/12$, twice that of Timer 0 and Timer 1. The prescaler is incremented on a rising edge. It is cleared if its division factor or its input source is changed, or if the timer/counter is reset (see also Figure 22: TM2CON). T2 is readable 'on the fly', without any extra read latches; this means that software precautions have to be taken against misinterpretation at overflow from least to most significant byte while T2 is being read. T2 is not loadable and is reset by the RST signal or at the positive edge of the input signal RT2, if enabled. In the Idle or Power-down Mode the timer/counter and prescaler are reset and halted.

T2 is connected to four 16-bit Capture Registers: CT0, CT1, CT2 and CT3. A rising or falling edge on the inputs CT0I, CT1I, CT2I or CT3I (alternative function of Port 1) results in loading the contents of T2 into the respective Capture Registers and an interrupt request.

Using the Capture Register CTCON (see Figure 23), these inputs may invoke capture and interrupt request on a positive, a negative edge or on both edges. If neither a positive nor a negative edge is selected for capture input, no capture or interrupt request can be generated by this input.

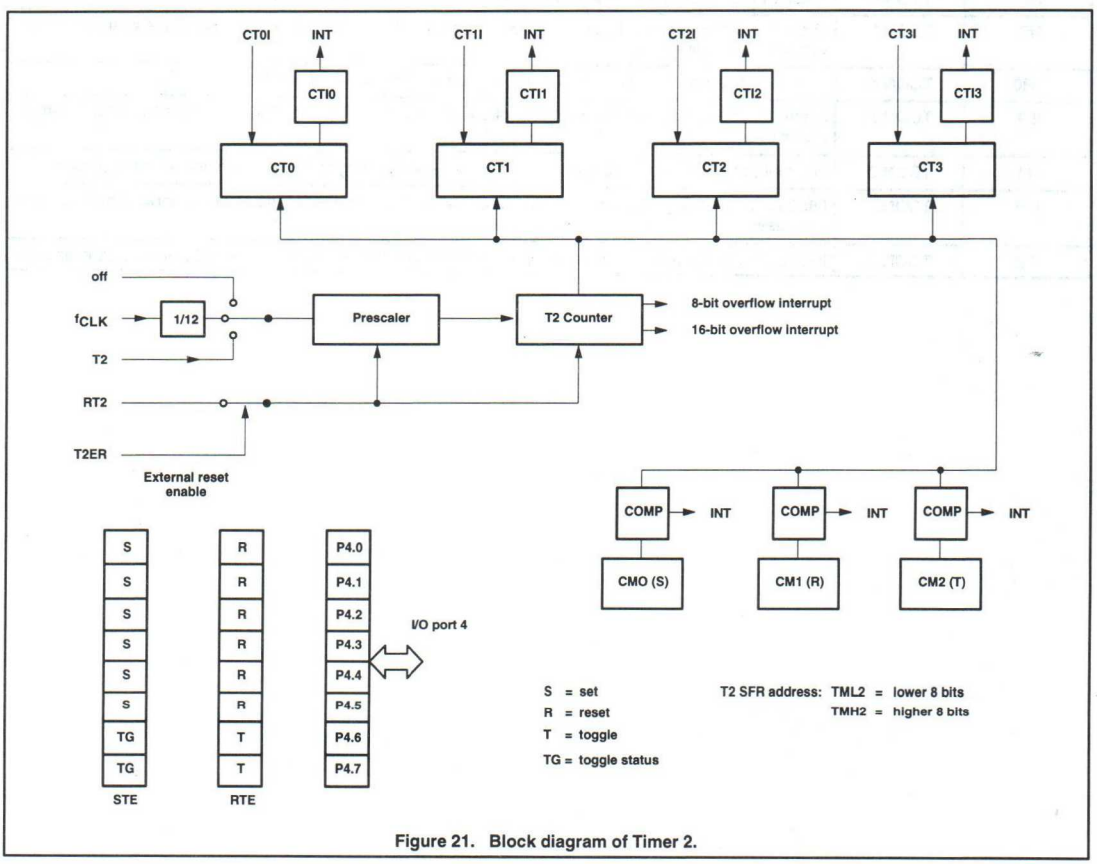



**Figure 21. Block diagram of Timer 2.**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **TM2CON (EAH)** | T2IS1 | T2IS0 | T2ER | T2BO | T2P1 | T2P0 | T2MS1 | T2MS0 |

**Figure 22. T2 control register (TM2CON).**

**Table 15. Description of TM2CON bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| T2IS1 | TM2CON.7 | Timer T2 16-bit overflow interrupt select |
| T2IS0 | TM2CON.6 | Timer T2 byte overflow interrupt select |
| T2ER | TM2CON.5 | Timer T2 external reset enable. When this bit is set, Timer T2 may be reset by a rising edge on RT2 (P1.5). |
| T2BO | TM2CON.4 | Timer T2 byte overflow interrupt flag |
| T2P1 | TM2CON.3 | Timer T2 prescaler select |
| T2P0 | TM2CON.2 | |
| T2MS1 | TM2CON.1 | Timer T2 mode select |
| T2MS0 | TM2CON.0 | |

**Table 16. Timer 2 prescaler select**

| T2P1 | T2P0 | TIMER T2 CLOCK |
|---|---|---|
| 0 | 0 | Clock source |
| 0 | 1 | Clock source/2 |
| 1 | 0 | Clock source/4 |
| 1 | 1 | Clock source/8 |

**Table 17. Timer 2 mode select**

| T2MS1 | T2MS0 | MODE SELECTED |
|---|---|---|
| 0 | 0 | Timer T2 halted (off) |
| 0 | 1 | T2 clock source = $f_{CLK}/12$ |
| 1 | 0 | Test mode; do not use |
| 1 | 1 | T2 clock source = pin T2 |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **CTCON (E8H)** | CTN3 | CTP3 | CTN2 | CTP2 | CTN1 | CTP1 | CTN0 | CTP0 |

**Figure 23. Capture control register (CTCON).**

**Table 18. Description of CTCON bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| CTN3 | CTCON.7 | Capture Register 3 triggered by a falling edge on CT3I |
| CTP3 | CTCON.6 | Capture Register 3 triggered by a rising edge on CT3I |
| CTN2 | CTCON.5 | Capture Register 2 triggered by a falling edge on CT2I |
| CTP2 | CTCON.4 | Capture Register 2 triggered by a rising edge on CT2I |
| CTN1 | CTCON.3 | Capture Register 1 triggered by a falling edge on CT1I |
| CTP1 | CTCON.2 | Capture Register 1 triggered by a rising edge on CT1I |
| CTN0 | CTCON.1 | Capture Register 0 triggered by a falling edge on CT0I |
| CTP0 | CTCON.0 | Capture Register 0 triggered by a rising edge on CT0I |

The contents of the Compare Registers CM0, CM1 and CM2 are continuously compared with the counter value of Timer T2. When a match occurs, an interrupt may be invoked. A match of CM0 sets the bits 0–5 of Port 4, a CM1 match resets these bits and a CM2 match toggles bits 6 and 7 of Port 4, provided these functions are enabled by the STE respectively RTE registers. A match of CM0 and CM1 at the same time results in resetting bits 0–5 of Port 4. CM0, CM1 and CM2 are reset by the RSTIN signal.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **TM2IR (C8H)** | T2OV | CMI2 | CMI1 | CMI0 | CTI3 | CTI2 | CTI1 | CTI0 |

**Figure 24. Interrupt flag register (TM2IR).**

**Table 19. Description of TM2IR bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| T2OV | TM2IR.7 | Timer T2 16-bit overflow interrupt flag |
| CMI2 | TM2IR.6 | CM2 interrupt flag |
| CMI1 | TM2IR.5 | CM1 interrupt flag |
| CMI0 | TM2IR.4 | CM0 interrupt flag |
| CTI3 | TM2IR.3 | CT3 interrupt flag |
| CTI2 | TM2IR.2 | CT2 interrupt flag |
| CTI1 | TM2IR.1 | CT1 interrupt flag |
| CTI0 | TM2IR.0 | CT0 interrupt flag |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **STE (EEH)** | TG47 | TG46 | SP45 | SP44 | SP43 | SP42 | SP41 | SP40 |

**Figure 25. Set enable register (STE).**

**Table 20. Description of STE bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| TG47 | STE.7 | If "1" then P4.7 is reset on the next toggle, if LOW P4.7 is set on the next toggle |
| TG46 | STE.6 | If "1" then P4.6 is reset on the next toggle, if LOW P4.6 is set on the next toggle |
| SP45 | STE.5 | If "1" then P4.5 is set on a match between CM0 and Timer T2 |
| SP44 | STE.4 | If "1" then P4.4 is set on a match between CM0 and Timer T2 |
| SP43 | STE.3 | If "1" then P4.3 is set on a match between CM0 and Timer T2 |
| SP42 | STE.2 | If "1" then P4.2 is set on a match between CM0 and Timer T2 |
| SP41 | STE.1 | If "1" then P4.1 is set on a match between CM0 and Timer T2 |
| SP40 | STE.0 | If "1" then P4.0 is set on a match between CM0 and Timer T2 |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| RTE (EFH) | TP47 | TP46 | RP45 | RP44 | RP43 | RP42 | RP41 | RP40 |

**Figure 26. Reset/Toggle enable register (RTE).**

**Table 21. Description of RTE bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| TP47 | RTE.7 | If "1" then P4.7 toggles on a match between CM2 and Timer T2 |
| TP46 | RTE.6 | If "1" then P4.6 toggles on a match between CM2 and Timer T2 |
| RP45 | RTE.5 | If "1" then P4.5 toggles on a match between CM1 and Timer T2 |
| RP44 | RTE.4 | If "1" then P4.4 toggles on a match between CM1 and Timer T2 |
| RP43 | RTE.3 | If "1" then P4.3 toggles on a match between CM1 and Timer T2 |
| RP42 | RTE.2 | If "1" then P4.2 toggles on a match between CM1 and Timer T2 |
| RP41 | RTE.1 | If "1" then P4.1 toggles on a match between CM1 and Timer T2 |
| RP40 | RTE.0 | If "1" then P4.0 toggles on a match between CM1 and Timer T2 |

For more information concerning the TM2CON, CTCON, TM2IR and the STE/RTE registers see IC20 handbook, chapter "80C51 family hardware description".

Port 4 can be read and written by software without affecting the toggle, set and reset signals. At a byte overflow of the least significant byte, or at a 16-bit overflow of the timer/counter, an interrupt sharing the same interrupt vector is requested. Either one or both of these overflows can be programmed to request an interrupt.

All interrupt flags must be reset by software.

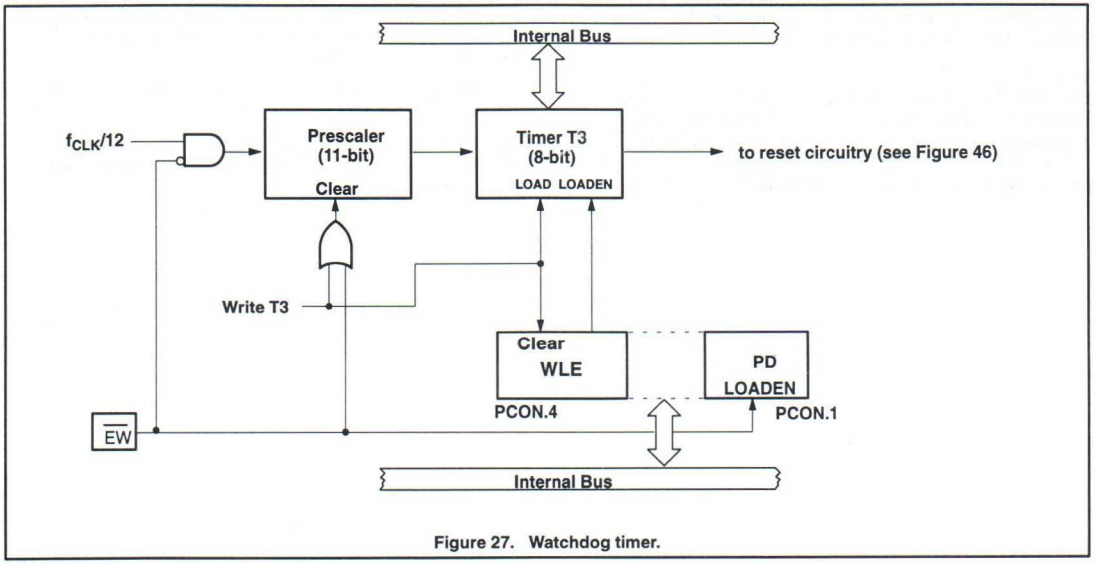



**Figure 27. Watchdog timer.**

## 6.8 Watchdog Timer T3

In addition to Timer T2 and the standard timers, a watchdog timer (T3) consisting of an 11–bit prescaler and an 8–bit timer is also incorporated (see Figure 27). The timer is incremented every 1.5 ms, derived from the system clock frequency of 16 MHz by the following:

$$f_{TIMER} = \frac{f_{CLK}}{12 \times 2048}$$

When a timer overflow occurs, the microcontroller is reset and a reset output pulse is generated at pin RSTOUT. Also the PLL control register is reset.

To prevent a system reset the timer must be reloaded in time by the application software. If the processor suffers a hardware/software malfunction, the software will fail to reload the timer. This failure will produce a reset upon overflow thus preventing the processor running out of control.

The watchdog timer can only be reloaded if the condition flag WLE = PCON.4 has been previously set by software.

At the moment the counter is loaded the condition flag is automatically cleared.

The time interval between the timer's reloading and the occurrence of a reset depends on the reloaded value. For example, this may range from 1.5 ms to 0.375 s when using an oscillator frequency of 16 MHz.

In the Idle state the watchdog timer and reset circuitry remain active.

The watchdog timer is controlled by the watchdog enable pin ($\overline{EW}$). A LOW level enables the watchdog timer and disables the Power–down Mode. A HIGH level disables the watchdog timer and enables the Power–down Mode.

## 6.9 Serial I/O

The P8xCE559 is equipped with two independent serial ports: SIO0 and SI01. SIO0 is the full duplex UART port, identical to the PCB80C51 serial port. SIO1 is an $I^2C$–bus serial I/O interface with byte oriented master and slave functions.

### 6.9.1 SIO0 (UART)

SIO 0 is a full duplex serial I/O port – it can transmit and receive simultaneously. This serial port is also receive–buffered. It can commence reception of a second byte before the previously received byte has been read from the receive register. If, however, the first byte has still not been read by the time reception of the second byte is complete, one of the bytes will be lost. The SIO0 receive and transmit registers are both accessed via the S0BUF special function register. Writing to S0BUF loads the transmit register, and reading S0BUF accesses to a physically separate receive register. SIO0 can operate in 4 modes:

**Mode 0:** Serial data is transmitted and received through RXD. TXD outputs the shift clock. 8 data bits are transmitted/received (LSB first). The baud rate is fixed at 1/12 of the oscillator frequency. A write into S0CON should be avoided during a transmission to avoid spikes on RXD/TXD.

**Mode 1:** 10 bits are transmitted via TXD or received through RXD: a start bit (0), 8 data bits (LSB first), and a stop bit(1). On receive, the stop bit is put into RB8 (S0CON special function register). The baud rate is variable.

**Mode 2:** 11 bits are transmitted through TXD or received through RXD: a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). On transmit, the 9th data bit (TB8 in S0CON) can be assigned the value of 0 or 1. With nominal software, TB8 can be the parity bit (P in PSW). During a receive, the 9th data bit is stored in RB8 (S0CON), and the stop bit is ignored. The baud rate is programmable to either 1/32 or 1/64 of the oscillator frequency.

**Mode 3:** 11 bits are transmitted through TXD or received through RXD: a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). Mode 3 is the same as Mode 2 except the baud rate which is variable in Mode 3.

In all four modes, transmission is initiated by any instruction that writes to the S0BUF function register. Reception is initiated in Mode 0 when RI = 0 and REN = 1. In the other three modes, reception is initiated by the incoming start bit provided that REN = 1.

Modes 2 and 3 are provided for multiprocessor communications. In these modes, 9 data bits are received with the 9th bit written to RB8. The 9th bit is followed by the stop bit. The port can be programmed so that with receiving the stop bit, the serial port interrupt will be activated if, and only if RB8 = 1.

This feature is enabled by setting bit SM2 in S0CON. This feature may be used in multiprocessor systems.

For more information about how to use the UART in combination with the registers S0CON, PCON, IEN0, S0BUF and Timer register refer to the 80C51 Data Handbook IC20.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **S0CON (98H)** | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |

**Figure 28. Serial port control (S0CON) register.**

**Table 22. Description of S0CON bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| SM0 | S0CON.7 | This bit is used to select the serial port mode. See Table 23 |
| SM1 | S0CON.6 | This bit is used to select the serial port mode. See Table 23 |
| SM2 | S0CON.5 | Enables the multiprocessor communication feature in modes 2 and 3. In mode 2 or 3, if SM2 is set to 1, then RI will not be activated if the received 9th data bit (RB8) is 0. In mode 1, if SM2 = 1, then RI will not be activated if a valid stop bit was not received. In mode 0, SM2 should be 0. |
| REN | S0CON.4 | Enables serial reception. Set by software to enable reception. Clear by software to disable reception. |
| TB8 | S0CON.3 | The 9th data bit that will be transmitted in modes 2 and 3. Set or clear by software as desired. |
| RB8 | S0CON.2 | In modes 2 and 3, RB8 is the 9th data bit that was received. In mode 1, if SM2 = 0, RB8 is the stop bit that was received. In mode 0, RB8 is not used. |
| TI | S0CON.1 | The transmit interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software. |
| RI | S0CON.0 | The receive interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software. |

**Table 23. Description of S0CON bits**

| SM0 | SM1 | MODE | DESCRIPTION | BAUD RATE |
|---|---|---|---|---|
| 0 | 0 | 0 | Shift register | $f_{CLK}/12$ |
| 0 | 1 | 1 | 8–bit UART | variable |
| 1 | 0 | 2 | 9–bit UART | $f_{CLK}/64$ or $f_{CLK}/32$ |
| 1 | 1 | 3 | 9–bit UART | variable |

### 6.9.2 SIO1 ($I^2C$-bus Interface)

The SIO1 of the P8xCE559 provides the fast–mode, which allows a fourthfold increase of the bitrate up to 400 kHz. Nevertheless it is downward compatible, i.e. it can be used in a 0 to 100 Kbit/s $I^2C$ bus system.

Except from the bit rate selection (see Table 25) and the timing of the SCL and SDA signals (see AC electrical characteristics in section 11) the SIO circuit is the same as described in detail in the 80C51 Data Handbook IC20 for the 8xC552 microcontroller.

The $I^2C$–bus is a simple bidirectional 2–wire bus for efficient inter–IC data exchange. Features of the $I^2C$–bus are:

- Only two bus lines are required: a serial clock line (SCL) and a serial data line (SDA)
- Each device connected to the bus is software addressable by a unique address
- Masters can operate as Master–transmitter or as Master–receiver
- It's a true multi–master bus including collision detection and arbitration to prevent data corruption if two or more masters simultaneously initiate data transfer
- Serial clock synchronization allows devices with different bit rates to communicate via the same serial bus
- ICs can be added to or removed from an $I^2C$–bus system without affecting any other circuit on the bus
- Fault diagnostics and debugging are simple; malfunctions can be immediately traced

For more information on the $I^2C$–bus specification (including fast–mode) please refer to the Philips publication number 9398 393 40011 and/or the 80C51 Data Handbook IC20.

The on–chip $I^2C$ logic provides a serial interface that meets the $I^2C$–bus specification, supporting all $I^2C$–bus modes of operation, they are:

- Master transmitter
- Master receiver
- Slave transmitter
- Slave receiver

The SI01 logic performs a byte oriented data transport, clock generation, address recognition and bus control arbitration are all controlled by hardware. Via two pins the external $I^2C$–bus is interfaced to the SIO1 logic: SCL serial clock I/O and SDA serial data I/O, (see Special Function Register bit S1CON.6/ENS1 for enabling the SIO1 logic).

The SIO1 logic handles byte transfer autonomously. It keeps track of the serial transfers, and a status register (S1STA) reflects the status of SIO1 and the $I^2C$–bus.

Via the following four Special Function Registers the CPU interfaces to the $I^2C$ logic.

S1CON control register. Bit addressable by the CPU

S1STA status register whose contents may be used as a vector to service routines.

S1DAT data shift register. The data byte is stable as long as S1CON.3/SI=1.

S1ADR slave address register. It's LSB enables/ disables general call address recognition.

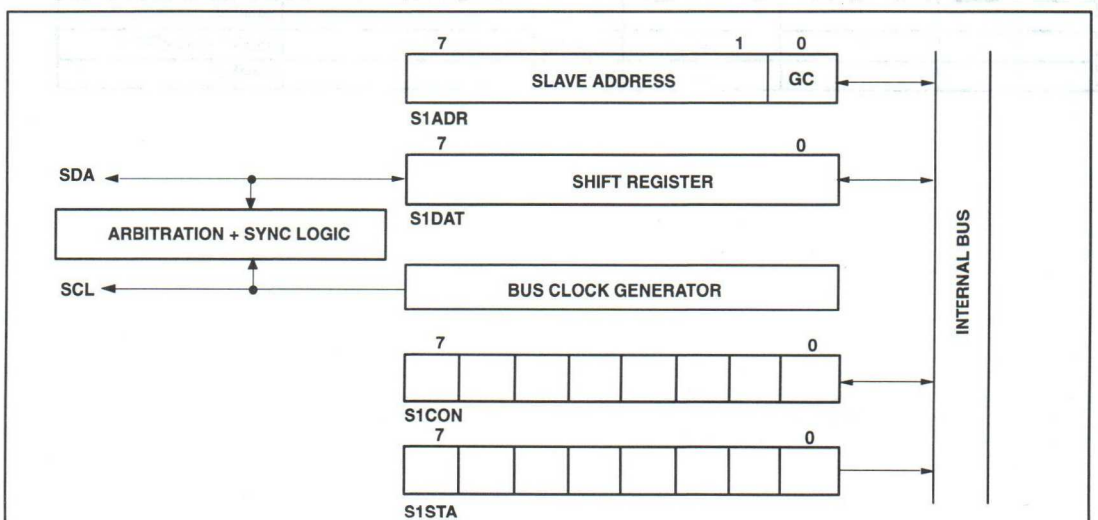



**Figure 29. Block diagram of $I^2C$ serial I/O interface.**

**The Control Register, S1CON**

The CPU can read from and write to this 8-bit, directly addressable SFR. Two bits are affected by the SIO1 hardware: the SI bit is set when a serial interrupt is requested, and the STO bit is cleared when a STOP condition is present on the $I^2C$ bus. The STO bit is also cleared when ENS1 = 0.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **S1CON (D8H)** | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |

**Figure 30. Serial control (S1CON) register.**

**Table 24. Description of S1CON bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| CR2 | S1CON.7 | Clock rate bit 2, see Table 25. |
| ENS1 | S1CON.6 | ENS1 = 0: Serial I/O disabled and reset. SDA and SCL outputs are high–Z.<br>ENS1 = 1: Serial I/O enabled. |
| STA | S1CON.5 | START flag. When this bit is set in slave mode, the hardware checks the $I^2C$ bus and generates a START condition if the bus is free or after the bus becomes free. If the device operates in master mode it will generate a repeated START condition. |
| STO | S1CON.4 | STOP flag. If this bit is set in a master mode a STOP condition is generated. A STOP condition detected on the $I^2C$ bus clears this bit. This bit may also be set in slave mode in order to recover from an error condition. In this case no STOP condition is generated to the $I^2C$ bus, but the hardware releases the SDA and SCL lines and switches to the not selected receiver mode. The STOP flag is cleared by the hardware. |
| SI | S1CON.3 | Serial Interrupt flag. This flag is set, and an interrupt request is generated, after any of the following events occur:<br>– A START condition is generated in master mode.<br>– The own slave address has been received during AA = 1.<br>– The general call address has been received while S1ADR.0 and AA = 1.<br>– A data byte has been received or transmitted in master mode (even if arbitration is lost).<br>– A data byte has been received or transmitted as selected slave.<br>– A STOP or START condition is received as selected slave receiver or transmitter.<br>While the SI flag is set, SCL remains LOW and the serial transfer is su–spended. SI must be reset by software. |
| AA | S1CON.2 | Assert Acknowledge flag. When this bit is set, an acknowledge is returned after any one of the following conditions:<br>– Own slave address is received.<br>– General call address is received (S1ADR.0 = 1).<br>– A data byte is received, while the device is programmed to be a master receiver.<br>– A data byte is received. while the device is a selected slave receiver.<br>When the bit is reset, no acknowledge is returned. Consequently, no interrupt is requested when the own address or general call address is received. |
| CR1<br>CR0 | S1CON.1<br>S1CON.0 | Clock rate bits 1 and 0, see Table 25. |

When SIO1 is in a master mode serial clock frequency is determined by the clock rate bits CR2, CR1 and CR0. The various bit rates are shown in Table 25.

The data shown in Table 25 do not apply to SIO1 in a slave mode. In the slave modes, SIO1 will automatically synchronize with any clock frequency up to 400kHz.

### Table 25. Selection of I²C-bus bit rate

| CR2 | CR1 | CR0 | BIT RATE (kHz) at $f_{CLK}$ | |
|---|---|---|---|---|
| | | | 12MHz | 16MHz |
| 1 | 0 | 0 | 50 | 66.7 |
| 1 | 0 | 1 | 3.75 | 5 |
| 1 | 1 | 0 | 75 | 100 |
| 1 | 1 | 1 | 100 | – |
| 0 | 0 | 0 | 200 [1] | 266.7 [1] |
| 0 | 0 | 1 | 7.5 | 10 |
| 0 | 1 | 0 | 300 [1] | 400 [1] |
| 0 | 1 | 1 | 400 [1] | – |

**NOTE:**
1. These bit rates are for "fast–mode" I²C bus applications and cannot be used for bit rates up to 100 kbit/sec.

**Serial status register S1STA**
S1STA is a read only register.

The contents of the status register may be used as a vector to a service routine. This optimizes the response time of the software and consequently that of the I²C–bus.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **S1STA (D9H)** | SC4 | SC3 | SC2 | SC1 | SC0 | 0 | 0 | 0 |

**Figure 31. Serial status (S1STA) register.**

### Table 26. Description of S1STA bits

| BIT | FUNCTION |
|---|---|
| S1STA.7 to 3 | 5–bit status code |
| S1STA.2 to 0 | These bits are held LOW (for service routine vector increment 8) |

The following is a list of the status codes:

### Table 27. MST/TRX mode

| S1STA VALUE | DESCRIPTION |
|---|---|
| 08H | A START condition has been transmitted |
| 10H | A repeated START condition has been transmitted |
| 18H | SLA and W have been transmitted, ACK has been received |
| 20H | SLA and W have been transmitted, $\overline{ACK}$ received |
| 28H | DATA and S1DAT has been transmitted, ACK received |
| 30H | DATA and S1DAT has been transmitted, $\overline{ACK}$ received |
| 38H | Arbitration lost in SLA, R/W or DATA |

**Table 28. MST/REC mode**

| S1STA VALUE | DESCRIPTION |
|---|---|
| 38H | Arbitration lost while returning $\overline{ACK}$ |
| 40H | SLA and R have been transmitted, ACK received |
| 48H | SLA and R have been transmitted, $\overline{ACK}$ received |
| 50H | DATA has been received, ACK returned |
| 58H | DATA has been received, $\overline{ACK}$ returned |

**Table 29. SLV/REC mode**

| S1STA VALUE | DESCRIPTION |
|---|---|
| 60H | Own SLA and W have been received, ACK returned |
| 68H | Arbitration lost in SLA, R/W as MST. Own SLA and W have been received, $\overline{ACK}$ returned |
| 70H | General CALL has been received, ACK returned |
| 78H | Arbitration lost in SLA, R/W as MST. General call has been received |
| 80H | Previously addressed with own SLA. DATA byte received, ACK returned |
| 88H | Previously addressed with own SLA. DATA byte received, $\overline{ACK}$ returned |
| 90H | Previously addressed with general call. DATA byte has been received, ACK has been returned |
| 98H | Previously addressed with general call. DATA byte has been received, $\overline{ACK}$ has been returned |
| A0H | A STOP condition or repeated START condition has been received while still addressed as SLV/REC or SLV/TRX |

**Table 30. SLV/TRX mode**

| S1STA VALUE | DESCRIPTION |
|---|---|
| A8H | Own SLA and R have been received, ACK returned |
| B0H | Arbitration lost in SLA, R/W as MST. Own SLA and R have been received, ACK returned |
| B8H | DATA byte has been transmitted, ACK returned |
| C0H | DATA byte has been transmitted, $\overline{ACK}$ returned |
| C8H | Last DATA byte has been transmitted (AA = logic 0), ACK received |

**Table 31. Miscellaneous**

| S1STA VALUE | DESCRIPTION |
|---|---|
| 00H | Bus error during MST mode or selected SLV mode, due to an erroneous START or STOP condition |
| F8H | No relevant information available, SI not set |

Abbreviations used:

| | | |
|---|---|---|
| SLA | : | 7–bit slave address |
| R | : | Read bit |
| W | : | Write bit |
| ACK | : | Acknowledgement (acknowledge bit = 0) |
| $\overline{ACK}$ | : | Not acknowledgement (acknowledge bit = 1) |
| DATA | : | 8–bit data byte to or from $I^2C$–bus |
| MST | : | Master |
| SLV | : | Slave |
| TRX | : | Transmitter |
| REC | : | Receiver |

**The data shift register S1DAT**
This register contains the serial data to be transmitted or data which has been received. Bit 7 is transmitted or received first; i.e. data is shifted from right to left.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **S1DAT (DAH)** | S1DAT.7 | S1DAT.6 | S1DAT.5 | S1DAT.4 | S1DAT.3 | S1DAT.2 | S1DAT.1 | S1DAT.0 |

**Figure 32. Data shift register.**

**The address register S1ADR**
This 8–bit register may be loaded with the 7–bit slave address to which the controller will respond when programmed as a slave receiver/transmitter. The LSB (GC) is used to determine whether the general call address is recognized.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **S1ADR (DBH)** | SLA6 | SLA5 | SLA4 | SLA3 | SLA2 | SLA1 | SLA0 | GC |

**Figure 33. Address register.**

## Table 32. Description of S1ADR bits

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| SLA6 to 0 | S1ADR.7 to 1 | Own slave address |
| GC | S1ADR.0 | 0 = general call address is not recognized<br>1 = general call address is recognized |

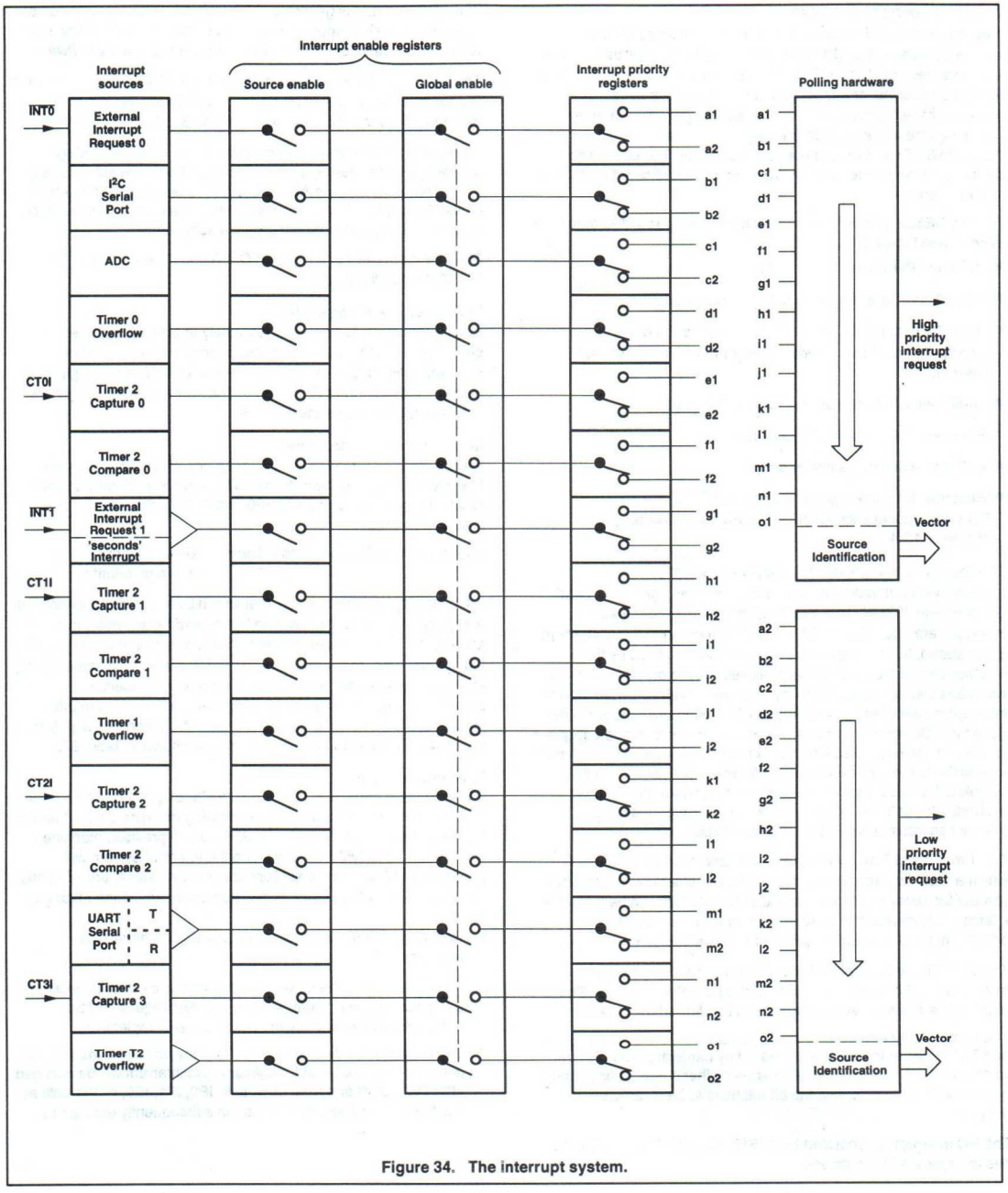



**Figure 34. The interrupt system.**

## 6.10 Interrupt System

External events and the real–time–driven on–chip peripherals require service by the CPU asynchronously to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple–source, two–priority–level, nested interrupt system is provided. Interrupt response time in a single–interrupt system is in the range from 2.25us to 6.75 us when using a 16 MHz crystal. The latency time depends on the sequence of instructions executed directly after an interrupt request.

The P8xCE559 acknowledges interrupt requests from 15 sources as follows (see Figure 34):

- $\overline{INT0}$ and $\overline{INT1}$ external interrupts
- Timer 0 and Timer 1 internal timer/counter interrupts
- Timer 2 internal timer/counter byte and/or 16–bit overflow, 3 compare and 4 capture interrupts (or 4 additional external interrupts) (1)
- UART serial I/O port receive/transmit interrupt
- I²C–bus interface serial I/O interrupt
- ADC autoscan completion interrupt
- 'Seconds' timer interrupt SEC (ored with $\overline{INT1}$). For details about seconds timer interrupts, please refer to chapter 6.13.4

The External Interrupts $\overline{INT0}$ and $\overline{INT1}$ can each be either level–activated or transition–activated, depending on bits IT0 and IT1 in register TCON. The flags that actually generate these interrupts are bits IE0 and IE1 in TCON. When an external interrupt is generated, the corresponding request flag is cleared by the hardware when the service routine is vectored to only if the interrupt was transition–activated. If the interrupt was level–activated then the interrupt request flag remains set until the external interrupt pin $\overline{INTx}$ goes high. Consequently the external source has to hold the request active until the requested interrupt is actually generated. Then it has to deactivate the request before the interrupt service routine is completed, or else another interrupt will be generated. As these external interrupts are active LOW a "wire–ORing" of several interrupt sources to one input pin allows expansion.

The Timer 0 and Timer 1 Interrupts are generated by TF0 and TF1, which are set by a rollover in their respective timer/counter register (except for Timer 0 in Mode 3 of the serial interface). When a Timer interrupt is generated, the flag that generated it is cleared by the on–chip hardware when the service routine is vectored to.

The eight Timer/Counter T2 Interrupt sources are: 4 capture Interrupts (1), 3 compare interrupts and an overflow interrupt. The appropriate interrupt request flags must be cleared by software.

The UART Serial Port Interrupt is generated by the logical OR of RI and TI. Neither of these flags is cleared by hardware. The service routine will normally have to determine whether it was RI or TI that generated the interrupt, and the bit will have to be cleared by software.

The I²C Interrupt is generated by bit SI in register S1CON. This flag has to be cleared by software.

The ADC Interrupt is generated by bit ADINT, which is set when of all selected analog inputs to be scanned, the conversion is finished. ADINT must be cleared by software. It cannot be set by software.

The 'Seconds' timer Interrupt is generated by bit SECINT in register PLLCON. This flag has to be cleared by software. Note that the 'Seconds' timer can only be used with the 32 kHz PLL oscillator.

All of the bits that generate interrupts can be set or cleared by software, with the same result as though it had been set or cleared by hardware (except the ADC interrupt request flag ADINT, which cannot be set by software). That is, interrupts can be generated or pending interrupts can be cancelled in software.

The Interrupts X0, T0, X1, T1, SEC, S0 and S1 are capable to terminate the Idle Mode.

**Interrupt Enable Registers**

Each interrupt source can be individually enabled or disabled by setting or clearing a bit in the interrupt enable special function registers IEN0 and IEN1. All interrupt sources can also be globally disabled by clearing bit EA in IEN0. The interrupt enable registers are described in Figures 35 and 36.

**Interrupt Priority Structure**

Each interrupt source can be assigned one of two priority levels. Interrupt priority levels are defined by the interrupt priority special function registers IP0 and IP1. IP0 and IP1 are described in Figures 37 and 38.

Interrupt priority levels are as follows: "0"—low priority
"1"—high priority

A low priority interrupt may be interrupted by a high priority interrupt. A high priority interrupt cannot be interrupted by any other interrupt source. If two requests of different priority occur simultaneously, the high priority level request is serviced. If requests of the same priority are received simultaneously, an internal polling sequence determines which request is serviced. Thus, within each priority level, there is a second priority structure determined by the polling sequence. This second priority structure is shown in Table 37.

**Interrupt Handling**

The interrupt sources are sampled at S5P2 of every machine cycle. The samples are polled during the following machine cycle. If one of the flags was in a set condition at S5P2 of the previous machine cycle, the polling cycle will find it and the interrupt system will generate an LCALL to the appropriate service routine, provided this hardware- generated LCALL is not blocked by any of the following conditions:

1.. An interrupt of higher or equal priority level is already in progress.

2.. The current machine cycle is not the final cycle in the execution of the instruction in progress. (No interrupt request will be serviced until the instruction in progress is completed.)

3.. The instruction in progress is RETI or any access to the interrupt priority or interrupt enable registers. (No interrupt will be serviced after RETI or after a read or write to IP0, IP1, IE0, or IE1 until at least one other instruction has been subsequently executed.)

1. If a capture register is unused and it's contents is of no interest, then the corresponding input pin CTnI/P1.n (n: 0...3) may be used as a (configurable) positive and/or negative edge triggered additional external interrupt input (INT2, INT3, INT4, INT5).

The polling cycle is repeated with every machine cycle, and the values polled are the values present at S5P2 of the previous machine cycle. Note that if an interrupt flag is active but is not being responded to because of one of the above conditions, and if the flag is inactive when the blocking condition is removed, then the blocked interrupt will not be serviced. Thus, the fact that the interrupt flag was once active but not serviced is not remembered. Every polling cycle is new.

The processor acknowledges an interrupt request by executing a hardware-generated LCALL to the appropriate service routine. In some cases it also clears the flag which generated the interrupt, and in others it does not. It clears the Timer 0, Timer 1, and external interrupt flags. An external interrupt flag (IE0 or IE1) is cleared only if it was transition-activated. All other interrupt flags are not cleared by hardware and must be cleared by the software. The LCALL pushes the contents of the program counter on to the stack (but it does not save the PSW) and reloads the PC with an address that depends on the source of the interrupt being vectored to as shown in Table 38.

Execution proceeds from the vector address until the RETI instruction is encountered. The RETI instruction clears the "priority level active" flip-flop that was set when this interrupt was acknowledged. It then pops the top two bytes from the stack and reloads the program counter. Execution of the in– terrupted program continues from where it was interrupted.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **IEN0 (A8H)** | EA | EAD | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 |

**Figure 35. Interrupt enable register (IEN0).**

**Table 33. Description of IEN0 bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| EA | IEN0.7 | Global enable/disable control<br>0 = No interrupt is enabled<br>1 = Any individually enabled interrupt will be accepted |
| EAD | IEN0.6 | Enable ADC interrupt |
| ES1 | IEN0.5 | Enable SIO1 ($I^2C$) interrupt |
| ES0 | IEN0.4 | Enable SIO0 (UART) interrupt |
| ET1 | IEN0.3 | Enable Timer 1 interrupt |
| EX1 | IEN0.2 | Enable External interrupt 1 / Seconds interrupt |
| ET0 | IEN0.1 | Enable Timer 0 interrupt |
| EX0 | IEN0.0 | Enable External interrupt 0 |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **IEN1 (E8H)** | ET2 | ECM2 | ECM1 | ECM0 | ECT3 | ECT2 | ECT1 | ECT0 |

**Figure 36. Interrupt enable register (IEN1).**

**Table 34. Description of IEN1 bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| ET2 | IEN1.7 | Enable T2 overflow interrupt(s) |
| ECM2 | IEN1.6 | Enable T2 comparator 2 interrupt |
| ECM1 | IEN1.5 | Enable T2 comparator 1 interrupt |
| ECM0 | IEN1.4 | Enable T2 comparator 0 interrupt |
| ECT3 | IEN1.3 | Enable T2 capture register 3 interrupt |
| ECT2 | IEN1.2 | Enable T2 capture register 2 interrupt |
| ECT1 | IEN1.1 | Enable T2 capture register 1 interrupt |
| ECT0 | IEN1.0 | Enable T2 capture register 0 interrupt |

If the enable bit is 0, then the interrupt is disabled, if the enable bit is 1, then the interrupt is enabled.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| IP0 (B8H) | – | PAD | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 |

**Figure 37. Interrupt priority register (IP0).**

**Table 35. Description of IP0 bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| – | IP0.7 | Reserved for future use |
| PAD | IP0.6 | ADC interrupt priority level |
| PS1 | IP0.5 | SIO1 ($I^2C$) interrupt priority level |
| PS0 | IP0.4 | SIO0 (UART) interrupt priority level |
| PT1 | IP0.3 | Timer 1 interrupt priority level |
| PX1 | IP0.2 | External interrupt 1/Seconds interrupt priority level |
| PT0 | IP0.1 | Timer 0 interrupt priority level |
| PX0 | IP0.0 | External interrupt 0 priority level |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| IP1 (F8H) | PT2 | PCM2 | PCM1 | PCM0 | PCT3 | PCT2 | PCT1 | PCT0 |

**Figure 38. Interrupt priority register (IP1).**

**Table 36. Description of IP1 bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| PT2 | IP1.7 | T2 overflow interrupt(s) priority level |
| PCM2 | IP1.6 | T2 comparator 2 interrupt priority level |
| PCM1 | IP1.5 | T2 comparator 1 interrupt priority level |
| PCM0 | IP1.4 | T2 comparator 0 interrupt priority level |
| PCT3 | IP1.3 | T2 capture register 3 interrupt priority level |
| PCT2 | IP1.2 | T2 capture register 2 interrupt priority level |
| PCT1 | IP1.1 | T2 capture register 1 interrupt priority level |
| PCT0 | IP1.0 | T2 capture register 0 interrupt priority level |

**Table 37. Interrupt Priority Structure**

| SOURCE | NAME | PRIORITY WITHIN LEVEL |
|---|---|---|
| | | (highest) ↑ |
| External interrupt 0 | X0 | |
| SIO1 ($I^2C$) | S1 | |
| ADC completion | ADC | |
| Timer 0 overflow | T0 | |
| Timer 2 capture 0 | CT0 | |
| Timer 2 compare 0 | CM0 | |
| External interrupt 1/Seconds interrupt | X1/SEC | |
| Timer 2 capture 1 | CT1 | |
| Timer 2 compare 1 | CM1 | |
| Timer 1 overflow | T1 | |
| Timer 2 capture 2 | CT2 | |
| Timer 2 compare 2 | CM2 | |
| SIO0 (UART) | S0 | |
| Timer 2 capture 3 | CT3 | |
| Timer 2 overflow | T2 | ↓ (lowest) |

**Table 38. Interrupt Vector Addresses**

| SOURCE | NAME | VECTOR ADDRESS |
|---|---|---|
| External interrupt 0 | X0 | 0003H |
| Timer 0 overflow | T0 | 000BH |
| External interrupt 1/Seconds interrupt | X1/SEC | 0013H |
| Timer 1 overflow | T1 | 001BH |
| SIO0 (UART) | S0 | 0023H |
| SIO1 ($I^2C$) | S1 | 002BH |
| Timer 2 capture 0 | CT0 | 0033H |
| Timer 2 capture 1 | CT1 | 003BH |
| Timer 2 capture 2 | CT2 | 0043H |
| Timer 2 capture 3 | CT3 | 004BH |
| ADC completion | ADC | 0053H |
| Timer 2 compare 0 | CM0 | 005BH |
| Timer 2 compare 1 | CM1 | 0063H |
| Timer 2 compare 2 | CM2 | 006BH |
| Timer 2 overflow | T2 | 0073H |

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PCON (87H) | SMOD | ARD | RFI | WLE | GF1 | GF0 | PD | IDL |

**Figure 39. Power control register (PCON).**

**Table 39. Description of PCON bits**

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| SMOD | PCON.7 | Double Baud rate bit. When set to logic 1 the baud rate is doubled when the serial port SIO0 is being used in modes 1, 2, or 3. |
| ARD | PCON.6 | AUX–RAM disable bit. When set to a 1 the internal 1280 bytes AUX–RAM is disabled, so that all MOVX–Instructions access the external data memory – as it is with the standard PCB80C51. |
| RFI | PCON.5 | Reduced radio frequency interference bit. When set to a 1 the toggling of ALE pin is prohibited. This bit is cleared on RESET (see also sections Features (EMC) and Pinning). |
| WLE | PCON.4 | Watchdog load enable. This flag must be set by software prior to loading timer T3 (watchdog timer). It is cleared when timer T3 is loaded. |
| GF1 | PCON.3 | General-purpose flag bit |
| GF0 | PCON.2 | General-purpose flag bit |
| PD | PCON.1 | Power-down bit. Setting this bit activates the power-down mode. It can only be set if input $\overline{EW}$ is high. |
| IDL | PCON.0 | Idle Mode bit. Setting this bit activates the Idle Mode. |

## 6.11 Power Reduction Modes

Two software–selectable modes of reduced power consumption are implemented. These are the Idle Mode and the Power–down Mode.

Idle Mode operation permits the interrupt, serial ports and timer blocks T0, T1 and T3 to function while the CPU is halted. The following functions are switched off when the microcontroller enters the Idle Mode:

- CPU (halted)
- Timer 2 (stopped and reset)
- PWM0, PWM1 (reset, output = HIGH)
- ADC (aborted if conversion in progress)

The following functions remain active during Idle Mode. These functions may generate an interrupt or reset and thus terminate the Idle Mode:

- Timer 0, Timer 1, Timer 3 (Watchdog timer)
- UART
- $I^2C$
- External interrupt
- Seconds Timer

In Power–down Mode the system clock is halted. If the PLL oscillator is selected (SELXTAL1 = 0) and the RUN32 bit is set, the 32 kHz oscillator keeps running, otherwise it is stopped. If the HF–oscillator (SELXTAL1 = 1) is selected, it is stopped after setting the bit PD in the PCON register.

### 6.11.1 Power Control Register

The modes Idle and Power–down are activated by software via the Special Function Register PCON. Its hardware address is 87H.

PCON is not bit addressable. The reset value of PCON is (00000000).

### 6.11.2 Idle Mode

The instruction that sets PCON.0 is the last instruction executed in the normal operating mode before Idle Mode is activated. Once in the Idle Mode, the CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator, RAM and all other registers maintain their data during Idle Mode. The status of external pins during Idle Mode is shown in Table 40.

There are three ways to terminate the Idle Mode:

Activation of any enabled interrupt X0, T0, X1, SEC, T1, S0 or S1 will cause PCON.0 to be cleared by hardware terminating Idle Mode but only, if there is no interrupt in service with the same or higher priority. The interrupt is serviced, and following return from interrupt instruction RETI, the next instruction to be executed will be the one which follows the instruction that wrote a logic 1 to PCON.0.

The flag bits GF0 and GF1 may be used to determine whether the interrupt was received during normal execution or during Idle Mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When Idle Mode is terminated by an interrupt, the service routine can examine the status of the flag bits.

The second way of terminating the Idle Mode is with an external hardware reset. Since the oscillator is still running, the hardware reset is required to be active for two machine cycles (24 HF oscillator periods) to complete the reset operation if the HF oscillator is selected.

When the PLL oscillator is selected a hardware reset of $\geq 1$ µsec (but no longer than 10 ms) is required and the microcontroller will typically restart within 63 msec after the reset has finished.

The third way of terminating the Idle Mode is by internal watchdog reset. The microcontroller restarts after 3 machine cycles in all cases.

**Table 40. External Pin Status During Idle and Power-Down Modes**

| MODE | MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 | PORT 4 | SCL/SDA | $\overline{PWM0}$/$\overline{PWM1}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | data | data | data | data | data | operative (1) | HIGH |
| Idle | External | 1 | 1 | high – Z | data | address | data | data | operative (1) | HIGH |
| Power-down | Internal | 0 | 0 | data | data | data | data | data | high–Z | HIGH |
| Power-down | External | 0 | 0 | high – Z | data | data | data | data | high–Z | HIGH |

**NOTE:**
1. In Idle Mode SCL and SDA can be active as outputs only if SIO1 is enabled; if SIO1 is disabled (S1CON.6/ENS1 = 0) these pins are in a high–impedance state.

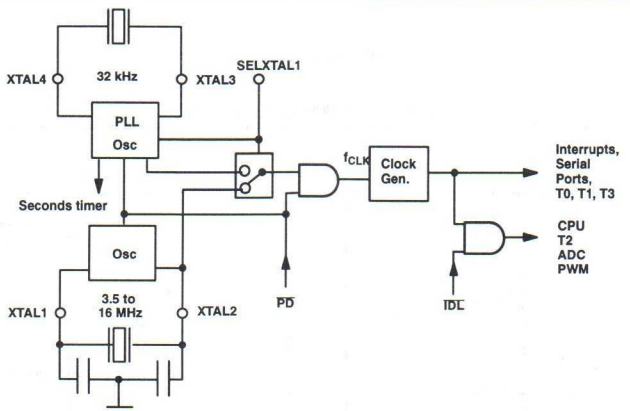



**Figure 40. Idle and Power Down Hardware for Clock Generation**

### 6.11.3 Power-down Mode

The instruction that sets PCON.1 is the last executed prior to going into the Power-down Mode. Once in Power-down Mode, the HF oscillator is stopped. The 32 kHz oscillator may stay running. The content of the on-chip RAM and the Special Function Registers are preserved. Note that the Power-down Mode can not be entered when the watchdog has been enabled.

The Power-down Mode can be terminated by an external RESET in the same way as in the 80C51 (RAM is saved, but SFRs are cleared due to RESET) or in addition by any one of the external interrupts ($\overline{INT0}$, $\overline{INT1}$) or Seconds interrupt.

The status of the external pins during Power-down Mode is shown in Table 40. If the Power-down Mode is activated while in external program memory, the port data that is held in the Special Function Register P2 is restored to Port 2. If the data is a logic1, the port pin is held HIGH during the Power-down Mode by the strong pull-up transistor P1 (see Figure 9).

The Power-down Mode should not be entered within an interrupt routine because Wake-up with an external or 'Seconds' interrupt is not possible in that case.

### 6.11.4 Wake-up from Power-down Mode

The Power-down Mode of the P8xCE559 can also be terminated by any one of the three enabled interrupts, $\overline{INT0}$, $\overline{INT1}$ or Seconds interrupt.

If there is an interrupt already in service, which has same or higher priority as the Wake-up interrupt, Power-down Mode will switch over to Idle Mode and stay there until an interrupt of higher priority terminates Idle Mode.

A termination with these interrupts does not affect the internal data memory and does not affect the Special Function Registers. This gives the possibility to exit Power-down without changing the port output levels. To terminate the Power-down Mode with an external interrupt, $\overline{INT0}$ or $\overline{INT1}$ must be switched to be level-sensitive and must be enabled. The external interrupt input signal $\overline{INT0}$ or $\overline{INT1}$ must be kept LOW till the oscillator has restarted and stabilized (see Figure 41). A Seconds interrupt will terminate the Power-down Mode if it is enabled and $\overline{INT1}$ is level sensitive. In order to prevent any interrupt priority problems during Wake-up, the priority of the desired Wake-up interrupt should be higher than the priorities of all other enabled interrupt sources.

The instruction following the one that put the device into the Power-down Mode will be the first one which will be executed after the interrupt routine has been serviced.

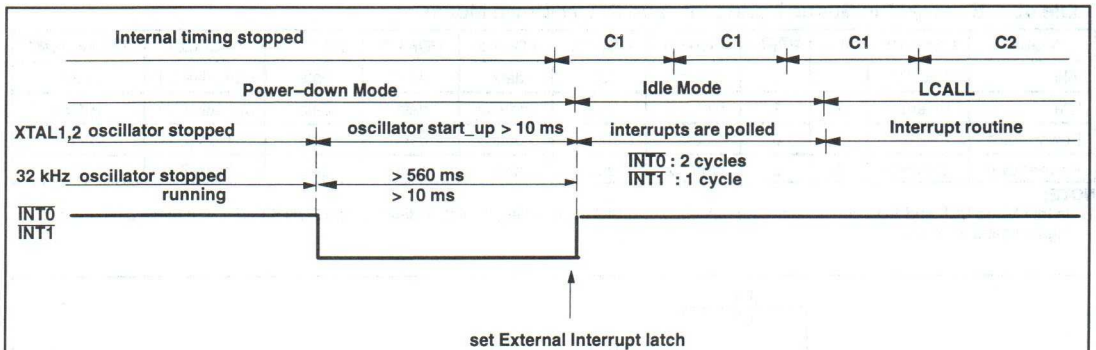



Figure 41. Wake-up by interrupt

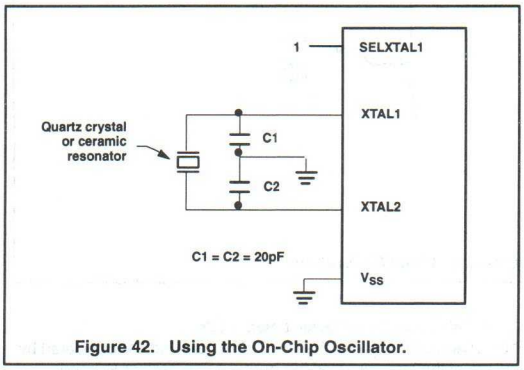



Figure 42. Using the On-Chip Oscillator.

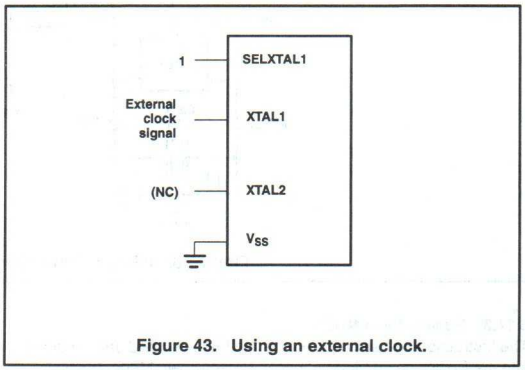



Figure 43. Using an external clock.

## 6.12 Oscillator Circuits

The input signal SELXTAL1 connected to logic "1" selects the XTAL1, 2 oscillator (standard 80C51) instead of the XTAL3, 4 oscillator, which is halted and XTAL3, 4 must not be connected.

### 6.12.1 XTAL1, 2 Oscillator circuit (standard 80C51)

The oscillator circuit of the P8xCE559 is a single–stage inverting amplifier in a Pierce oscillator configuration. The circuitry between the XTAL1 and XTAL2 is basically an inverter biased to the transfer point. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuitry. Both are operated in parallel resonance. XTAL1 is the high gain amplifier input, and XTAL2 is the output (see Figure 42). To drive the P8xCE559 externally, XTAL1 is driven from an external source and XTAL2 left open–circuit (see Figure 43).

### 6.12.2 XTAL3, 4 Circuitry

Please refer to chapter 6.13.1

## 6.13 32 kHz PLL Oscillator with Seconds Timer

### 6.13.1 XTAL3,4 Oscillator Circuitry

The input signal SELXTAL1 connected to logic "0" selects the 32kHz oscillator together with the PLL instead of the XTAL1,2 oscillator, which is halted. XTAL2 is floating in that case.

The 32kHz oscillator consists of an inverter, which forms a Pierce oscillator with the on–chip components C1,C2,Rf and an external crystal of 32768 Hz.

During the following situations, the inverter is switched to tristate and XTAL3 is pulled to Vss :

- during Power–down Mode, when the PLL control register bit RUN32 (PLLCON.7) was set to '0';
- during Reset (RSTIN = HIGH) ;
- when the XTAL1,2 oscillator is selected (SELXTAL1 = HIGH).

### 6.13.2 PLL CCO

A current controlled oscillator (CCO) generates a clock frequency $f_{CCO}$ of approx. 32 , 38 , 44 or 50 MHz , controlled by the PLL, with the 32kHz oscillator as the reference clock. The system clock frequency $f_{CLK}$ can be varied under software control by changing the contents of the PLL control register (PLLCON):

$f_{CCO}$ can be changed via the PLLCON bits FSEL(1:0) (see Table 41). The maximum locking time is 10 ms [2].

During the stabilization phase, no time critical routines should be executed.

The system clock frequency $f_{CLK}$ is derived from $f_{CCO}$ under control of the PLLCON bits FSEL(4:0) (see Table 41).

If only FSEL(4:2) is changed but not FSEL(1:0), then it takes about 1us until the new frequency is available.

Changing the system clock frequency has to be done in two steps.

**From HIGH to LOW frequencies:**
First change (FSEL(4:2), then FSEL (1:0).

**From LOW to HIGH frequencies:**
First change only FSEL (1:0) and after a stabilization phase of 10 ms change FSEL (4:2).

### 6.13.3 PLL Control Register – PLLCON

PLLCON is a special function register, which can be read and written by software. It contains the control bits:

- to select one of several system clock frequencies (see Table 41)
- the seconds interrupt flag: SECINT
- to enable the seconds interrupt flag: ENSECI
- the RUN32 bit, which defines if during Power–down Mode the 32kHz oscillator is halted or stays running .

PLLCON is initialized to 0DH upon Reset (RSTIN = '1') or Watchdog Timer Overflow. PLLCON = 0DH corresponds to a system clock frequency of 11.01 MHz.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **PLLCON (F9H)** | RUN32 | ENSECI | SECINT | FSEL.4 | FSEL.3 | FSEL.2 | FSEL.1 | FSEL.0 |

**Figure 44. PLL control register (PLLCON).**

## Table 41. PLLCON

| SYMBOL | BIT | FUNCTION |
|---|---|---|
| RUN32 | PLLCON.7 | RUN32 = 0: The 32 kHz oscillator halts during Power–down.<br>RUN32 = 1: The 32 kHz oscillator stays running during Power–down. |
| ENSECI | PLLCON.6 | Enable the seconds interrupt.<br>(enabling INT1 is also required) |
| SECINT | PLLCON.5 | Seconds interrupt requested by an overflow of the seconds timer (which occurs every second) or via writing a '1' to this bit.<br>SECINT can only be cleared by writing a '0' to this bit . |
| FSEL.4<br>to<br>FSEL.0 | PLLCON.4<br>to<br>PLLCON.0 | System clock frequency in MHz |

| | FSEL[4:2] | | |
|---|---|---|---|
| FSEL[1:0] | 100 | 011 | 010 |
| 11 | 3.93 | 7.86 | 15.73 |
| 10 | 4.72 | 9.44 | |
| 01 | 5.51 | 11.01 | |
| 00 | 6.29 | 12.58 | |

Other combinations, than mentioned above, are reserved and may not be selected .

This allows to generate the standard baudrates 19200, 9600, 4800, 2400 and 1200 Baud , when using the UART and Timer1.

2. This parameter is characterized.

### 6.13.4 Seconds Timer

This counter provides an overflow signal every second, when the 32kHz oscillator is running.

The overflow output sets the interrupt flag SECINT. This interrupt can be disabled/enabled by ENSECI. If SECINT is enabled, it is logically ORed with INT1 (external interrupt 1).

Seconds interrupt and INT1 therefor share the same priority and vector. The software has to check both flags SECINT (PLLCON.5) and IE1 (TCON.3), to distinguish between the two interrupt sources. SECINT can only be cleared via writing a '0' to this bit .

The external interrupts INT0 , INT1 or the seconds interrupt can Wake–up the PLL oscillator and the microcontroller as described in chapter "Wake–up from Power–down Mode".

For a Wake–up via INT1 or seconds interrupt, IE1 must be enabled and level–sensitive.

A further function of the seconds timer is to control the start–up timing of the microcontroller after Reset or after Wake–up from Power–down. It controls the stretching of the reset pulse to the microcontroller and controls releasing the system clock to the microcontroller.

A RSTIN signal of 1us at minimum will reset the microcontroller.

In case of Reset or Wake–up with halted 32kHz oscillator: From RSTIN falling edge or Wake–up interrupt it takes 560ms at maximum for the start–up of the 32kHz oscillator itself and the stabilization of the PLL's.

In case of Wake–up with running 32kHz oscillator: From Wake–up interrupt it takes about 1ms for the stabilization of the PLL's.

After this start–up time, the microcontroller is supplied with the system clock and – in case of a reset – the internally stretched reset signal overlaps about 45us, to guarantee a proper initialization of the microcontroller.

For further information refer to section 6.11 Power reduction modes.

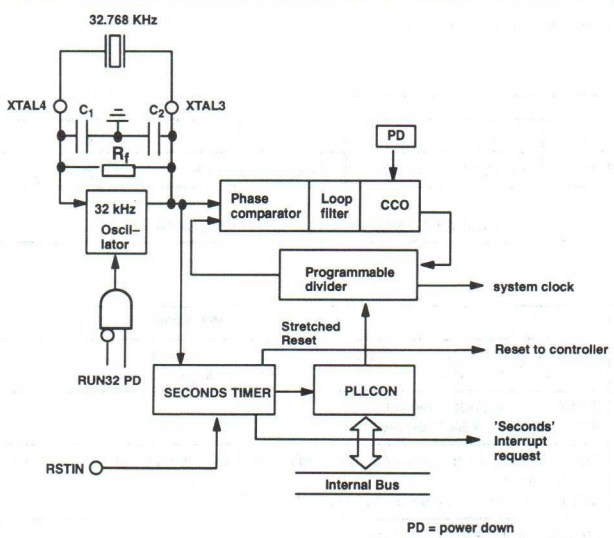



**Figure 45. Block diagram PLL**

## 6.14 Reset Circuitry

The reset input pin RSTIN is connected to a Schmitt trigger for noise reduction (see Figure 46). Is the HF–oscillator selected a Reset is accomplished by holding the RSTIN pin HIGH for at least 2 machine cycles (24 system clock periods). Is the PLL–oscillator selected the RSTIN–pulse must have a width of 1 µs at least, independent of the 32 kHz–oscillator is running or not (see PLL description). The CPU responds by executing an internal reset. The RSTOUT pin represents the signal resetting the CPU and can be used to reset peripheral devices.

The RSTOUT level also could be high due to a Watchdog timer overflow.

The length of the output pulse from T3 is 3 machine cycles. A pulse of such short duration is necessary in order to recover from a processor or system fault as fast as possible.

During Reset, ALE and $\overline{PSEN}$ output a HIGH level. In order to perform a correct reset, this level must not be affected by external elements.

A Reset leaves the internal registers as shown in Table 5.

The internal RAM is not affected by Reset. At power-on, the RAM content is indeterminate.

## 6.15 Power-on Reset

An automatic Reset can be obtained by switching on $V_{DD}$, if the RSTIN pin is connected to $V_{DD}$ via a capacitor, as shown in Figure 47.

If the HF oscillator selected the $V_{DD}$ rise time must not exceed 10 ms and the capacitor should be at least 2.2 µF. The decrease of the RSTIN pin voltage depends on the capacitor and the internal resistor $R_{RST}$. That voltage must remain above the lower threshold for at minimum the HF–oscillator start–up time plus 2 machine cycles.

Is the PLL–oscillator selected a 0.1 µF capacitor is sufficient to obtain an automatic reset.

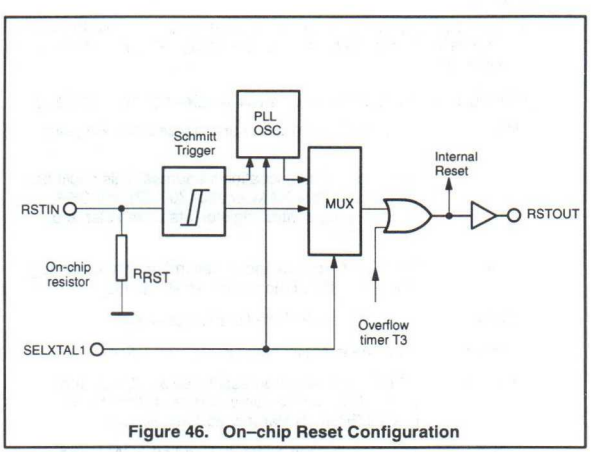



**Figure 46. On–chip Reset Configuration**

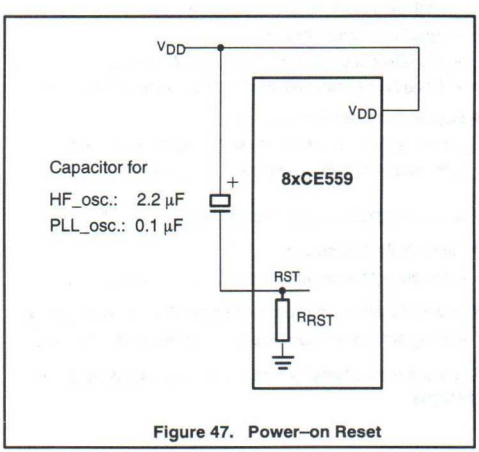



**Figure 47. Power–on Reset**

## 7. INSTRUCTION SET

The P8xCE559 uses the powerful instruction set of the PCB80C51. It consists of 49 single–byte, 45 two–byte and 17 three–byte instructions. Using a 16 MHz quartz, 64 of the instructions are executed in 0.75 µs, 45 in 1,5 µs and the multiply, divide instructions in 3 µs.

A summary of the instruction set is given in Table 43.

The P8xCE559 has additional Special Function Registers to control the on–chip peripherals.

### 7.1 Addressing Modes

Most instructions have a "destination, source" field that specifies the data type, addressing modes and operands involved. For all these instructions, except for MOVs, the destination operand is also the source operand (e.g., ADD A,R7).

There are five kinds of addressing modes:

- Register Addressing
  - R0 – R7 (4 banks)
  - A,B,C (bit), AB (2 bytes), DPTR (double byte)
- Direct Addressing
  - lower 128 bytes of internal Main RAM (including the 4 R0–R7 register banks)
  - Special Function Registers
  - 128 bits in a subset of the internal Main RAM
  - 128 bits in a subset of the Special Function Registers
- Register–Indirect Addressing
  - internal Main RAM (@R0, @R1, @SP [PUSH/POP])
  - internal Auxiliary RAM (@R0, @R1, @DPTR)
  - external Data Memory (@R0, @R1, @DPTR)
- Immediate Addressing
  - Program Memory (in–code 8 bit or 16 bit constant)
- Base–Register–plus Index–Register–Indirect Addressing
  - Program Memory look–up table (@DPTR+A, @PC+A)

The first three addressing modes are usable for destination operands.

#### 7.1.1 80C51 Family Instruction Set

**Table 42. Instruction that affect Flag settings[1]**

| INSTRUCTION | FLAG | | |
|---|---|---|---|
| | C | OV | AC |
| ADD | X | X | X |
| ADDC | X | X | X |
| SUBB | X | X | X |
| MUL | 0 | X | |
| DIV | 0 | X | |
| DA | X | X | |
| RRC | X | | |
| RLC | X | | |
| SETB C | 1 | | |
| CLR C | 0 | | |
| CPL C | X | | |
| ANL C, bit | X | | |
| ANL C,/bit | X | | |
| ANL C, bit | X | | |
| ORL C, bit | X | | |
| MOV C, bit | X | | |
| CJNE | X | | |

**NOTE:**
1. Note that operations on SFR byte address 208 or bit addresses 209-215 (i.e., the PSW or bits in the PSW) will also affect flag settings.

### Notes on instruction set and addressing modes:

| | |
|---|---|
| Rn | Register R7-R0 of the currently selected Register Bank. |
| direct | 8-bit internal data location's address. This could be an Internal Data RAM location (0-127) or a SFR [i.e., I/O port, control register, status register, etc. (128-255)]. |
| @Ri | 8-bit RAM location addressed indirectly through register R1 or R0 of the actual register bank. |
| #data | 8-bit constant included in the instruction. |
| #data 16 | 16-bit constant included in the instruction |
| addr 16 | 16-bit destination address. Used by LCALL and LJMP. A branch can be anywhere within the 64 Kbytes Program Memory address space. |
| addr 11 | 11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 Kbytes page of program memory as the first byte of the following instruction. |
| rel | Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is –128 to +127 bytes relative to first byte of the following instruction. |
| bit | Direct Addressed bit in Internal Data RAM or Special Function Register. |

### Hexadecimal opcode cross-reference to Table 43

| | | |
|---|---|---|
| * | : | 8, 9, A, B, C, D, E. F. |
| ** | : | 11, 31, 51, 71, 91, B1, D1, F1. |
| *** | : | 01, 21, 41, 61, 81, A1, C1, E1. |

**Table 43. 80C51 Instruction Set Summary**

| MNEMONIC | | DESCRIPTION | BYTE | CYCLES | OPCODE (HEX.) |
|---|---|---|---|---|---|
| **ARITHMETIC OPERATIONS** | | | | | |
| ADD | A,Rn | Add register to Accumulator | 1 | 1 | 2* |
| ADD | A,direct | Add direct byte to Accumulator | 2 | 1 | 25 |
| ADD | A,@Ri | Add indirect RAM to Accumulator | 1 | 1 | 26, 27 |
| ADD | A,#data | Add immediate data to Accumulator | 2 | 1 | 24 |
| ADDC | A,Rn | Add register to Accumulator with carry | 1 | 1 | 3* |
| ADDC | A,direct | Add direct byte to Accumulator with carry | 2 | 1 | 35 |
| ADDC | A,@Ri | Add indirect RAM to Accumulator with carry | 1 | 1 | 36, 37 |
| ADDC | A,#data | Add immediate data to ACC with carry | 2 | 1 | 34 |
| SUBB | A,Rn | Subtract Register from ACC with borrow | 1 | 1 | 9* |
| SUBB | A,direct | Subtract direct byte from ACC with borrow | 2 | 1 | 95 |
| SUBB | A,@Ri | Subtract indirect RAM from ACC with borrow | 1 | 1 | 96, 97 |
| SUBB | A,#data | Subtract immediate data from ACC with borrow | 2 | 1 | 94 |
| INC | A | Increment Accumulator | 1 | 1 | 04 |
| INC | Rn | Increment register | 1 | 1 | 0* |
| INC | direct | Increment direct byte | 2 | 1 | 05 |
| INC | @Ri | Increment indirect RAM | 1 | 1 | 06, 07 |
| DEC | A | Decrement Accumulator | 1 | 1 | 14 |
| DEC | Rn | Decrement Register | 1 | 1 | 1* |
| DEC | direct | Decrement direct byte | 2 | 1 | 15 |
| DEC | @Ri | Decrement indirect RAM | 1 | 1 | 16, 17 |
| INC | DPTR | Increment Data Pointer | 1 | 2 | A3 |
| MUL | AB | Multiply A and B | 1 | 4 | A4 |
| DIV | AB | Divide A by B | 1 | 4 | 84 |
| DA | A | Decimal Adjust Accumulator | 1 | 1 | D4 |
| **LOGICAL OPERATIONS** | | | | | |
| ANL | A,Rn | AND Register to Accumulator | 1 | 1 | 5* |
| ANL | A,direct | AND direct byte to Accumulator | 2 | 1 | 55 |
| ANL | A,@Ri | AND indirect RAM to Accumulator | 1 | 1 | 56, 57 |
| ANL | A,#data | AND immediate data to Accumulator | 2 | 1 | 54 |
| ANL | direct,A | AND Accumulator to direct byte | 2 | 1 | 52 |
| ANL | direct,#data | AND immediate data to direct byte | 3 | 2 | 53 |
| ORL | A,Rn | OR register to Accumulator | 1 | 1 | 4* |
| ORL | A,direct | OR direct byte to Accumulator | 2 | 1 | 45 |
| ORL | A,@Ri | OR indirect RAM to Accumulator | 1 | 1 | 46, 47 |
| ORL | A,#data | OR immediate data to Accumulator | 2 | 1 | 44 |
| ORL | direct,A | OR Accumulator to direct byte | 2 | 1 | 42 |
| ORL | direct,#data | OR immediate data to direct byte | 3 | 2 | 43 |
| XRL | A,Rn | Exclusive-OR register to Accumulator | 1 | 1 | 6* |
| XRL | A,direct | Exclusive-OR direct byte to Accumulator | 2 | 1 | 65 |
| XRL | A,@Ri | Exclusive-OR indirect RAM to Accumulator | 1 | 1 | 66, 67 |

**Table 43. 80C51 Instruction Set Summary** (Continued)

| MNEMONIC | | DESCRIPTION | BYTE | CYCLES | OPCODE (HEX.) |
|---|---|---|---|---|---|
| **LOGICAL OPERATIONS** (Continued) | | | | | |
| XRL | A,#data | Exclusive-OR immediate data to Accumulator | 2 | 1 | 64 |
| XRL | direct,A | Exclusive-OR Accumulator to direct byte | 2 | 1 | 62 |
| XRL | direct,#data | Exclusive-OR immediate data to direct byte | 3 | 2 | 63 |
| CLR | A | Clear Accumulator | 1 | 1 | E4 |
| CPL | A | Complement Accumulator | 1 | 1 | F4 |
| RL | A | Rotate Accumulator left | 1 | 1 | 23 |
| RLC | A | Rotate Accumulator left through the carry | 1 | 1 | 33 |
| RR | A | Rotate Accumulator right | 1 | 1 | 03 |
| RRC | A | Rotate Accumulator right through the carry | 1 | 1 | 13 |
| SWAP | A | Swap nibbles within the Accumulator | 1 | 1 | C4 |
| **DATA TRANSFER** | | | | | |
| MOV | A,Rn | Move register to Accumulator | 1 | 1 | E* |
| MOV | A,direct | Move direct byte to Accumulator | 2 | 1 | E5 |
| MOV | A,@Ri | Move indirect RAM to Accumulator | 1 | 1 | E6, E7 |
| MOV | A,#data | Move immediate data to Accumulator | 2 | 1 | 74 |
| MOV | Rn,A | Move Accumulator to register | 1 | 1 | F* |
| MOV | Rn,direct | Move direct byte to register | 2 | 2 | A* |
| MOV | RN,#data | Move immediate data to register | 2 | 1 | 7* |
| MOV | direct,A | Move Accumulator to direct byte | 2 | 1 | F5 |
| MOV | direct,Rn | Move register to direct byte | 2 | 2 | 8* |
| MOV | direct,direct | Move direct byte to direct | 3 | 2 | 85 |
| MOV | direct,@Ri | Move indirect RAM to direct byte | 2 | 2 | 86, 87 |
| MOV | direct,#data | Move immediate data to direct byte | 3 | 2 | 75 |
| MOV | @Ri,A | Move Accumulator to indirect RAM | 1 | 1 | F6, F7 |
| MOV | @Ri,direct | Move direct byte to indirect RAM | 2 | 2 | A6, A7 |
| MOV | @Ri,#data | Move immediate data to indirect RAM | 2 | 1 | 76, 77 |
| MOV | DPTR,#data16 | Load Data Pointer with a 16-bit constant | 3 | 2 | 90 |
| MOVC | A,@A+DPTR | Move Code byte relative to DPTR to ACC | 1 | 2 | 93 |
| MOVC | A,@A+PC | Move Code byte relative to PC to ACC | 1 | 2 | 83 |
| MOVX | A,@Ri | Move AUX–RAM (8-bit addr) to ACC | 1 | 2 | E3, E3 |
| MOVX | A,@DPTR | Move AUX–RAM (16-bit addr) to $A_{CC}$ | 1 | 2 | E0 |
| MOVX | @Ri,A | Move ACC to AUX–RAM (8-bit addr) | 1 | 2 | F2, F3 |
| MOVX | @DPTR,A | Move ACC to AUX–RAM (16-bit addr) | 1 | 2 | F0 |
| PUSH | direct | Push direct byte onto stack | 2 | 2 | C0 |
| POP | direct | Pop direct byte from stack | 2 | 2 | D0 |
| XCH | A,Rn | Exchange register with Accumulator | 1 | 1 | C* |
| XCH | A,direct | Exchange direct byte with Accumulator | 2 | 1 | C5 |
| XCH | A,@Ri | Exchange indirect RAM with Accumulator | 1 | 1 | C6, C7 |
| XCHD | A,@Ri | Exchange low-order digit indirect RAM with ACC | 1 | 1 | D6, D7 |

**Table 43. 80C51 Instruction Set Summary** (Continued)

| MNEMONIC | | DESCRIPTION | BYTE / CYCLES | | OPCODE (HEX.) |
|---|---|---|---|---|---|
| **BOOLEAN VARIABLE MANIPULATION** | | | | | |
| CLR | C | Clear carry | 1 | 1 | C3 |
| CLR | bit | Clear direct bit | 2 | 1 | C2 |
| SETB | C | Set carry | 1 | 1 | D3 |
| SETB | bit | Set direct bit | 2 | 1 | D2 |
| CPL | C | Complement carry | 1 | 1 | B3 |
| CPL | bit | Complement direct bit | 2 | 1 | B2 |
| ANL | C,bit | AND direct bit to carry | 2 | 2 | B2 |
| ANL | C,/bit | AND complement of direct bit to carry | 2 | 2 | B0 |
| ORL | C,bit | OR direct bit to carry | 2 | 2 | 72 |
| ORL | C,/bit | OR complement of direct bit to carry | 2 | 2 | A0 |
| MOV | C,bit | Move direct bit to carry | 2 | 1 | A2 |
| MOV | bit,C | Move carry to direct bit | 2 | 2 | 92 |
| JC | rel | Jump if carry is set | 2 | 2 | 40 |
| JNC | rel | Jump if carry not set | 2 | 2 | 50 |
| JB | rel | Jump if direct bit is set | 2 | 2 | 20 |
| JNB | rel | Jump if direct bit is not set | 2 | 2 | 30 |
| JBC | bit,rel | Jump if direct bit is set and clear bit | 3 | 2 | 10 |
| **PROGRAM BRANCHING** | | | | | |
| ACALL | addr11 | Absolute subroutine call | 2 | 2 | **1addr |
| LCALL | addr16 | Long subroutine call | 3 | 2 | 12 |
| RET | | Return from subroutine | 1 | 2 | 22 |
| RETI | | Return from interrupt | 1 | 2 | 32 |
| AJMP | addr11 | Absolute jump | 2 | 2 | ***1addr |
| LJMP | addr16 | Long jump | 3 | 2 | 02 |
| SJMP | rel | Short jump (relative addr) | 2 | 2 | 80 |
| JMP | @A+DPTR | Jump indirect relative to the DPTR | 1 | 2 | 73 |
| JZ | rel | Jump if Accumulator is zero | 2 | 2 | 60 |
| JNZ | rel | Jump if Accumulator is not zero | 2 | 2 | 70 |
| CJNE | A,direct,rel | Compare direct byte to ACC and jump if not equal | 3 | 2 | B5 |
| CJNE | A,#data,rel | Compare immediate to ACC and jump if not equal | 3 | 2 | B4 |
| CJNE | RN,#data,rel | Compare immediate to register and jump if not equal | 3 | 2 | B* |
| CJNE | @Ri,#data,rel | Compare immediate to indirect and jump if not equal | 3 | 2 | B6, B7 |
| DJNZ | Rn,rel | Decrement register and jump if not zero | 2 | 2 | D* |
| DJNZ | direct,rel | Decrement direct byte and jump if not zero | 3 | 2 | D5 |
| NOP | | No operation | 1 | 1 | 00 |

**NOTE:**

**Note:**

1. All mnemonics copyrighted © Intel Corporation 1980

**Table 44. Instruction map P8xCE559**

**second hexadecimal character of opcode**

| first hexadecimal character of opcode | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | NOP | AJMP addr11 | LJMP addr16 | RR A | INC A | INC dir | INC @ Ri 0 | 1 | INC Rr 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | JBC bit, rel | ACALL addr11 | LCALL addr16 | RRC A | DEC A | DEC dir | DEC @ Ri 0 | 1 | DEC Rr 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 2 | JB bit, rel | AJMP addr11 | RET | RL A | ADD A, #data | ADD A, dir | ADD A, @ Ri 0 | 1 | ADD A, Rr 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 3 | JNB bit, rel | ACALL addr11 | RETI | RLC A | ADDC A, #data | ADDC A, dir | ADDC A, @ Ri 0 | 1 | ADDC A, Rr 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 4 | JC rel | AJMP addr11 | ORL dir, A | ORL dir, #data | ORL A, #data | ORL A, dir | ORL A, @ Ri 0 | 1 | ORL A, Rr 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 5 | JNC rel | ACALL addr11 | ANL dir, A | ANL dir, #data | ANL A, #data | ANL A, dir | ANL A, @ Ri 0 | 1 | ANL A, Rr 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 6 | JZ rel | AJMP addr11 | XRL dir, A | XRL dir, #data | XRL A, #data | XRL A, dir | XRL A, @ Ri 0 | 1 | XRL A, Rr 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 7 | JNZ rel | ACALL addr11 | ORL C, bit | JMP @A+DPTR | MOV A, #data | MOV dir,#data | MOV @ Ri, #data 0 | 1 | MOV Rr, #data 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 8 | SJMP rel | AJMP addr11 | ANL C, bit | MOVC A, @A+PC | DIV AB | MOV dir, dir | MOV dir, @ Ri 0 | 1 | MOV dir, Rr 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 9 | MOV DPTR,#data16 | ACALL addr11 | MOV bit, C | MOVC A,@A+DPTR | SUBB A, #data | SUBB A, dir | SUBB A, @ Ri 0 | 1 | SUBB A, Rr 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| A | ORL C,/bit | AJMP addr11 | MOV C, bit | INC DPTR | MUL AB | | MOV @ Ri, dir 0 | 1 | MOV Rr, dir 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| B | ANL C,/bit | ACALL addr11 | CPL bit | CPL C | CJNE A,#data,rel | CJNE A,dir, rel | CJNE @Ri,#data,rel 0 | 1 | CJNE Rr, #data, rel 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| C | PUSH dir | AJMP addr11 | CLR bit | CLR C | SWAP A | XCH A, dir | XCH A, @ Ri 0 | 1 | XCH A, Rr 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| D | POP dir | ACALL addr11 | SETB bit | SETB C | DA A | DNJZ dir, rel | XCHD A, @ Ri 0 | 1 | DJNZ Rr, rel 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| E | MOVX A, @DPTR | AJMP addr11 | MOVX A, @Ri 0 | 1 | CLR A | MOV A, dir *) | MOV A, @ Ri 0 | 1 | MOV A, Rr 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| F | MOVX @DPTR, A | ACALL addr11 | MOVX A, @Ri, A 0 | 1 | CPL A | MOV dir, A | MOV @ Ri, A 0 | 1 | MOV Rr, A 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |

*) **MOV A, ACC is not a valid instruction**

# 8. FLASH EEPROM

## 8.1 General

- 48 Kbytes electrically erasable internal program memory with Block– and Page-Erase option ("Flash Memory").
- Internal fixed boot ROM.
- Up to 16 Kbytes external program memory in combination with the internal FEEPROM ($\overline{EA}$=1).
- Up to 64 Kbytes external program memory if the internal program memory is switched off ($\overline{EA}$=0).

The FEEPROM can be read and written byte-wise. Full Erase, Block Erase, and Page erase will erase 48 Kbytes, 256 bytes and 32 bytes respectively. In-circuit programming and out-of-circuit programming is possible. On-chip erase and write timing generation and on chip high voltage generation contribute to a user friendly interface.

## 8.2 Features

- Read:
  byte-wise
- Write:
  byte-wise within 2.5 ms.
  (previously erased by a page, block or full erase).
- Erase:
  Page Erase (32 bytes) within 5 ms.
  Block Erase (256 bytes) within 5 ms.
  Full Erase (32 Kbytes) within 5 ms.
  Erased bytes contain FFH.
- Endurance:
  100 erase and write cycles each byte at $T_{amb}$ = 22°C
- Retention:
  10 years
- Out-of-circuit programming:
  Parallel programming with 87C51 compatible hardware Interface to programmer.
- In-circuit programming:
  Serial programming via RS232 interface under boot ROM program control. Auto baud rate selection.
  Intel Hex Object file Format.
  The user program can call routines in the boot ROM for erase, write and verify of the FEEPROM.
- High programming voltage generation: on chip
- Zero point on-chip oscillator and timer to generate the write and erase time durations.
- Programmable security for the code in the FEEPROM to prevent software piracy. The Security Byte is located in the highest address (0BFFFH) of the FEEPROM.
- Supply voltage monitoring circuit on–chip to prevent loss of information in the FEEPROM during power–on and power–off.

## 8.3 Memory Map

Figure 48 shows the memory map of the user program memory and the boot ROM. They are located in the same program address space. Two bits UBS1 and UBS0 of the FEEPROM control special function register FMCON select between the two memory blocks.

**User program memory selection**
If UBS1 and UBS0 are both 0, then the user program memory is mapped into the 64 K program memory space and the boot ROM cannot be selected. This is the situation after a reset when $\overline{PSEN}$ and ALE have not been pulled down during reset. Program execution starts at 0000H in the internal FEEPROM or in the external program memory dependent on the level of $\overline{EA}$ during reset.

**Boot ROM selection**
After a reset program execution starts in the boot ROM when during reset $\overline{PSEN}$ and $\overline{EA}$ are pulled down while ALE stay high. The boot ROM size is 1 Kbyte. Besides the serial in–circuit programming routine the boot ROM contains the routines for erase, write and verify of the FEEPROM, which can be called by the user program (LCALL to the address space between 63 K and 64 K).

**Switching between user program memory and boot ROM**
Switching between user program memory (internal or external) and boot ROM is possible if UBS1 and UBS0 are 0,1. Then in the program memory address space between 0 and 63k the user program memory is selected and in the memory space between 63 K and 64 K the boot ROM is selected.

To switch from user program memory to boot ROM first UBS0 must be set (UBS1 stay 0) and a jump or call instruction to a location >63 K must be executed.

At the moment of crossing the 63 K address border by a return instruction the switching from boot ROM to user memory (internal or external) is performed. After crossing the 63 K address border UBS1 and UBS0 are cleared and the total 64 K memory space is mapped as user program memory. By clearing UBS1 and UBS0, no special requirements to the user program are necessary to do that after a read or erase or write routine.

A small restriction for memory switching is that no memory switching is allowed from or to the address space between 63 K and 64 K of the user program memory because the UBS bits must stay 0 in this range. This restriction can be avoided if the memory switching is always done by a subroutine in the address range between 0 and 63 K.

**Description**
The user program code in the FEEPROM is executed as in the standard 80C51 microcontroller. Erase and write cycles in the FEEPROM are always performed under control of the boot program in the boot ROM in the address space between 63 K and 64 K. Address and data parameters are passed via DPTR and accumulator A respectively. During an erase or write cycle in the FEEPROM no other access or program execution in the FEEPROM is possible. All interrupts must be disabled when the user program calls a user routine in the boot ROM.

The boot routine for serial programming takes care of addressing, data transfer, verify, high voltage control, error message and return to the user program memory. It also contains the serial communication routine.

The FEEPROM control register FMCON is a special function register. It contains the control bits for verify, write, erase and boot ROM switching.

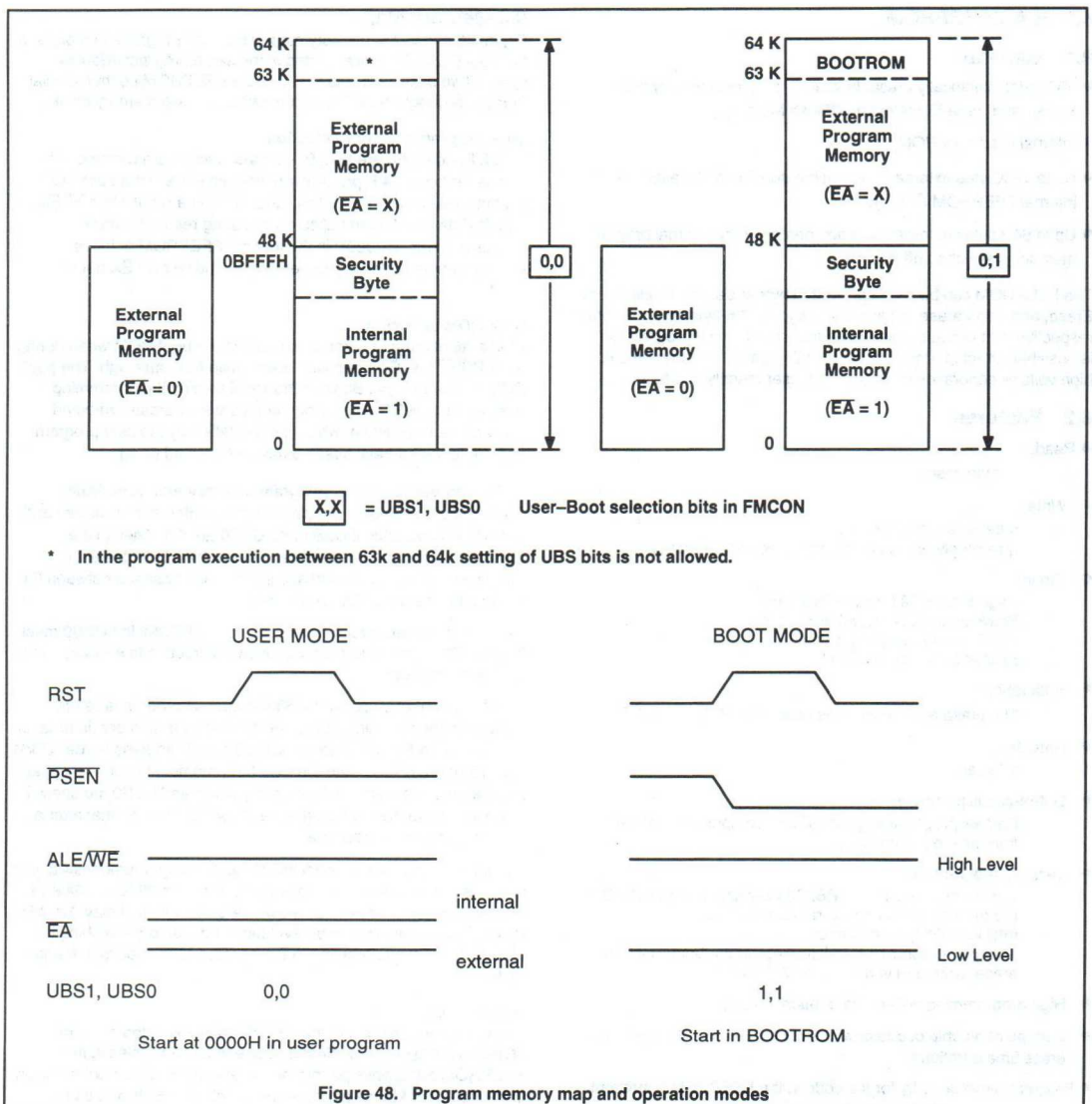



**Figure 48. Program memory map and operation modes**

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **FMCON (FB)** | UBS1 | UBS0 | HV | – [1] | FCB3 | FCB2 | FCB1 | FCB0 |

**Figure 49. FEEPROM control register.**

**NOTE:**
1. Reserved for future use; a write operation must write "0" to the location

## Table 45. Description of FMCON bits

| UBS1 | UBS0 | User – Boot selection bits |
|---|---|---|
| 0 | 0 | User memory mapped from 0 to 64 K. |
| 0 | 1 | User memory mapped from 0 to 63 K.<br>Boot ROM mapped from 63 K to 64 K. |
| 1 | 0 | User memory mapped from 0 to 63 K, but UBS1 bit cleared by hardware in this user address range.<br>Boot ROM mapped from 63 K to 64 K. User software should not write "1" UBS1. |
| 1 | 1 | Boot ROM mapped from 0 to 64 K. User software should not write "1" UBS1. |
| | | |
| **HV** | | **High voltage indication bit.** Read only.<br>Is "1" as long as the high voltage for an erase or write operation is present. |

| FCB3 | FCB2 | FCB1 | FCB0 | Function Code Bits |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | Value after Reset. |
| 0 | 1 | 0 | 1 | Byte Write or byte read (verify) |
| 1 | 1 | 0 | 0 | Page Erase (32 bytes boundaries). |
| 0 | 0 | 1 | 1 | Block Erase (256 bytes boundaries). |
| 1 | 0 | 1 | 0 | Full Erase (48 Kbytes). |

The four FCB bits are write protected if the security feature is activated. Then only instructions in the internal program memory (FEEPROM) are able to write FCB (3–0), boot ROM and external program memory instructions cannot change FCB (3–0) except the full erase code can be loaded.

The duration of a write or erase operation is determined by the FEEPROM timer. This timer includes a zero point RC oscillator and cannot be controlled by software.

For calling a user routine in the boot ROM first all interrupts must be disabled and the DPTR and A have to be loaded with the desired values. After setting UBS0 = 1 and UBS1 = 0 and selecting the function via FCB–bits the respective user routine has to be called.

The table below lists the boot ROM user routines, which can be called by the user program. The content of FMCON, A and DPTR before the call is described by "(IN)" and the contents after the return is described by "(OUT)". The boot ROM user routines do not change other registers or Data memory.

| BOOT-ROM ROUTINE | CALL ADDRESS | FMCON (IN) | FMCON (OUT) | ACC (IN) | ACC (OUT) | DPTR (IN) | DPTR (OUT) |
|---|---|---|---|---|---|---|---|
| BYTE_READ | FFBAH | 45H | 15H | XXH | BYTE | BYTE ADDRESS | BYTE ADDRESS |
| BYTE_WRITE | FFADH | 45H | 15H | BYTE | BYTE (V) | BYTE ADDRESS | BYTE ADDRESS |
| PAGE_ERASE | FFAAH | 4CH | 1CH | XXH | 08H | PAGE ADDRESS 1) | PAGE ADDRESS 2) |
| BLOCK_ERASE | FFA5H | 43H | 13H | XXH | 02H | BLOCK ADDRESS 3) | BLOCK ADDRESS 4) |
| FULL_ERASE | FFA0H | 4AH | 1AH | XXH | 0AH | XXXXH | 0018H |

X = don't care or not defined
V = verified byte (read back)
1) = 5 LSB's of DPTR are don't care
2) = 5 LSB's of DPTR are "0"
3) = 8 LSB's of DPTR are don't care
4) = 8 LSB's of DPTR contain 08H.

Example of user software (internal or external) that calls the **Page Erase** routine in the boot ROM to erase a page in the FEEPROM (32 bytes) starting at address location 1260H.

```
CLR EA                  ; Disable all interrupts
MOV DPTR, # 1260H       ; Load page–address
MOV FMCON, # 4CH        ; Load Page–Erase code
LCALL 0FFAAH            ; Call Page–Erase routine in boot ROM
                        ; (inherent delay 5 ms)
MOV FMCON, #00H         ; Clear FMCON for security
SETB EA                 ; Enable interrupts again
```

Example of user software (internal or external) that calls the **Byte–Write** routine in the boot ROM to write the content of R5 into the FEEPROM address location 1263H.

```
CLR EA                  ; Disable all interrupts
MOV DPTR, # 1263H       ; Load byte address
MOV A, R5               ; Load byte to be written
MOV FMCON, # 45H        ; Load byte–write code
LCALL 0FFADH            ; Call byte–write routine in boot ROM
                        ; (inherent delay 2.5 ms)
MOV FMCON, #00H         ; Clear FMCON for security
SETB EA                 ; Enable interrupts again
XRL A, R5               ; Compare the "read–back" byte
JNZ ERROR               ; Jump if verify error
```

## 8.4 Security

The security feature protects against software piracy and prevents that the content of the FEEPROM can be read undesirable. The Security Byte is located in the highest address location 0BFFFH of the FEEPROM.

The Security Byte should be 50H to activate and 00H or FFH to deactivate the security feature. This security code is chosen in such a way that single bit failures will not deactivate the security feature.

If the security feature is deactivated, then there are no access restrictions to the FEEPROM.

If the security feature is activated, then the external program memory has no access to the FEEPROM with the MOVC instructions. Also bits FCB (3–0) of FMCON cannot be written by external program code or boot ROM code. This prevents in–circuit programming and verification. Only the Full Erase code can be written to FCB (0–3) of FMCON. Note that for the internal program code no restrictions exist if the security feature is activated. At the end of a full erase operation the security feature is deactivated. Also parallel programming and verify is inhibited if the security feature is activated, only a full erase is possible.

Note that the security mode does not change immediately when the security code is written into the security byte 0BFFFH, but after a reset or power–on. This allows the verification of the loaded code in the FEEPROM, including the Security Byte.

## 8.5 Parallel Programming

Unlike standard EPROM programming, no high programming supply voltage must be applied to the $\overline{EA}$ pin and only one programming pulse must be applied to the ALE/$\overline{WE}$ pin. The parallel programming mode is entered with the steady signals RST=1, PSEN=0, $\overline{EA}$=1 and SELXTAL1 = 1. The XTAL1,2 clock must have a frequency between 4 and 6MHz. The following table shows the logic levels for programming, erasing, verifying and read signature.

| MODE | ALE/$\overline{WE}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|
| Full erase | ⊓ | 1 | 1 | 0 | 1 |
| Program FEEPROM | ⊓ | 1 | 0 | 1 | 1 |
| Verify FEEPROM | 1 | 0 | 0 | 1 | 1 |
| Read signature | 1 | 0 | 0 | 0 | 0 |

ALE/$\overline{WE}$ Write Enable signal (program/erase), active low
P2.6, P2.7, P3.6, P3.7 control signals

**Data and address bits:**

P0.0 – P0.7 : D0 – D7 Program data input / verify or read data output
P1.0 – P1.7 : A0 – A7 Input low order address bits.
P2.0 – P2.5, P3.4, P3.5 : A8 – A15 Input high order address bits.

The P89CE559 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. These bytes are read by the same procedure as for a normal verification of locations 30H and 31H, except that P3.6 and P3.7 need to be pulled to LOW.

| ADDRESS | CONTENT | MEANING |
|---|---|---|
| 30H | 15H | Philips |
| 31H | B7H | P89CE559 |

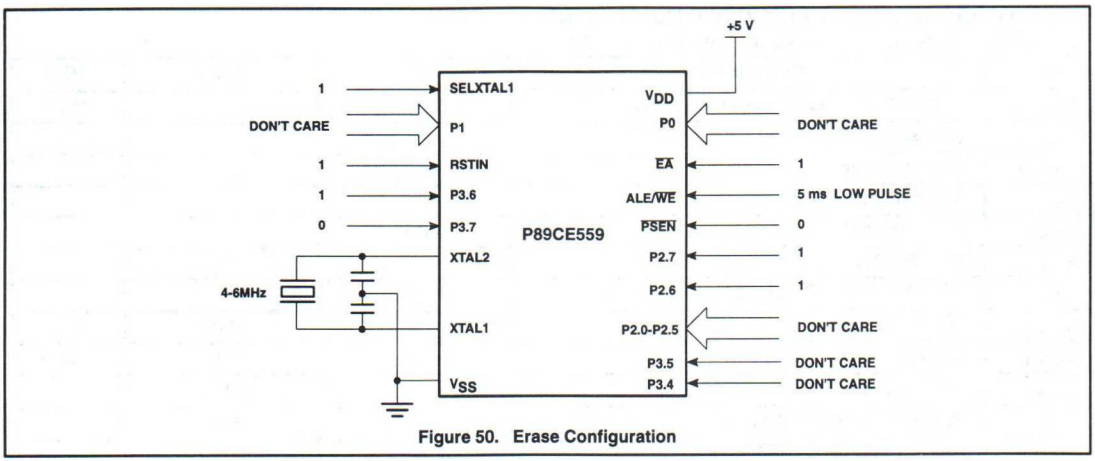



Figure 50. Erase Configuration

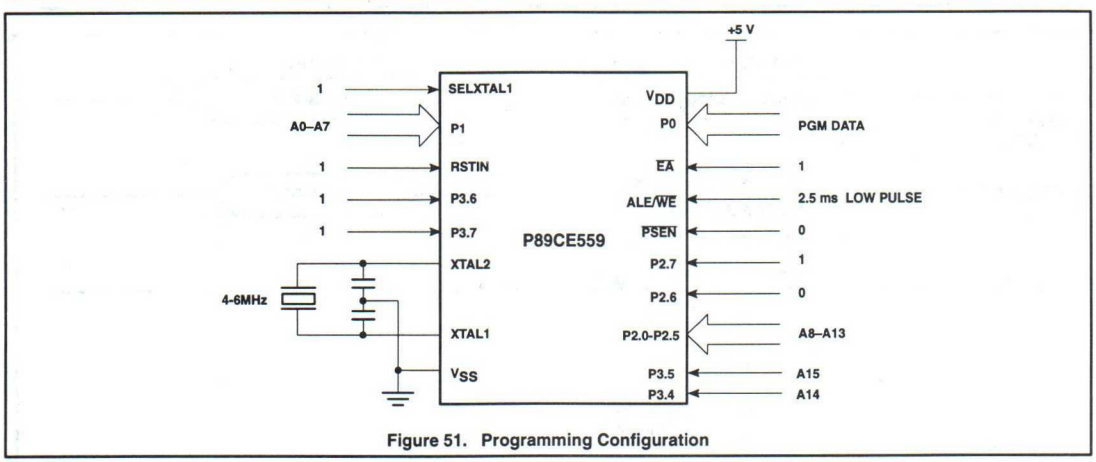



Figure 51. Programming Configuration

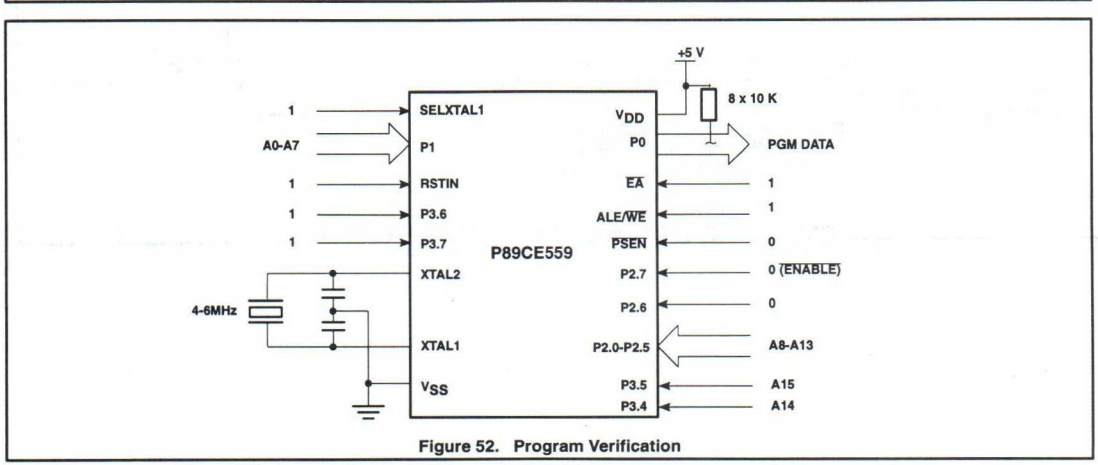
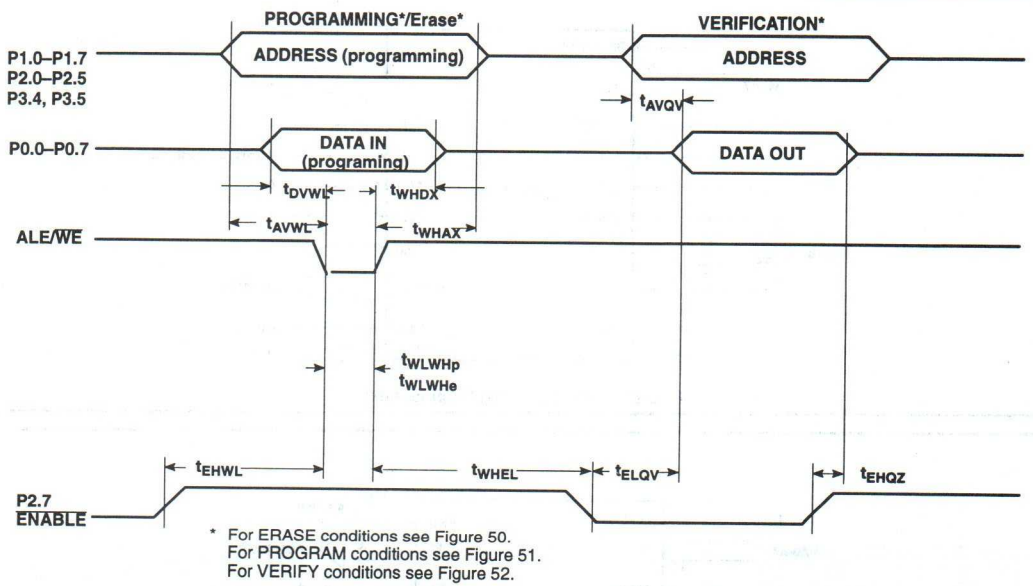



Figure 52. Program Verification

## FEEPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

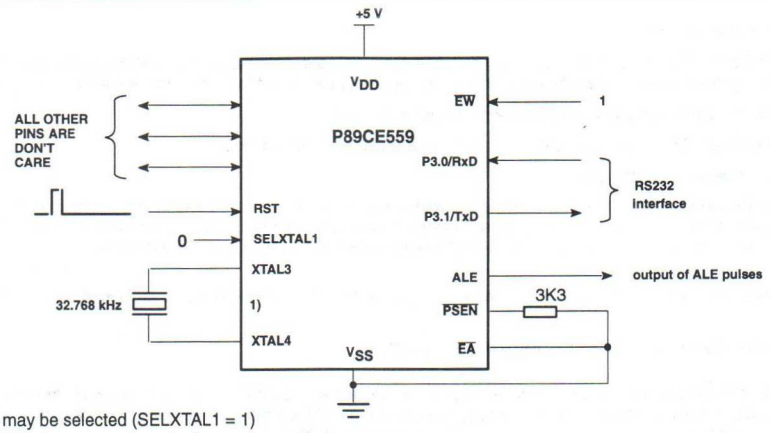
$T_{amb}$ = –40 °C to +85 °C, $V_{DD}$ = 5 V ± 10%, $V_{SS}$ = 0 V (see Figure 53)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $1/t_{CLK}$ | System clock frequency (standard oscillator) | 4 | 6 | MHz |
| $t_{AVWL}$ | Address setup to $\overline{WE}$ LOW | $48t_{CLK}$ | – | |
| $t_{WHAX}$ | Address hold after $\overline{WE}$ HIGH | $48t_{CLK}$ | – | |
| $t_{DVWL}$ | Data setup to $\overline{WE}$ LOW | $48t_{CLK}$ | – | |
| $t_{WHDX}$ | Data hold after $\overline{WE}$ HIGH | $48t_{CLK}$ | – | |
| $t_{EHWL}$ | P2.7 ($\overline{ENABLE}$) HIGH to $\overline{WE}$ LOW | $48t_{CLK}$ | – | |
| $t_{WHEL}$ | $\overline{WE}$ HIGH to P2.7 ($\overline{ENABLE}$) LOW | $48t_{CLK}$ | – | |
| $t_{WLWHp}$ | $\overline{WE}$ width (programming) | 2.25 | 2.75 | ms |
| $t_{WLWHe}$ | $\overline{WE}$ width (erase) | 4.5 | 5.5 | ms |
| $t_{AVQV}$ | Address to data valid | – | $48t_{CLK}$ | |
| $t_{ELQV}$ | P2.7 ($\overline{ENABLE}$) Low to data valid | – | $48t_{CLK}$ | |
| $t_{EHQZ}$ | Data float after P2.7 ($\overline{ENABLE}$) HIGH | 0 | $48t_{CLK}$ | |




* For ERASE conditions see Figure 50.
For PROGRAM conditions see Figure 51.
For VERIFY conditions see Figure 52.

**Figure 53. FEEPROM Programming/Erase and Verification Waveforms**




1) Alternative XTAL1, 2 may be selected (SELXTAL1 = 1)

**Figure 54. Serial programming (boot mode) Configuration**

## 8.6 Serial Programming of FEEPROM

Serial in–circuit programming (boot–mode) is entered if during and after RESET $\overline{PSEN}$ and $\overline{EA}$ are pulled down, $\overline{PSEN}$ via a resistor of 3.3 k Ohm to VSS. The two UBS bits are set to 1 by hardware and program execution starts at 0000H of the boot ROM. P3.0 (RXD) and P3.1 (TXD) form the serial RS232 interface. A baud rate of 4800 or 9600 Baud is possible, if the PLL oscillator is selected. The receive and transmit channel have the same baudrate. The format is: Startbit, 8 data bits (last bit always 0), no parity bit and at least one stopbit. The boot routine inputs the Intel Hex Object Format. The baud rate will be selected automatically after reception of the first character (:) of the object file. No other characters are allowed to preceed the first (:) character. Programming is only started if the first received record has the right type indication (TT). If the security feature is activated (contents of the security byte = 50H) then the programming starts with a Full Erase, otherwise only the addressed page(s) will be erased and the not altered bytes are rewritten. During the erase or write operation the next string of bytes can be received. Xon and Xoff handshake codes are used to control the serial transfer. At the end of the programming a message that indicates a successful or not successful programming, will be returned over the RS232 interface channel. If the programming was successful then the user program can be started up at 0000H in FEEPROM by a reset for user mode ($\overline{EA}$ = high, $\overline{PSEN}$ not affected). If the programming was not successful the boot program halts and a retry can be started by a reset for the boot mode.

## 8.7 Boot Routine

The boot routine transmits the next "one ASCII character" messages via the RS232 interface:

- " . " After each record type TT = 00H indication in the HEX file.
- " X " Checksum error of a record in the HEX file detected.
- " Y " Wrong record type received
- " Z " Buffer overflow error (Check Xon/Xoff of terminal)
- " R " Verification error (of last written byte)
- " V " End record received and programming of FEEPROM was successful

No messages are transmitted if the baud rate of the first character (:) can not be detected.

The boot routine can also be started by the internal or external user program (LJMP FC07H). FMCON must be loaded previously with 40H. Interrupt registers, stack pointer, Timer 0, UART, P3.0 and P3.1 must be in the reset state. $\overline{EA}$ and $\overline{PSEN}$ must not be affected. A reset is needed to restart the user program after programming.

The following baudrates will be detected automatically within the specified µC clock range in MHz.

| Baudrate | $f_{CLK}$ (min) | $f_{CLK}$ (max) |
|---|---|---|
| 1200 | 1 [1)] | 3.6 |
| 2400 | 2 [1)] | 7.3 |
| 4800 | 4 | 14.7 |
| 9600 | 7.9 | 29.5 [1)] |
| 19200 | 15.7 | 59 [1)] |

**NOTE:**
1. Value outside the specified clock range

Note that the boot routines can (re) program any number of bytes from 1 byte to 48 Kbytes, independent in which order or at which location, but if the security feature is activated, a full erase is performed and all not programmed bytes become FFH.

**Definitions:**

: – Record start character

**BC** – Byte Count. The hexadecimal number of data bytes in the record. This may theoretically be any number from 0 to 255, although many assemblers prefer to deal with 16 data bytes per record (as shown in the example below).

**AAAA** – Load address in hexadecimal of first data byte in this record.

**TT** – Record type. The record type is 00 for data records and 01 for the end record.

**HH** – One hexadecimal data byte.

**CC** – Record checksum. This is the 2's complement of the summation of all of the bytes in the record from the byte count through the last data byte. While the summation is calculated, it is always truncated to a one byte result. Thus, if all of the bytes in the record are summed, including the checksum itself, the result will always be 00 if the record is valid.

Construction of data records (using the notation defined above, each letter corresponds to one hexadecimal digit in ASCII representation) is as follows:

: BCAAAATTHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHCC

The last record in a file is the end record and contains no data. Usually the end record will appear as shown in the first example below. However, in some cases a 16 bit checksum of all of the data bytes in the entire file may be inserted in the address field of the end record. This checksum would correspond to one generated by an EPROM programmer during file load, and its inclusion does not violate the rules for this format. This is shown in the second example.

:00000001FF

:00B12C0122

Successive hex records need not appear in sequential address order . For instance, a record for address 0000H might appear after a record for address 7FE0H. All of the bytes in a single record, however, must be in sequence. Any characters that appear outside of a record (i.e. after a checksum, but before the next ":") will be ignored, if present.

An example of a valid hex file follows:

:10010000C2F0E53030E704F404D2F08531F030F786

:100110000763F0FF05F0B2F0A430F00A63F0FFF4DB

:0C0120002401500205F085F032F5332276

:00000001FF

## 9. ABSOLUTE MAXIMUM RATINGS

### ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage on $V_{DD}$ to $V_{SS}$ and SCL, SDA to $V_{SS}$ | –0.5 to +6.5 | V |
| Input / output current on any I/O pin | 10 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.0 | W |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions are taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## 10. DC CHARACTERISTICS

### DC ELECTRICAL CHARACTERISTICS

$V_{DD}$ = 5V (± 10%), $V_{SS}$ = 0V, $T_{amb}$ = 0°C to +70°C (P8xCE559EBx). All voltages with respect to $V_{SS}$ unless otherwise specified.

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|
| $V_{DD}$ | Supply voltage | | 4.5 | 5.5 | V |
| $I_{DD}$ | Supply current operating : | See notes 1 and 2 | | | |
| | P89CE559 | $f_{CLK}$ = 16MHz | | 50 | mA |
| | P83CE559 | $V_{DD}$ = 5.5 V | | 40 | mA |
| $I_{ID}$ | Supply current Idle Mode : | See notes 1 and 3 | | | |
| | P89CE559 | $f_{CLK}$ = 16MHz | | 15 | mA |
| | P83CE559 | $V_{DD}$ = 5.5 V | | 12 | mA |
| $I_{PD}$ | Supply current Power-down mode | See note 4<br>2 V < $V_{PD}$ < $V_{DDmax}$ | | 100 | µA |
| | Supply current Power-down mode:<br>32 kHz / PLL operation | See note 17<br>$V_{DD}$ = 5.5 V | | 100 | µA |
| **Inputs** | | | | | |
| $V_{IL}$ | Input LOW voltage, except $\overline{EA}$, SCL, SDA | | –0.5 | $0.2V_{DD}$–0.1 | V |
| $V_{IL1}$ | Input LOW voltage to $\overline{EA}$ | | –0.5 | $0.2V_{DD}$–0.3 | V |
| $V_{IL2}$ | Input LOW voltage to SCL, SDA [5] | | –0.5 | $0.3V_{DD}$ | V |
| $V_{IH}$ | Input HIGH voltage, except XTAL1, RSTIN, SCL, SDA, ADEXS | | $0.2V_{DD}$+0.9 | $V_{DD}$+0.5 | V |
| $V_{IH1}$ | Input HIGH voltage, XTAL1, RSTIN, ADEXS | | $0.7V_{DD}$ | $V_{DD}$+0.5 | V |
| $V_{IH2}$ | Input HIGH voltage, SCL, SDA [5] | | $0.7V_{DD}$ | 6.0 | V |
| $I_{IL}$ | Input current LOW level, Ports 1, 2, 3, 4 | $V_{IN}$ = 0.45 V | | –50 | µA |
| $I_{TL}$ | Transition current HIGH to LOW, Ports 1, 2, 3, 4 | See note 6 | | –650 | µA |
| $\pm I_{LI1}$ | Input leakage current, Port 0, $\overline{EA}$, ADEXS, $\overline{EW}$, SELXTAL1 | 0.45 V < $V_I$ < $V_{DD}$ | | 10 | µA |
| $\pm I_{LI2}$ | Input leakage current, SCL, SDA | 0 V < $V_I$ < 6 V<br>0 V < $V_{DD}$ < 5.5 V | | 10 | µA |
| $\pm I_{LI3}$ | Input leakage current, Port 5 | 0.45 V < $V_I$ < $V_{DD}$ | | 1 | µA |
| **Outputs** | | | | | |
| $V_{OL}$ | Output low voltage, Ports 1, 2, 3, 4 | $I_{OL}$ = 1.6mA[7] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, Port 0, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$, RSTOUT | $I_{OL}$ = 3.2mA[7] | | 0.45 | V |
| $V_{OL2}$ | Output low voltage, SCL, SDA | $I_{OL}$ = 3.0mA[7, 19] | | 0.4 | V |
| | | $I_{OL}$ = 6.0mA[7, 19] | | 0.6 | |
| $V_{OH}$ | Output high voltage, Ports 1, 2, 3, 4 | $V_{DD}$ = 5 V ± 10%<br>$-I_{OH}$ = 60µA<br>$-I_{OH}$ = 25µA<br>$-I_{OH}$ = 10µA | <br>2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | <br>V<br>V<br>V |
| $V_{OH1}$ | Output high voltage (Port 0 in external bus mode, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$, RSTOUT) [8] | $V_{DD}$ = 5 V ± 10%<br>$-I_{OH}$ = 800µA<br>$-I_{OH}$ = 300µA<br>$-I_{OH}$ = 80µA | <br>2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | <br>V<br>V<br>V |
| $V_{HYS}$ | Hysteresis of Schmitt Trigger inputs SCL, SDA (Fast-mode) | | $0.05V_{DD}$ [20] | | V |

**NOTES: See Page 3-697.**

## DC ELECTRICAL CHARACTERISTICS (Continued)

$V_{DD}$ = 5 V (± 10%), $V_{SS}$ = 0 V, $T_{amb}$ = –40°C to +85°C (P8xCE559EFx). DC parameters not included here are the same as in the P8xCE559EBx, DC electrical characteristics.
All voltages with respect to $V_{SS}$ unless otherwise specified.

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $R_{RST}$ | Internal reset pull-down resistor | | 50 | 150 | kΩ |
| $C_{IO}$ | Pin capacitance | Test freq = 1MHz, $T_{amb}$ = 25 °C | | 10 | pF |
| **Inputs** | | | | | |
| $V_{IL}$ | Input LOW voltage, except $\overline{EA}$, SCL, SDA | | –0.5 | $0.2V_{DD}$–0.15 | V |
| $V_{IL1}$ | Input LOW voltage to $\overline{EA}$ | | –0.5 | $0.2V_{DD}$–0.35 | V |
| $V_{IH}$ | Input HIGH voltage, except XTAL1, RSTIN, SCL, SDA, ADEXS | | $0.2V_{DD}$+1.0 | $V_{DD}$+0.5 | V |
| $V_{IH1}$ | Input HIGH voltage, XTAL1, RSTIN, ADEXS | | $0.7V_{DD}$+0.1 | $V_{DD}$+0.5 | V |
| $I_{IL}$ | Input current LOW level, Ports 1, 2, 3, 4 | $V_{IN}$ = 0.45 V | | –75 | µA |
| $I_{TL}$ | Transition current HIGH to LOW, Ports 1, 2, 3, 4 | See note 6 | | –750 | µA |

**NOTES: See Page 3-697.**

## DC ELECTRICAL CHARACTERISTICS ANALOG

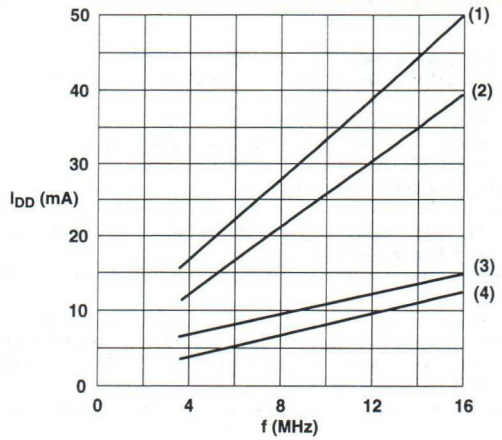
$AV_{DD}$ = 5 V (± 10%), $AV_{SS}$ = 0 V, $T_{amb}$ = 0 °C to +70 °C (P8xCE559EBx).
$AV_{DD}$ = 5 V (± 10%), $AV_{SS}$ = 0 V, $T_{amb}$ = –40 °C to +85 °C (P8xCE559EFx).
All voltages with respect to $V_{SS}$ unless otherwise specified.

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $AV_{DD}$ | Analog supply voltage | $AV_{DD} = V_{DD} \pm 0.2$ V | 4.5 | 5.5 | V |
| $AI_{DD}$ | Analog supply current operating | Port 5 = 0 to $AV_{DD}$ see notes 1 and 2 | | 1.2 | mA |
| | Analog supply current operating: 32 kHz/PLL operation | Port 5 = 0 to $AV_{DD}$ see note 17, 18 | | 7.2 | mA |
| $AI_{ID}$ | Analog supply current Idle Mode | see notes 1 and 3 | | 70 | µA |
| | Analog supply current Idle Mode: 32 kHz/PLL operation | see note 17 | | 6.0 | mA |
| $AI_{PD}$ | Supply current Power-down mode | $2 V < V_{PD} < V_{DDmax}$ see note 4 | | 50 | µA |
| | Supply current Power-down mode: 32 kHz/PLL operation | $V_{DD}$ = 5.5V see note 17 | | 200 | µA |
| **Analog Inputs** | | | | | |
| $AV_{IN}$ | Analog input voltage | | $AV_{SS}$–0.2 | $AV_{DD}$+0.2 | V |
| $AV_{REF}$ | Reference voltage: $AV_{REF-}$ $AV_{REF+}$ | | $AV_{SS}$–0.2 | $AV_{DD}$+0.2 | V V |
| $R_{REF}$ | Resistance between $AV_{REF+}$ and $AV_{REF-}$ | | 10 | 50 | kΩ |
| $C_{IA}$ | Analog input capacitance | | | 15 | pF |
| $DL_e$ | Differential non-linearity [9, 10, 11] | | | ±1 | LSB |
| $IL_e$ | Integral non-linearity [9, 12] | | | ±2 | LSB |
| $OS_e$ | Offset error [9, 13] | | | ±2 | LSB |
| $G_e$ | Gain error [9, 14] | | | ±0.4 | % |
| $A_e$ | Absolute voltage error [9, 15] | | | ±3 | LSB |
| $M_{CTC}$ | Channel to channel matching | | | ±1 | LSB |
| $C_t$ | Crosstalk between inputs of port 5 [16] | 0–100kHz | | –60 | dB |

**NOTES: See Page 3-697.**

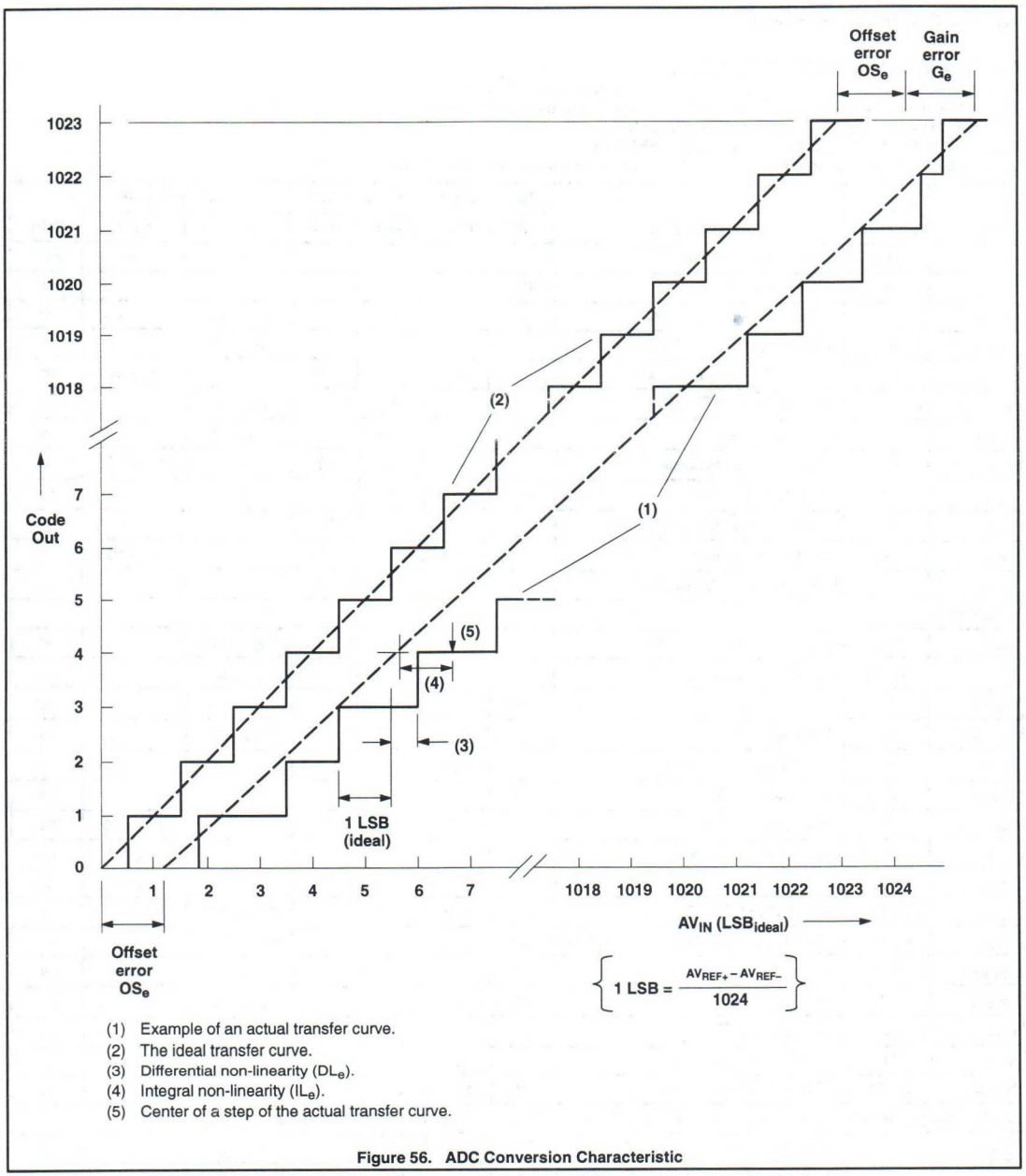
**NOTES FOR DC ELECTRICAL CHARACTERISTICS:**

1. See Figures 55 and 57 through 59 for $I_{DD}$ test conditions.
2. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5$ V; $V_{IH} = V_{DD} - 0.5$ V; XTAL2, XTAL3 not connected; $\overline{EA}$ = RSTIN = Port 0 = $\overline{EW}$ = SCL = SDA = SELXTAL1 = $V_{DD}$; ADEXS = XTAL4 = $V_{SS}$.
3. The Idle Mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5$ V; $V_{IH} = V_{DD} - 0.5$ V; XTAL2, XTAL3 not connected; Port 0 = $\overline{EW}$ = SCL = SDA = SELXTAL 1 = $V_{DD}$; $\overline{EA}$ = RSTIN = ADEXS = XTAL4 = $V_{SS}$.
4. The Power-down current is measured with all output pins disconnected; XTAL2 not connected; Port 0 = $\overline{EW}$ = SCL = SDA = SELXTAL 1 = $V_{DD}$; $\overline{EA}$ = RSTIN = ADEXS = XTAL1 = XTAL4 = $V_{SS}$.
5. The input threshold voltage of SCL and SDA (SIO1) meets the $I^2C$ specification, so an input voltage below 0.3 $V_{DD}$ will be recognized as a logic 0 while an input voltage above 0.7 $V_{DD}$ will be recognized as a logic 1.
6. Pins of ports 1, 2, 3, and 4 source a transition current when they are being externally driven from HIGH to LOW. The transition current reaches its maximum value when $V_{IN}$ is approximately 2 V.
7. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$ of ALE and ports 1, 3 and 4. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8 V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.
8. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{DD}$ specification when the address bits are stabilizing.
9. Conditions: $AV_{REF-} = 0$ V; $AV_{DD} = 5.0$ V, $AV_{REF+} = 5.12$ V. $V_{DD} = 5.0$ V, $V_{SS} = 0$ V, ADC is monotonic with no missing codes. Measurement by continuous conversion of $AV_{IN} = -20mV$ to 5.12 V in steps of 0.5mV, derivating parameters from collected conversion results of ADC. ADC prescaler programmed according to the actual oscillator frequency, resulting in a conversion time within the specified range for $t_{conv}$ (15µs ... 50µs).
10. The differential non-linearity ($DL_e$) is the difference between the actual step width and the ideal step width.
11. The ADC is monotonic; there are no missing codes.
12. The integral non-linearity ($IL_e$) is the peak difference between the center of the steps of the actual and the ideal transfer curve after appropriate adjustment of gain and offset error.
13. The offset error ($OS_e$) is the absolute difference between the straight line which fits the actual transfer curve (after removing gain error), and a straight line which fits the ideal transfer curve. The offset error is constant at every point of the actual transfer curve.
14. The gain error ($G_e$) is the relative difference in percent between the straight line fitting the actual transfer curve (after removing offset error), and the straight line which fits the ideal transfer curve. Gain error is constant at every point on the transfer curve.
15. The absolute voltage error ($A_e$) is the maximum difference between the center of the steps of the actual transfer curve of the non-calibrated ADC and the ideal transfer curve.
16. This should be considered when both analog and digital signals are simultaneously input to port 5.
17. The supply current with 32 kHz oscillator running and PLL operation (SELXTAL1 = 0) is measured with all output pins disconnected; XTAL4 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5$ V; $V_{IH} = V_{DD} - 0.5$ V; XTAL2 not connected; Port 0 = $\overline{EW}$ = SCL = SDA = $V_{DD}$; $\overline{EA}$ = RSTIN = ADEXS = SELXTAL 1 = XTAL1 = $V_{SS}$.
18. Not 100% tested; sum of $A_{IID}$ (PLL) and $A_{IDD}$ (HF–Oscillator).
19. The parameter meets the $I^2C$ bus specification for standard–mode and fast–mode devices.
20. Not 100% tested.




(1) Maximum operating mode P89CE559 : $V_{DD}$ = 5.5 V
(2) Maximum operating mode P83CE559/P80CE559 : $V_{DD}$ = 5.5 V
(3) Maximum Idle Mode P89CE559 : $V_{DD}$ = 5.5 V
(4) Maximum Idle Mode P83CE559/P80CE559 : $V_{DD}$ = 5.5 V

**Figure 55. Supply Current ($I_{DD}$) as a Function of Frequency at XTAL1**




(1) Example of an actual transfer curve.
(2) The ideal transfer curve.
(3) Differential non-linearity ($DL_e$).
(4) Integral non-linearity ($IL_e$).
(5) Center of a step of the actual transfer curve.

**Figure 56. ADC Conversion Characteristic**

## 11. AC CHARACTERISTICS

### AC ELECTRICAL CHARACTERISTICS

$V_{DD}$ = 5 V ± 10% (EBx), $V_{SS}$ = 0 V, $t_{CLK}$ min = 1/fmax (maximum operating frequency)
$V_{DD}$ = 5 V ± 10% (EFx), $V_{SS}$ = 0 V, $t_{CLK}$ min = 1/fmax (maximum operating frequency)
$T_{amb}$ = 0 °C to +70 °C, $t_{CLK}$ min = 63 ns for P8xCE559EBx
$T_{amb}$ = –40 °C to +85 °C, $t_{CLK}$ min = 63 ns for P8xCE559EFx
C1 = 100 pF for Port 0, ALE and $\overline{PSEN}$ ; C1 = 80 pF for all other outputs unless otherwise specified.

| SYMBOL | FIGURE | PARAMETER | 12MHz CLOCK | | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | MIN | MAX | |
| $1/t_{CLK}$ | 60 | System clock frequency | | | | | 3.5 | 16 | MHz |
| $t_{LHLL}$ | 60 | ALE pulse width | 127 | | 85 | | $2t_{CLK}$–40 | | ns |
| $t_{AVLL}$ | 60 | Address valid to ALE LOW | 43 | | 23 | | $t_{CLK}$–40 | | ns |
| $t_{LLAX}$ | 60 | Address hold after ALE LOW | 53 | | 33 | | $t_{CLK}$–30 | | ns |
| $t_{LLIV}$ | 60 | ALE LOW to valid instruction in | | 234 | | 150 | | $4t_{CLK}$–100 | ns |
| $t_{LLPL}$ | 60 | ALE LOW to $\overline{PSEN}$ LOW | 53 | | 33 | | $t_{CLK}$–30 | | ns |
| $t_{PLPH}$ | 60 | $\overline{PSEN}$ pulse width | 205 | | 143 | | $3t_{CLK}$–45 | | ns |
| $t_{PLIV}$ | 60 | $\overline{PSEN}$ LOW to valid instruction in | | 145 | | 83 | | $3t_{CLK}$–105 | ns |
| $t_{PXIX}$ | 60 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 60 | Input instruction float after $\overline{PSEN}$ | | 59 | | 38 | | $t_{CLK}$–25 | ns |
| $t_{AVIV}$ | 60 | Address to valid instruction in | | 312 | | 208 | | $5t_{CLK}$–105 | ns |
| $t_{PLAZ}$ | 60 | $\overline{PSEN}$ LOW to address float | | 10 | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | | | |
| $t_{AVLL}$ | 61, 62 | Address valid to ALE LOW | 43 | | 23 | | $t_{CLK}$–40 | | ns |
| $t_{LLAX}$ | 61, 62 | Address hold after ALE LOW | 48 | | 28 | | $t_{CLK}$–35 | | ns |
| $t_{RLRH}$ | 61 | $\overline{RD}$ pulse width | 400 | | 275 | | $6t_{CLK}$–100 | | ns |
| $t_{WLWH}$ | 62 | $\overline{WR}$ pulse width | 400 | | 275 | | $6t_{CLK}$–100 | | ns |
| $t_{RLDV}$ | 61 | $\overline{RD}$ LOW to valid data in | | 252 | | 148 | | $5t_{CLK}$–165 | ns |
| $t_{RHDX}$ | 61 | Data hold after $\overline{RD}$ | 0 | | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 61 | Data float after $\overline{RD}$ | | 97 | | 55 | | $2t_{CLK}$–70 | ns |
| $t_{LLDV}$ | 61 | ALE LOW to valid data in | | 517 | | 350 | | $8t_{CLK}$–150 | ns |
| $t_{AVDV}$ | 61 | Address to valid data in | | 585 | | 398 | | $9t_{CLK}$–165 | ns |
| $t_{LLWL}$ | 61, 62 | ALE LOW to $\overline{RD}$ or $\overline{WR}$ LOW | 200 | 300 | 138 | 238 | $3t_{CLK}$–50 | $3t_{CLK}$+50 | ns |
| $t_{AVWL}$ | 61, 62 | Address valid to $\overline{WR}$ LOW or $\overline{RD}$ LOW | 203 | | 120 | | $4t_{CLK}$–130 | | ns |
| $t_{QVWX}$ | 62 | Data valid to $\overline{WR}$ transition | 33 | | 13 | | $t_{CLK}$–50 | | ns |
| $t_{QVWH}$ | 62 | Data before $\overline{WR}$ | 433 | | 288 | | $7t_{CLK}$–150 | | ns |
| $t_{WHQX}$ | 62 | Data hold after $\overline{WR}$ | 33 | | 13 | | $t_{CLK}$–50 | | ns |
| $t_{RLAZ}$ | 61 | $\overline{RD}$ low to address float | | 0 | | 0 | | 0 | ns |
| $t_{WHLH}$ | 61, 62 | $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH | 43 | 123 | 23 | 103 | $t_{CLK}$–40 | $t_{CLK}$+40 | ns |
| **UART Timing – Shift Register Mode** (Test Conditions: $T_{amb}$ = 0 °C to +70 °C; $V_{SS}$ = 0 V; Load Capacitance = 80pF) | | | | | | | | | |
| $t_{XLXL}$ | 64 | Serial port clock cycle time | 1.0 | | 0.75 | | $12t_{CLK}$ | | µs |
| $t_{QVXH}$ | 64 | Output data setup to clock rising edge | 700 | | 492 | | $10t_{CLK}$–133 | | ns |
| $t_{XHQX}$ | 64 | Output data hold after clock rising edge | 50 | | 8 | | $2t_{CLK}$–117 | | ns |
| $t_{XHDX}$ | 64 | Input data hold after clock rising edge | 0 | | 0 | | 0 | | ns |
| $t_{XHDV}$ | 64 | Clock rising edge to input data valid | | 700 | | 492 | | $10t_{CLK}$–133 | ns |

## AC ELECTRICAL CHARACTERISTICS (Continued)

| SYMBOL | PARAMETER | Standard-mode $I^2C$-bus | | Fast-mode $I^2C$-bus | | UNIT |
|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | |
| **$I^2C$ Interface timing (refer to Figure 63)** | | | | | | |
| $f_{SCL}$ | SCL clock frequency | 0 | 100 | 0 | 400 | kHz |
| $t_{BUF}$ | Bus free time between a STOP and START condition | 4.7 | – | 1.3 | – | µs |
| $t_{HD;\ STA}$ | Hold time (repeated) START condition. After this period, the first clock pulse is generated | 4.0 | – | 0.6 | – | µs |
| $t_{LOW}$ | LOW period of the SCL clock | 4.7 | – | 1.3 | – | µs |
| $t_{HIGH}$ | High period of the SCL clock | 4.0 | – | 0.6 | – | µs |
| $t_{SU;\ STA}$ | Set–up time for a repeated START condition | 4.7 | – | 0.6 | – | µs |
| $t_{HD;\ DAT}$ | Data hold time:<br>for CBUS competible masters (see Section 9, Notes 1, 3)<br>for $I^2C$–bus devices | <br>5.0<br>0 [1] | | <br>–<br>0[1] | <br>–<br>0.9[2] | µs |
| $t_{SU;\ DAT}$ | Data set–up time | 250 | – | 100[3] | – | ns |
| $t_{FD}$, $t_{FC}$ | Rise time of both SDA and SCL signals | – | 1000 | $20 + 0.1C_b$[4] | 300 | ns |
| $t_{FD}$, $t_{FC}$ | Fall time of both SDA and SCL signals | – | 300 | $20 + 0.1C_b$[4] | 300 | ns |
| $t_{SU;\ STO}$ | Set–up time for STOP condition | 4.0 | – | 0.6 | – | µs |
| $C_b$ | Capacitive load for each bus line | – | 400 | – | 400 | pF |
| $t_{SP}$ | Pulse width of spikes which must be suppressed by the input filter | – | – | 0 | 50 | ns |

All values referred to $V_{IH}$ and $V_{IL\ max}$ levels.

**NOTES:**
1. A device must internally provide a hold time of at least 300 ns from the SDA signal (referred to the $V_{IH\ min}$ of the SCL signal) in order to bridge the undefined region of the falling edge of SCL.
2. The maximum $t_{HD,DAT}$ has only to be met if the device does not stretch the LOW period ($t_{LOW}$) of the SCL signal.
3. A fast–mode $I^2C$–bus device can be used in a standard–mode $I^2C$–bus system, but the requirement $t_{SU,DAT} \geq 250$ ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line $t_{Rmax} + t_{SU,DAT} = 1000 + 250 = 1250$ ns (according to the standard–mode $I^2C$–bus specification) before the SCL line is released.
4. $C_b$ = total capacitance of one bus line in pF.

### Table 46. External clock drive XTAL1 (refer to Figure 57)

| SYMBOL | PARAMETER | VARIABLE CLOCK $f_{CLK}$ = 3.5 to 16 MHz | | UNIT |
|---|---|---|---|---|
| | | MIN | MAX | |
| $t_{CLK}$ | XTAL1 Period | 63 | 286 | ns |
| $t_{CLKH}$ | XTAL1 HIGH time | 20 | – | ns |
| $t_{CLKL}$ | XTAL1 LOW time | 20 | – | ns |
| $t_{CLKR}$ | XTAL1 rise time | – | 20 | ns |
| $t_{CLKF}$ | XTAL1 fall time | – | 20 | ns |
| $t_{CYC}$ [1] | Controller cycle time | 0.75 | 3.4 | µs |

**NOTE:**
1. $t_{CYC} = 12\ f_{CLK}$

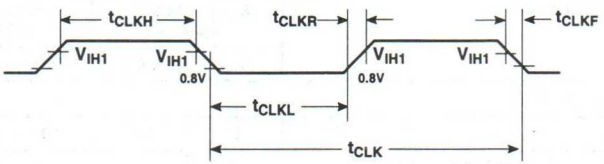
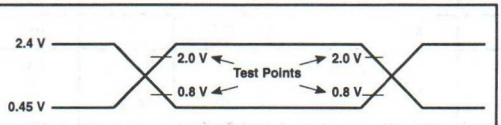
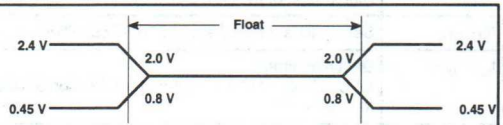



**Figure 57. External Clock Drive waveform**




**NOTE:**
AC inputs during testing are driven at 2.4V for a logic 'HIGH' and 0.45V for a logic 'LOW'. Timing measurements are made at 2.0 V for a logic 'HIGH' and 0.8 V for a logic 'LOW'.

**Figure 58. AC Testing Input/Output**




**NOTE:**
The float state is defined as the point at which a port 0 pin sinks 3.2mA or sources 400µA at the voltage test levels.

**Figure 59. AC Testing, Float Waveform**

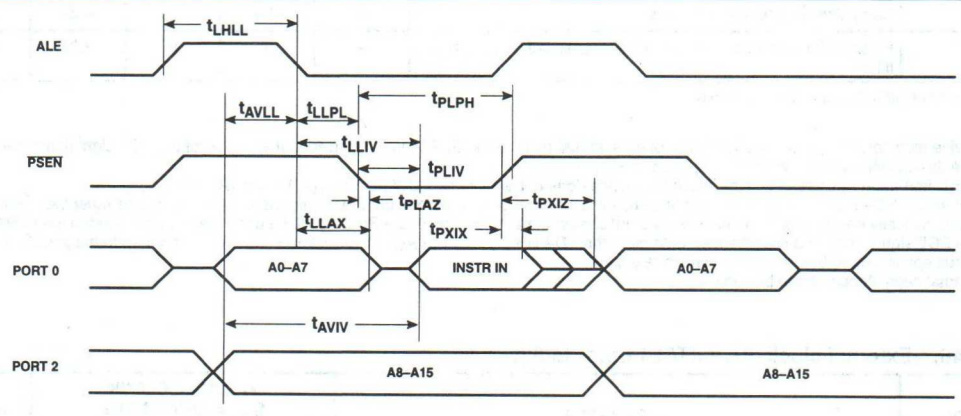



**Figure 60. External Program Memory Read Cycle**

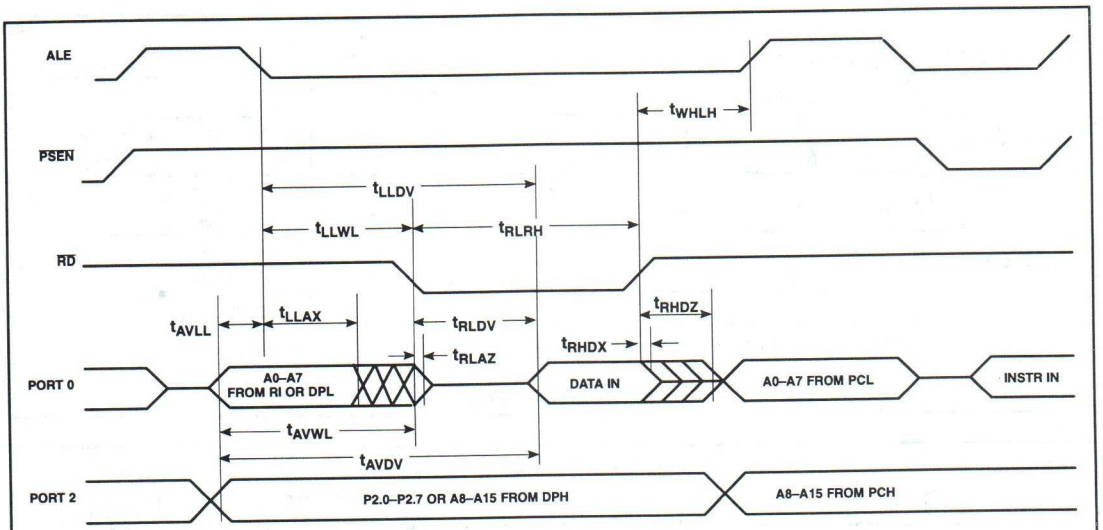



Figure 61. External Data Memory Read Cycle

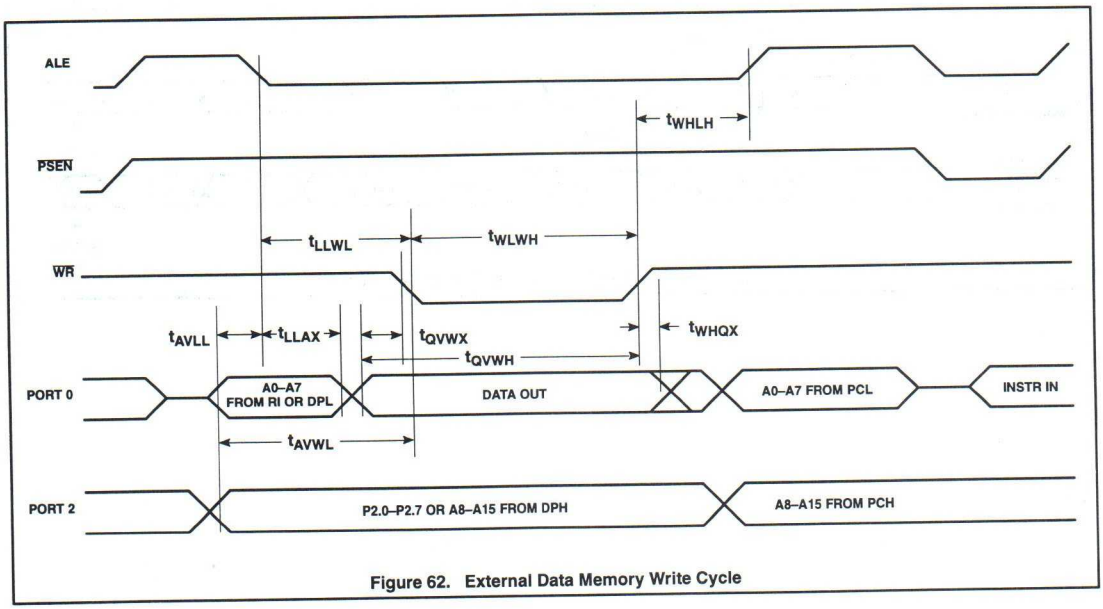



Figure 62. External Data Memory Write Cycle

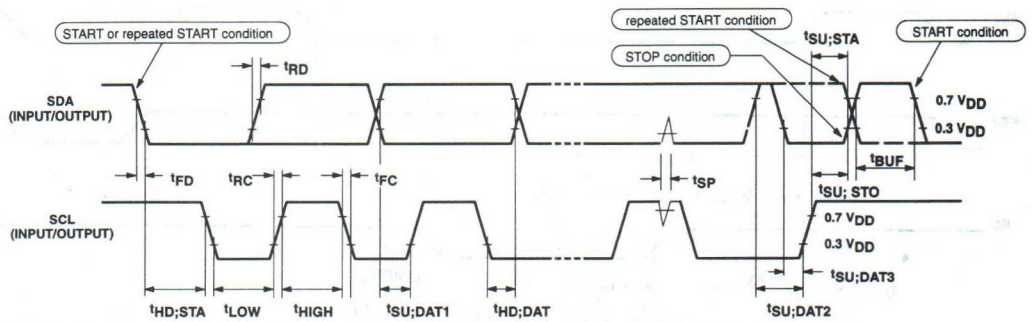



Figure 63. Timing SIO1 ($I^2C$) Interface

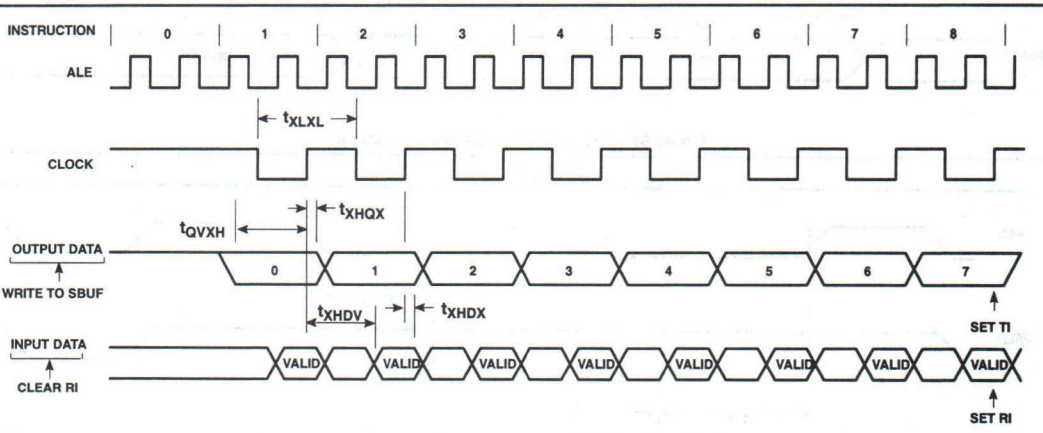



Figure 64. UART waveforms in Shift Register Mode

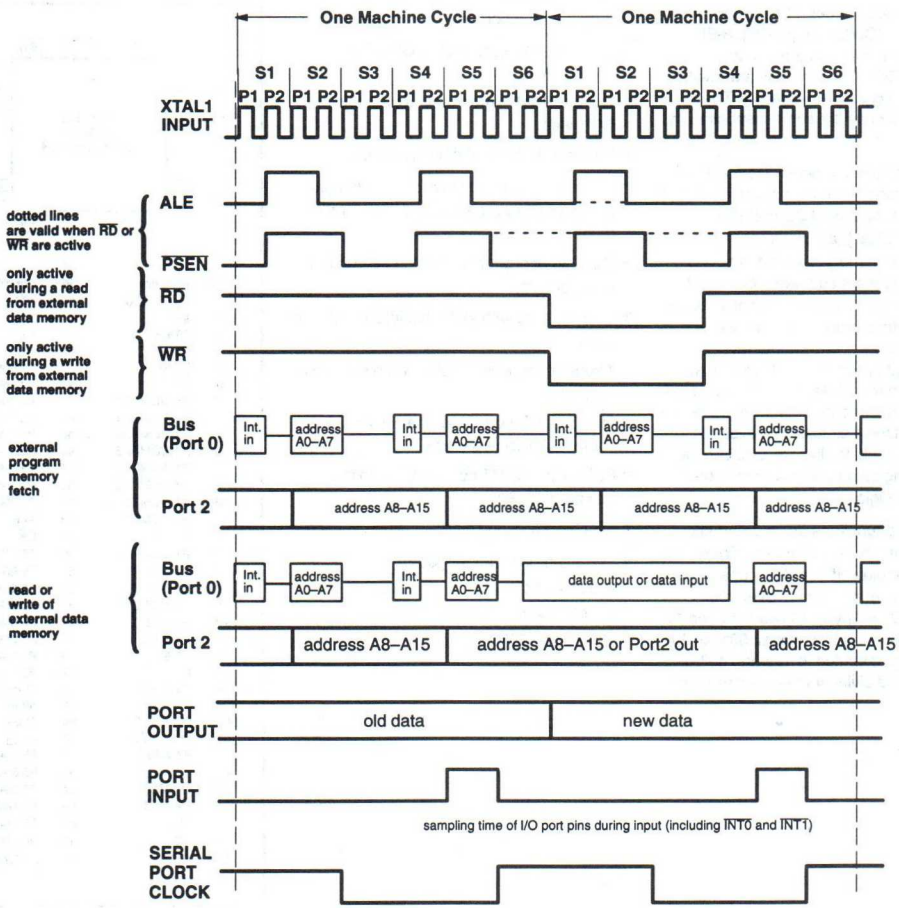



**Figure 65. Instruction cycle timing**

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# Single-chip 8-bit microcontroller 80C562/83C562

*Single-chip 8-bit microcontroller with 8-bit A/D, capture/compare timer, high-speed outputs, PWM*

## DESCRIPTION

The 80C562/83C562 (hereafter generically referred to as 8XC562) Single-Chip 8-Bit Microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 83C562/83C562 has the same instruction set as the 80C51.

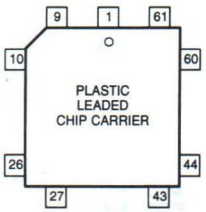
The 8XC562 contains a non-volatile 256 × 8 read-only program memory, a volatile 256 × 8 read/write data memory (83C562) (the 80C562 is ROMless), a volatile 256 × 8 read/write data memory, six 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), an additional 16-bit timer coupled to capture and compare latches, a 15-source, two-priority-level, nested interrupt structure, an 8-input ADC, two pulse width modulated outputs, standard 80C51 UART, a "watchdog" timer and on-chip oscillator and timing circuits. For systems that require extra capability, the 83C562 can be expanded using standard TTL compatible memories and logic.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte and 17 three-byte. With a 12MHz crystal, 58% of the instructions are executed in 1μs and 40% in 2μs. Multiply and divide instructions require 4μs.

## FEATURES

- 80C51 instruction set
- 8k × 8 ROM expandable externally to 64k bytes
- 256 × 8 RAM, expandable externally to 64k bytes
- Two standard 16-bit timer/counters
- An additional 16-bit timer/counter coupled to four capture registers and three compare registers
- Capable of producing eight synchronized, timed outputs
- An 8-bit ADC with eight multiplexed analog inputs
- Two 8-bit resolution, pulse width modulated outputs
- Five 8-bit I/O ports plus one 8-bit input port shared with analog inputs
- Full-duplex UART compatible with the standard 80C51
- On-chip watchdog timer
- Three temperature ranges
  - 0 to +70°C
  - –40 to +85°C
  - –40 to +125°C

## PIN CONFIGURATION




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P5.0/ADC0 | 35 | XTAL1 |
| 2 | $V_{DD}$ | 36 | $V_{SS}$ |
| 3 | STADC | 37 | $V_{SS}$ |
| 4 | $\overline{PWM0}$ | 38 | NC |
| 5 | $\overline{PWM1}$ | 39 | P2.0/A08 |
| 6 | $\overline{EW}$ | 40 | P2.1/A09 |
| 7 | P4.0/CMSR0 | 41 | P2.2/A10 |
| 8 | P4.1/CMSR1 | 42 | P2.3/A11 |
| 9 | P4.2/CMSR2 | 43 | P2.4/A12 |
| 10 | P4.3/CMSR3 | 44 | P2.5/A13 |
| 11 | P4.4/CMSR4 | 45 | P2.6/A14 |
| 12 | P4.5/CMSR5 | 46 | P2.7/A15 |
| 13 | P4.6/CMT0 | 47 | $\overline{PSEN}$ |
| 14 | P4.7/CMT1 | 48 | ALE |
| 15 | RST | 49 | $\overline{EA}$ |
| 16 | P1.0/CT0I | 50 | P0.7/AD7 |
| 17 | P1.1/CT1I | 51 | P0.6/AD6 |
| 18 | P1.2/CT2I | 52 | P0.5/AD5 |
| 19 | P1.3/CT3I | 53 | P0.4/AD4 |
| 20 | P1.4/T2 | 54 | P0.3/AD3 |
| 21 | P1.5/RT2 | 55 | P0.2/AD2 |
| 22 | P1.6 | 56 | P0.1/AD1 |
| 23 | P1.7 | 57 | P0.0/AD0 |
| 24 | P3.0/RxD | 58 | AVref– |
| 25 | P3.1/TxD | 59 | AVref+ |
| 26 | P3.2/$\overline{INT0}$ | 60 | $AV_{SS}$ |
| 27 | P3.3/$\overline{INT1}$ | 61 | $AV_{DD}$ |
| 28 | P3.4/T0 | 62 | P5.7/ADC7 |
| 29 | P3.5/T1 | 63 | P5.6/ADC6 |
| 30 | P3.6/$\overline{WR}$ | 64 | P5.5/ADC5 |
| 31 | P3.7/$\overline{RD}$ | 65 | P5.4/ADC4 |
| 32 | NC | 66 | P5.3/ADC3 |
| 33 | NC | 67 | P5.2/ADC2 |
| 34 | XTAL2 | 68 | P5.1/ADC1 |

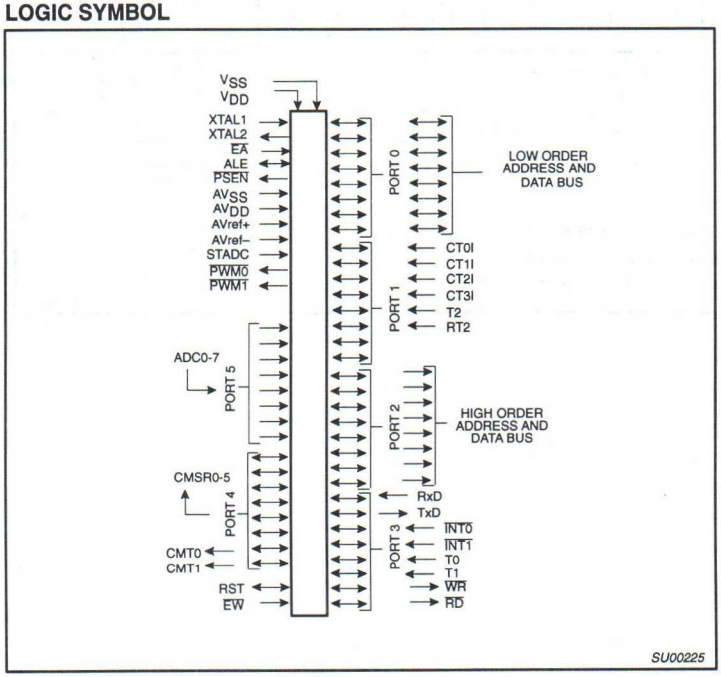
*SU00224*

## ORDERING INFORMATION

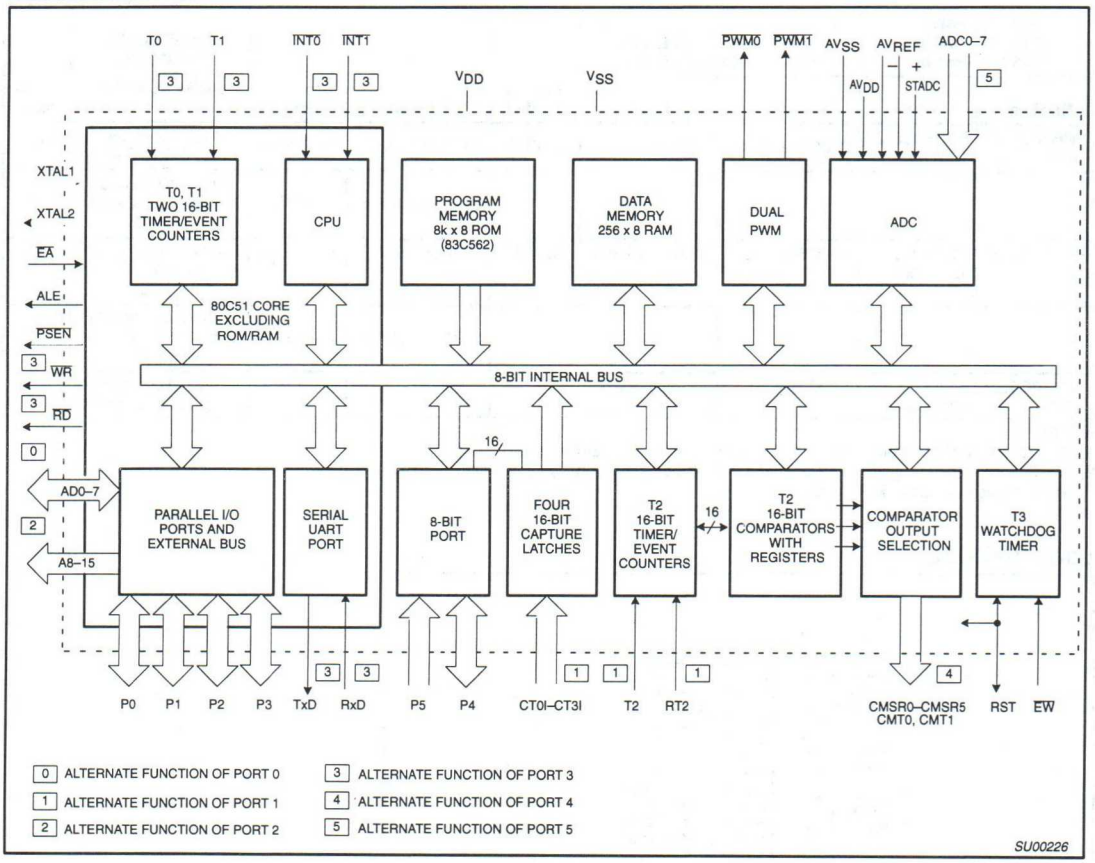
| PHILIPS PART ORDER NUMBER PART MARKING | | PHILIPS NORTH AMERICA PART ORDER NUMBER | | Drawing Number | EPROM | Drawing Number | TEMPERATURE RANGE °C AND PACKAGE | FREQ MHz |
|---|---|---|---|---|---|---|---|---|
| ROMless | ROM | ROMless | ROM | | | | | |
| PCB80C562-16WP | PCB83C562-16WP/xxx | S80C562-4A68 | S83C562-4A68 | SOT188 | S87C552-4A68[2] | SOT188-3 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| | | | | | S87C552-4K68[2] | 1473A | 0 to +70, Plastic Leaded Chip Carrier w/Window | 16 |
| PCF80C562-12WP | PCF83C562-12WP/xxx | S80C562-2A68 | S83C562-2A68 | SOT188 | S87C552-5A68[2] | SOT188-3 | –40 to +85, Plastic Leaded Chip Carrier | 12 |
| | | | | | S87C552-5K68[2] | 1473A | –40 to +85, Plastic Leaded Chip Carrier w/Window | 12 |
| PCA80C562-12WP | PCA83C562-12WP/xxx | S80C562-6A68 | S83C562-6A68 | SOT188 | | | –40 to +125, Plastic Leaded Chip Carrier | 12 |

**NOTES:**
1. 80C562 and 83C562 frequency range is 1.2MHz–12MHz or 1.2MHz–16MHz.
2. 87C552 frequency range is 3.5MHz–16MHz. For full specification, see the 87C552 data sheets.
3. xxx denotes the ROM code number.

## LOGIC SYMBOL



## BLOCK DIAGRAM



## PIN DESCRIPTION

| MNEMONIC | PIN NO. | TYPE | NAME AND FUNCTION |
|---|---|---|---|
| $V_{DD}$ | 2 | I | **Digital Power Supply:** +5V power supply pin during normal operation, idle and power-down mode. |
| STADC | 3 | I | **Start ADC Operation:** Input starting analog to digital conversion (ADC operation can also be started by software). |
| $\overline{PWM0}$ | 4 | O | **Pulse Width Modulation:** Output 0. |
| $\overline{PWM1}$ | 5 | O | **Pulse Width Modulation:** Output 1. |
| $\overline{EW}$ | 6 | I | **Enable Watchdog Timer:** Enable for T3 watchdog timer and disable power-down mode. |
| P0.0–P0.7 | 57–50 | I/O | **Port 0:** Port 0 is an 8-bit open-drain bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application it uses strong internal pull-ups when emitting 1s. |
| P1.0–P1.7 | 16–23 | I/O | **Port 1:** 8-bit I/O port. Alternate functions include: |
| | 16–23 | I/O | **(P1.0–P1.7):** Quasi-bidirectional port pins. |
| | 16–19 | I/O | **CT0I–CT3I (P1.0–P1.3):** Capture timer input signals for timer T2. |
| | 20 | I | **T2 (P1.4):** T2 event input |
| | 21 | I | **RT2 (P1.5):** T2 timer reset signal. Rising edge triggered. |
| P2.0–P2.7 | 39–46 | I/O | **Port 2:** 8-bit quasi-bidirectional I/O port. Alternate function: High-order address byte for external memory (A08–A15). |
| P3.0–P3.7 | 24–31 | I/O | **Port 3:** 8-bit quasi-bidirectional I/O port. Alternate functions include: |
| | 24 | | **RxD(P3.0):** Serial input port. |
| | 25 | | **TxD (P3.1):** Serial output port. |
| | 26 | | **$\overline{INT0}$ (P3.2):** External interrupt. |
| | 27 | | **$\overline{INT1}$ (P3.3):** External interrupt. |
| | 28 | | **T0 (P3.4):** Timer 0 external input. |
| | 29 | | **T1 (P3.5):** Timer 1 external input. |
| | 30 | | **$\overline{WR}$ (P3.6):** External data memory write strobe. |
| | 31 | | **$\overline{RD}$ (P3.7):** External data memory read strobe. |
| P4.0–P4.7 | 7–14 | I/O | **Port 4:** 8-bit quasi-bidirectional I/O port. Alternate functions include: |
| | 7–12 | O | **CMSR0–CMSR5 (P4.0–P4.5):** Timer T2 compare and set/reset outputs on a match with timer T2. |
| | 13, 14 | O | **CMT0, CMT1 (P4.6, P4.7):** Timer T2 compare and toggle outputs on a match with timer T2. |
| P5.0–P5.7 | 68–62, 1 | I | **Port 5:** 8-bit input port. **ADC0–ADC7 (P5.0–P5.7):** Alternate function: Eight input channels to ADC. |
| RST | 15 | I/O | **Reset:** Input to reset the 87C552. It also provides a reset pulse as output when timer T3 overflows. |
| XTAL1 | 35 | I | **Crystal Input 1:** Input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external clock signal when an external oscillator is used. |
| XTAL2 | 34 | O | **Crystal Input 2:** Output of the inverting amplifier that forms the oscillator. Left open–circuit when an external clock is used. |
| $V_{SS}$ | 36, 37 | I | **Digital ground.** |
| $\overline{PSEN}$ | 47 | O | **Program Store Enable:** Active-low read strobe to external program memory. |
| ALE | 48 | O | **Address Latch Enable:** Latches the low byte of the address during accesses to external memory. It is activated every six oscillator periods. During an external data memory access, one ALE pulse is skipped. ALE can drive up to eight LS TTL inputs and handles CMOS inputs without an external pull-up. |
| $\overline{EA}$ | 49 | I | **External Access:** When $\overline{EA}$ is held at TTL level high, the CPU executes out of the internal program ROM provided the program counter is less than 8192. When $\overline{EA}$ is held at TTL low level, the CPU executes out of external program memory. $\overline{EA}$ is not allowed to float. |
| $AV_{REF-}$ | 58 | I | **Analog to Digital Conversion Reference Resistor:** Low-end. |
| $AV_{REF+}$ | 59 | I | **Analog to Digital Conversion Reference Resistor:** High-end. |
| $AV_{SS}$ | 60 | I | **Analog Ground** |
| $AV_{DD}$ | 61 | I | **Analog Power Supply** |

**NOTE:**
1. To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher or lower than $V_{DD}$ +0.5V or $V_{SS}$ – 0.5V, respectively.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the logic symbol.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To ensure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{DD}$ and RST must come up at the same time for a proper start-up.

## IDLE MODE

In the idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. the control bits for the reduced power modes are in the special function register PCON. Table 1 shows the state of the I/O ports during low current operating modes.

**Table 1. External Pin Status During Idle and Power-Down Modes**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 | PORT 4 | PWM0/ PWM1 |
|---|---|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data | Data | High |
| Idle | External | 1 | 1 | Float | Data | Address | Data | Data | High |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data | Data | High |
| Power-down | External | 0 | 0 | Float | Data | Data | Data | Data | High |

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Input, output DC current on any single I/O pin | 5.0 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.0 | W |
| Storage temperature range | –65 to +150 | °C |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$V_{SS}$, $AV_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|
| $V_{DD}$ | Supply voltage | | | | |
| | PCB8XC562 | | 4.0 | 6.0 | V |
| | PCF8XC562 | | 4.0 | 6.0 | V |
| | PCA8XC562 | | 4.5 | 5.5 | V |
| $I_{DD}$ | Supply current operating: | See notes 1 and 2 | | | |
| | PCB8XC562 | $f_{OSC}$ = 16MHz | | 45 | mA |
| | PCF8XC562 | $f_{OSC}$ = 12MHz | | 34 | mA |
| | PCA8XC562 | $f_{OSC}$ = 12MHz | | 30 | mA |
| $I_{ID}$ | Idle mode: | See notes 1 and 3 | | | |
| | PCB8XC562 | $f_{OSC}$ = 16MHz | | 10 | mA |
| | PCF8XC562 | $f_{OSC}$ = 12MHz | | 8 | mA |
| | PCA8XC562 | $f_{OSC}$ = 12MHz | | 7 | mA |
| $I_{PD}$ | Power-down current: | See notes 1 and 4; $2V < V_{PD} < V_{DD}$ max | | | |
| | PCB8XC562 | | | 50 | µA |
| | PCF8XC562 | | | 50 | µA |
| | PCA8XC562 | | | 100 | µA |
| **Inputs** | | | | | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$ | | –0.5 | $0.2V_{DD}$–0.1 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | | –0.5 | $0.2V_{DD}$–0.3 | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST | | $0.2V_{DD}$+0.9 | $V_{DD}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | | $0.7V_{DD}$ | $V_{DD}$+0.5 | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3, 4 | $V_{IN}$ = 0.45V | | –50 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3, 4 | See note 5 | | –650 | µA |
| $\pm I_{IL1}$ | Input leakage current, port 0, $\overline{EA}$, STADC, $\overline{EW}$ | $0.45V < V_I < V_{DD}$ | | 10 | µA |
| **Outputs** | | | | | |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3, 4 | $I_{OL}$ = 1.6mA[6] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$ | $I_{OL}$ = 3.2mA[6] | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3, 4 | $V_{DD}$ + 5V±10% | | | |
| | | $-I_{OH}$ = 60µA | 2.4 | | V |
| | | $-I_{OH}$ = 25µA | $0.75V_{DD}$ | | V |
| | | $-I_{OH}$ = 10µA | $0.9V_{DD}$ | | V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$)[7] | $V_{DD}$ + 5V±10% | | | |
| | | $-I_{OH}$ = 400µA | 2.4 | | V |
| | | $-I_{OH}$ = 150µA | $0.75V_{DD}$ | | V |
| | | $-I_{OH}$ = 40µA | $0.9V_{DD}$ | | V |
| $V_{OH2}$ | Output high voltage (RST) | $-I_{OH}$ = 400µA | 2.4 | | V |
| | | $-I_{OH}$ = 120µA | $0.8V_{DD}$ | | V |
| $R_{RST}$ | Internal reset pull-down resistor | | 50 | 150 | kΩ |
| $C_{IO}$ | Pin capacitance | Test freq = 1MHz, $T_{amb}$ = 25°C | | 10 | pF |

## DC ELECTRICAL CHARACTERISTICS (Continued)

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|
| **Analog Inputs** | | | | | |
| $AV_{DD}$ | Analog supply voltage:<br>PCB8XC562<br>PCF8XC562<br>PCA8XC562 | <br>$AV_{DD} = V_{DD} \pm 0.2V$<br>$AV_{DD} = V_{DD} \pm 0.2V$<br>$AV_{DD} = V_{DD} \pm 0.2V$ | <br>4.0<br>4.0<br>4.5 | <br>6.0<br>6.0<br>5.5 | <br>V<br>V<br>V |
| $AI_{DD}$ | Analog supply current: operating: | Port 5 = 0 to $AV_{DD}$ | | 1.2 | mA |
| $AI_{ID}$ | Idle mode:<br>PCB8XC562<br>PCF8XC562<br>PCA8XC562 | | | <br>50<br>50<br>100 | <br>µA<br>µA<br>µA |
| $AI_{PD}$ | Power-down mode:<br>PCB8XC562<br>PCF8XC562<br>PCA8XC562 | $2V < AV_{PD} < AV_{DD}$ max | | <br>50<br>50<br>100 | <br>µA<br>µA<br>µA |
| $AV_{IN}$ | Analog input voltage | | $AV_{SS}-0.2$ | $AV_{DD}+0.2$ | V |
| $AV_{REF}$ | Reference voltage:<br>$AV_{REF-}$<br>$AV_{REF+}$ | | <br>$AV_{SS}-0.2$ | <br>$AV_{DD}+0.2$ | <br>V<br>V |
| $R_{REF}$ | Resistance between $AV_{REF+}$ and $AV_{REF-}$ | | 5 | 25 | kΩ |
| $C_{IA}$ | Analog input capacitance | | | 15 | pF |
| $t_{ADS}$ | Sampling time | | | $6t_{CY}$ | µs |
| $t_{ADC}$ | Conversion time (including sampling time) | | | $24t_{CY}$ | µs |
| $DL_e$ | Differential non-linearity[8, 9, 10] | | | ±1 | LSB |
| $IL_e$ | Integral non-linearity[8, 11] | | | ±1 | LSB |
| $OS_e$ | Offset error[8, 12] | | | ±1 | LSB |
| $G_e$ | Gain error[8, 13] | | | 0.4 | % |
| $M_{CTC}$ | Channel to channel matching | | | ±1 | LSB |
| $C_t$ | Crosstalk between inputs of port 5[14] | 0–100kHz | | –60 | dB |

**NOTES:**

1. See Figures 8 through 12 for $I_{DD}$ test conditions.
2. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 10ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5V$; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = $\overline{EW}$ = $V_{DD}$; STADC = $V_{SS}$.
3. The idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 10ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5V$; XTAL2 not connected; Port 0 = $\overline{EW}$ = $V_{DD}$; $\overline{EA}$ = RST = STADC = $V_{SS}$.
4. The power-down current is measured with all output pins disconnected; XTAL2 not connected; Port 0 = $\overline{EW}$ = $V_{DD}$; $\overline{EA}$ = RST = STADC = XTAL1 = $V_{SS}$.
5. Pins of ports 1, 2, 3, and 4 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
6. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
7. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{DD}$ specification when the address bits are stabilizing.
8. Conditions: $AV_{REF-} = 0V$; $AV_{DD} = 5.0V$, $AV_{REF+} = 5.12V$. ADC is monotonic with no missing codes.
9. The differential non-linearity ($DL_e$) is the difference between the actual step width and the ideal step width. (See Figure 1.)
10. The ADC is monotonic; there are no missing codes.
11. The integral non-linearity ($IL_e$) is the peak difference between the center of the steps of the actual and the ideal transfer curve after appropriate adjustment of gain and offset error. (See Figure 1.)
12. The offset error ($OS_e$) is the absolute difference between the straight line which fits the actual transfer curve (after removing gain error), and a straight line which fits the ideal transfer curve. (See Figure 1.)
13. The gain error ($G_e$) is the relative difference in percent between the straight line fitting the actual transfer curve (after removing offset error), and the straight line which fits the ideal transfer curve. Gain error is constant at every point on the transfer curve. (See Figure 1.)
14. This should be considered when both analog and digital signals are simultaneously input to port 5.

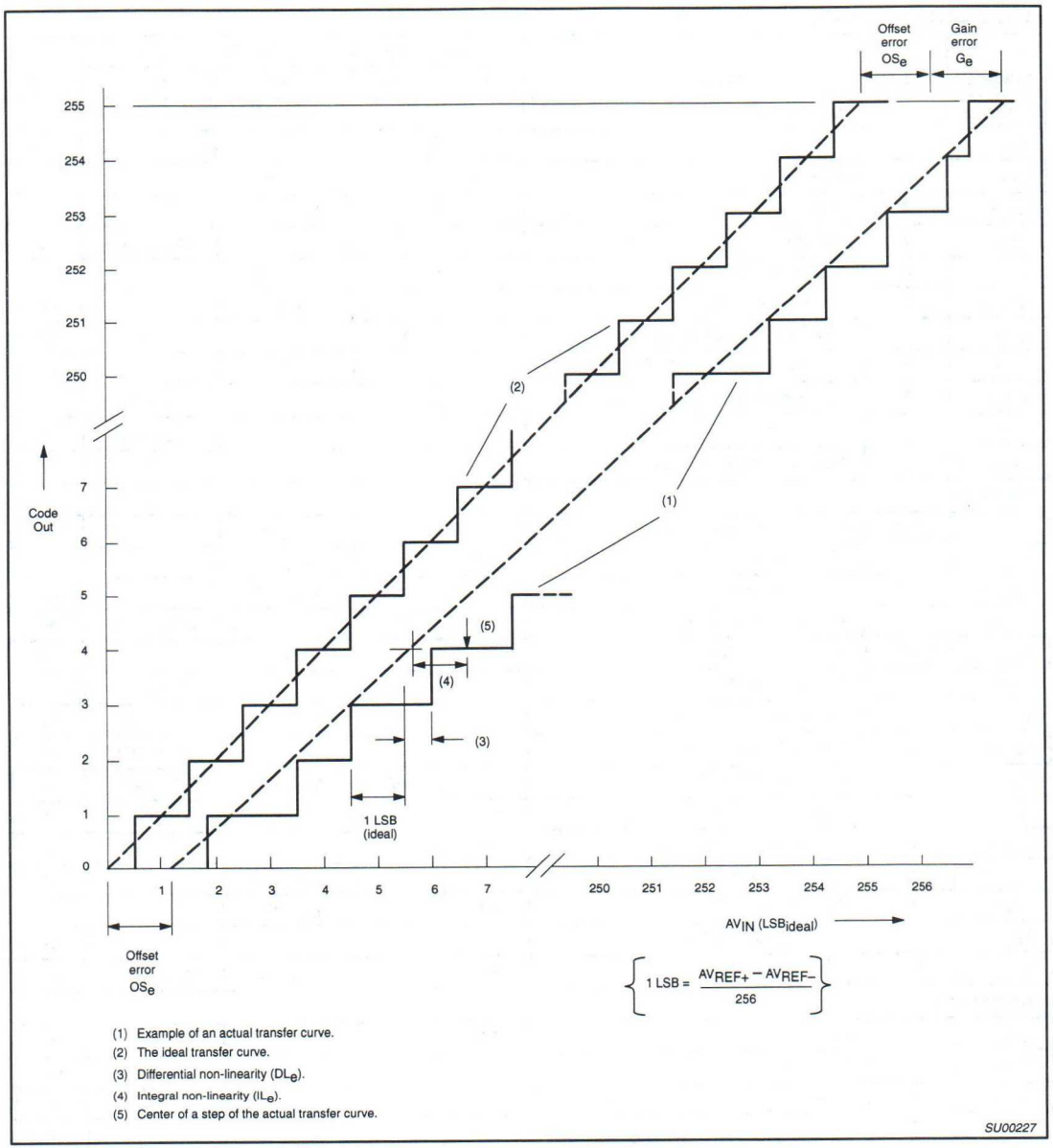



**Figure 1. ADC Conversion Characteristic**

## AC ELECTRICAL CHARACTERISTICS[1, 2]

| SYMBOL | FIGURE | PARAMETER | 12MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 2 | Oscillator frequency | | | 1.2 | 16 | MHz |
| $t_{LHLL}$ | 2 | ALE pulse width | 127 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 2 | Address valid to ALE low | 28 | | $t_{CLCL}-55$ | | ns |
| $t_{LLAX}$ | 2 | Address hold after ALE low | 48 | | $t_{CLCL}-35$ | | ns |
| $t_{LLIV}$ | 2 | ALE low to valid instruction in | | 234 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 2 | ALE low to $\overline{PSEN}$ low | 43 | | $t_{CLCL}-40$ | | ns |
| $t_{PLPH}$ | 2 | $\overline{PSEN}$ pulse width | 205 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 2 | $\overline{PSEN}$ low to valid instruction in | | 145 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 2 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 2 | Input instruction float after $\overline{PSEN}$ | | 59 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 2 | Address to valid instruction in | | 312 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 2 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{AVLL}$ | 3, 4 | Address valid to ALE low | 43 | | $t_{CLCL}-35$ | | ns |
| $t_{RLRH}$ | 3 | $\overline{RD}$ pulse width | 400 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 4 | $\overline{WR}$ pulse width | 400 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 3 | $\overline{RD}$ low to valid data in | | 252 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 3 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 3 | Data float after $\overline{RD}$ | | 97 | | $2t_{CLCL}-70$ | ns |
| $t_{LLDV}$ | 3 | ALE low to valid data in | | 517 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 3 | Address to valid data in | | 585 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 3, 4 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 200 | 300 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 3, 4 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 203 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 4 | Data valid to $\overline{WR}$ transition | 23 | | $t_{CLCL}-60$ | | ns |
| $t_{DW}$ | 4 | Data before $\overline{WR}$ | 433 | | $7t_{CLCL}-150$ | | ns |
| $t_{WHQX}$ | 4 | Data hold after $\overline{WR}$ | 33 | | $t_{CLCL}-50$ | | ns |
| $t_{RLAZ}$ | 3 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 3, 4 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 43 | 123 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 5 | High time[3] | 20 | | 20 | | ns |
| $t_{CLCX}$ | 5 | Low time[3] | 20 | | 20 | | ns |
| $t_{CLCH}$ | 5 | Rise time[3] | | 20 | | 20 | ns |
| $t_{CHCL}$ | 5 | Fall time[3] | | 20 | | 20 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. These values are characterized but not 100% production tested.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{PSEN}$ low.

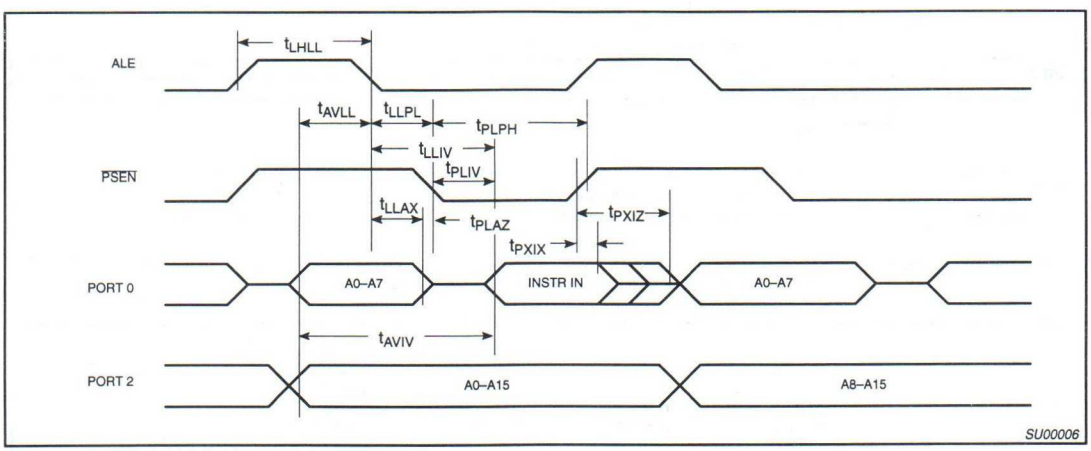



**Figure 2. External Program Memory Read Cycle**

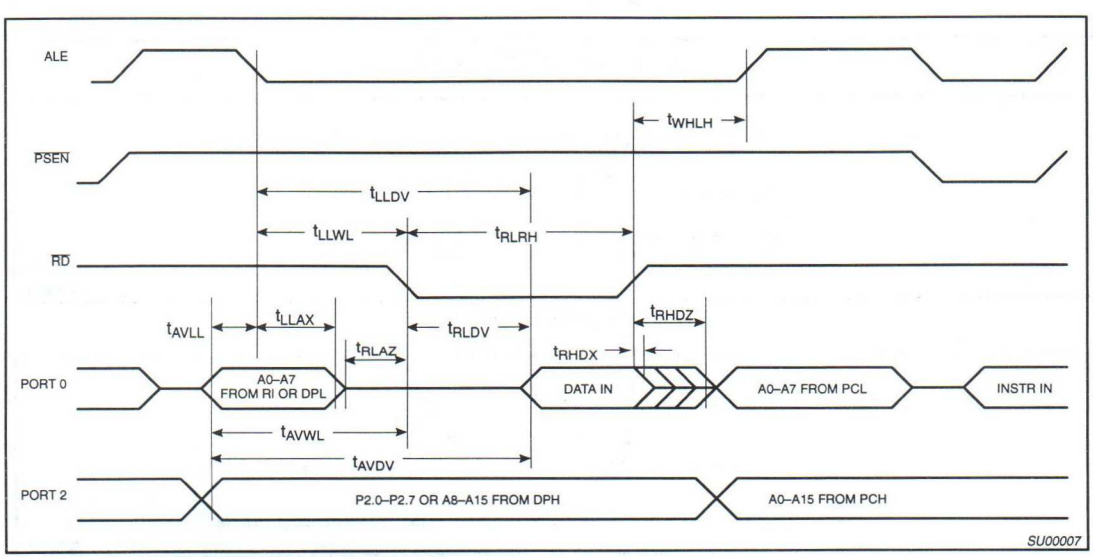



**Figure 3. External Data Memory Read Cycle**

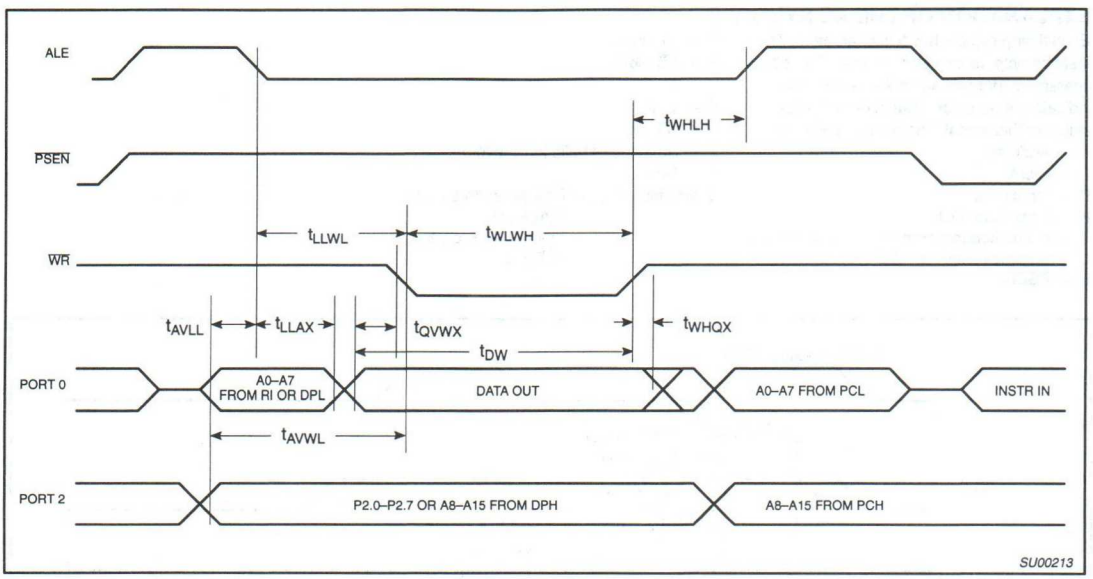



**Figure 4. External Data Memory Write Cycle**

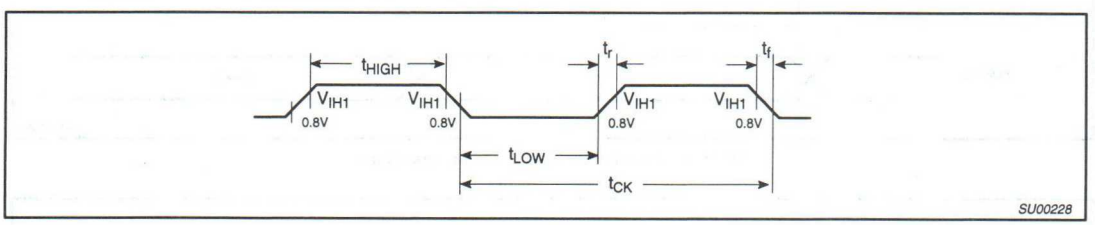



**Figure 5. External Clock Drive XTAL1**

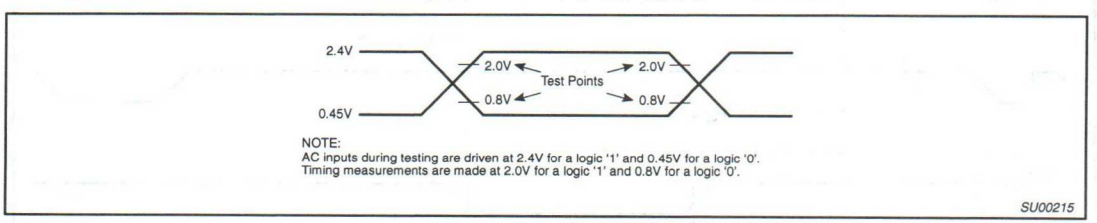



**Figure 6. AC Testing Input/Output**

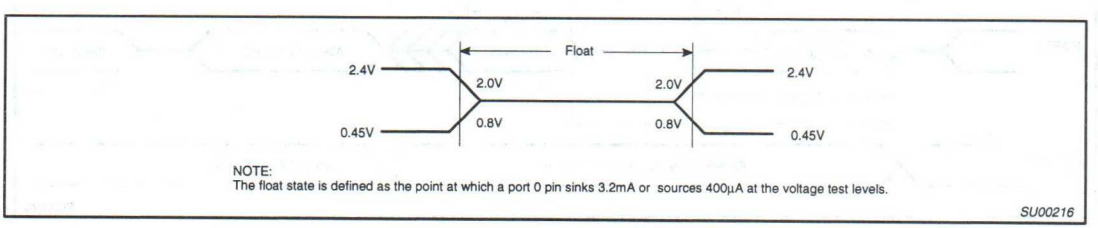



**Figure 7. AC Testing Input, Float Waveform**

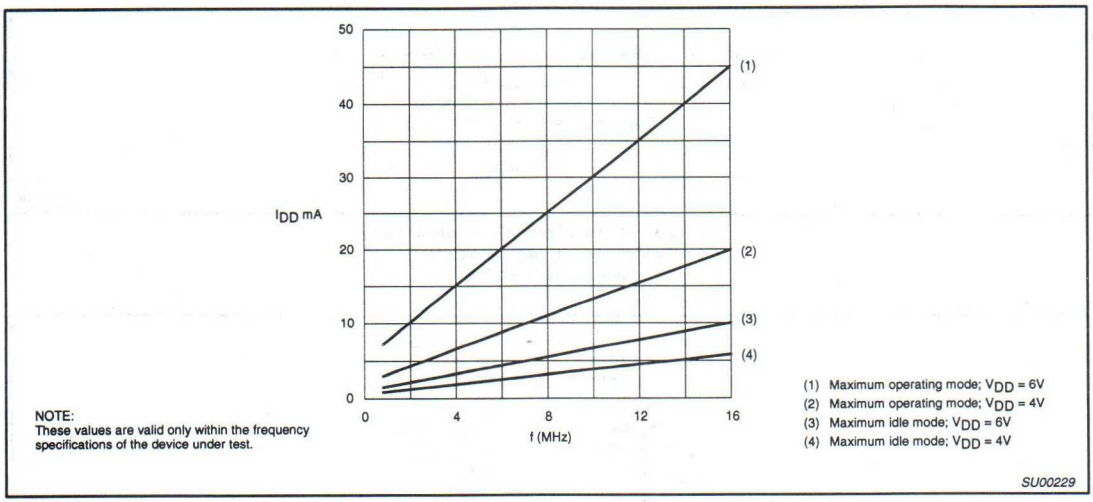



**Figure 8. Supply Current ($I_{DD}$) as a Function of Frequency at XTAL1 ($f_{OSC}$)**

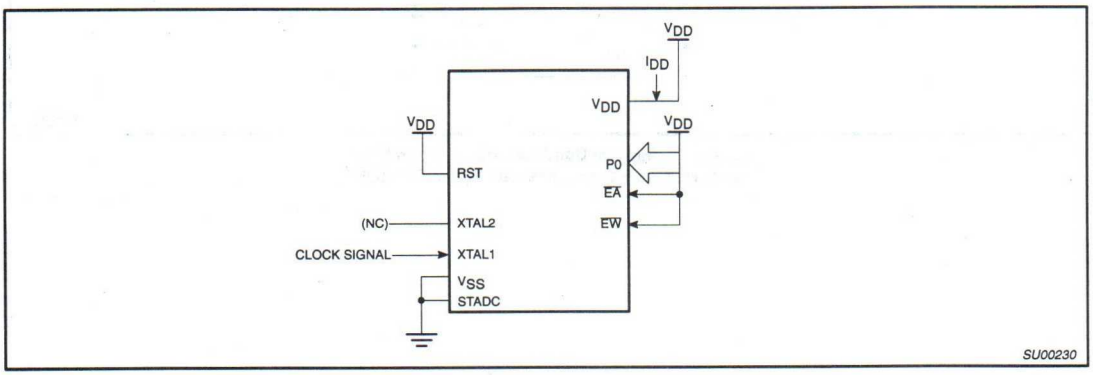



**Figure 9. $I_{DD}$ Test Condition, Active Mode**
**All other pins are disconnected**

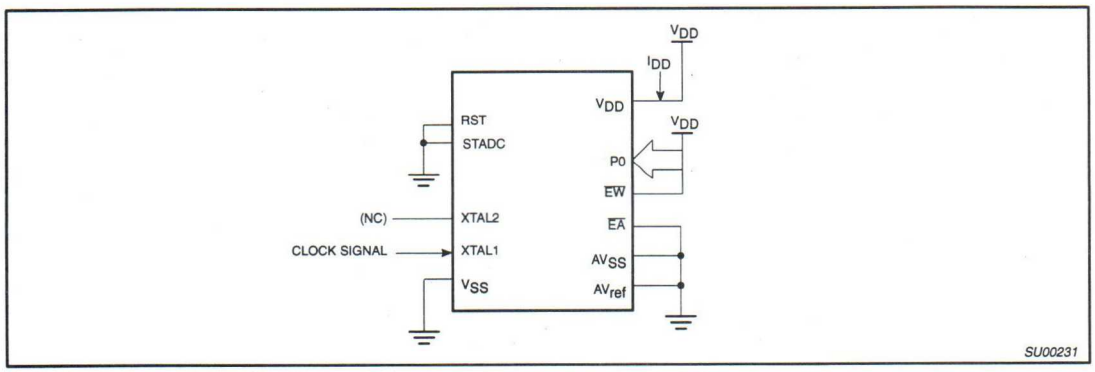



**Figure 10. $I_{DD}$ Test Condition, Idle Mode**
**All other pins are disconnected**

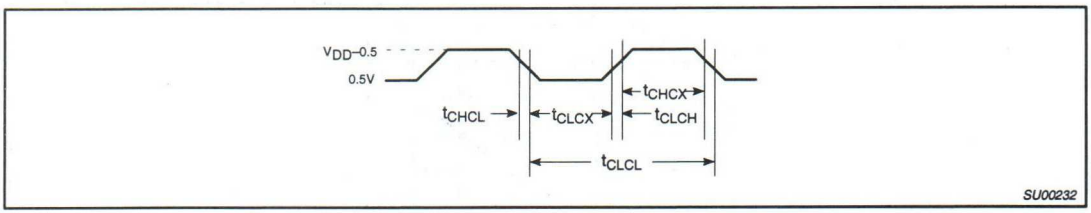



**Figure 11. Clock Signal Waveform for $I_{DD}$ Tests in Active and Idle Modes $t_{CLCH} = t_{CHCL} = 10ns$**

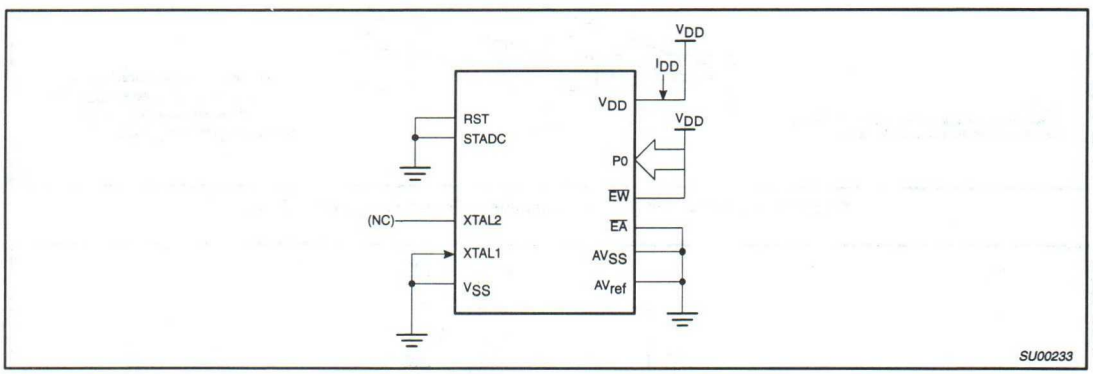



**Figure 12. $I_{DD}$ Test Condition, Power Down Mode All other pins are disconnected. $V_{DD}$ = 2V to 5.5V**

# CMOS single-chip 8-bit microcontrollers 80C575/83C575/87C575

## DESCRIPTION

The Philips 80C575/83C575/87C575 is a high-performance microcontroller fabricated with Philips high-density CMOS technology. The Philips CMOS technology combines the high speed and density characteristics of HMOS with the low power attributes of CMOS. Philips epitaxial substrate minimizes latch-up sensitivity.

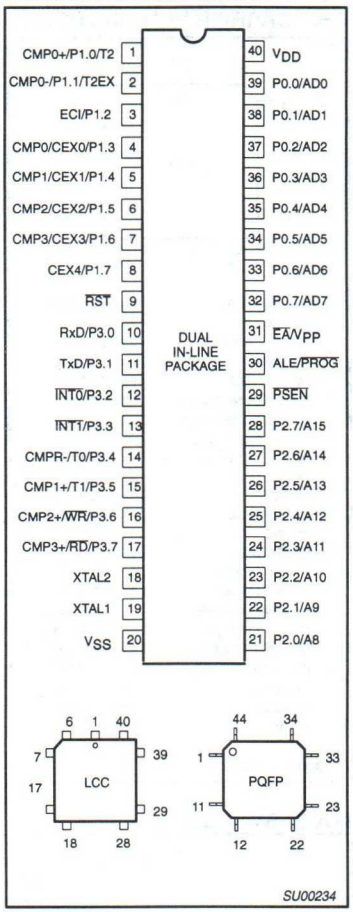
The 8XC575 contains an 8k × 8 ROM (83C575) EPROM (87C575), a 256 × 8 RAM, 32 I/O lines, three 16-bit counter/timers, a Programmable Counter Array (PCA), a seven-source, two-priority level nested interrupt structure, an enhanced UART, four analog comparators, power-fail detect and oscillator fail detect circuits, and on-chip oscillator and clock circuits.

In addition, the 8XC575 has a low active reset, and the port pins are reset to a low level. There is also a fully configurable watchdog timer, and internal power on clear circuit. The part includes idle mode and power-down mode states for reduced power consumption.

## FEATURES

- 80C51 based architecture
  - 8k × 8 ROM (83C575)
  - 8k × 8 EPROM (87C575)
  - ROMless (80C575)
  - 256 × 8 RAM
  - Three 16-bit counter/timers
  - Programmable Counter Array
  - Enhanced UART
  - Boolean processor
  - Oscillator fail detect
  - Low active reset
  - Asynchronous low port reset
  - Schmitt trigger inputs
  - 4 analog comparators
  - Watchdog timer
  - Low $V_{CC}$ detect
- Memory addressing capability
  - 64k ROM and 64k RAM
- Power control modes:
  - Idle mode
  - Power-down mode
- CMOS and TTL compatible
- 4.0 to 16MHz
- Extended temperature ranges
- OTP package available

## PIN CONFIGURATIONS

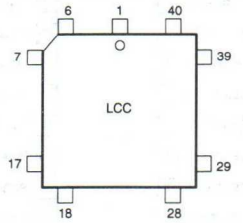



## ORDERING INFORMATION

| ROMless | ROM | EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQ (MHz) | DRAWING NUMBER |
|---|---|---|---|---|---|---|
| P80C575EBPN | P83C575EBPN | P87C575EBPN | OTP | 0 to +70, 40-Pin Plastic Dual In-line Package | 16 | SOT129-1 |
| P80C575EBAA | P83C575EBAA | P87C575EBAA | OTP | 0 to +70, 44-Pin Plastic Leaded Chip Carrier | 16 | SOT187-2 |
| | | P87C575EBFFA | UV | 0 to +70, 40-Pin Ceramic Dual In-line Package | 16 | 0590B |
| | | P87C575EBLKA | UV | 0 to +70, 44-Pin Ceramic Leaded Chip Carrier | 16 | 1472A |
| P80C575EHPN | P83C575EHPN | P87C575EHPN | OTP | –40 to +125, 40-Pin Plastic Dual In-line Package | 16 | SOT129-1 |
| P80C575EHAA | P83C575EHAA | P87C575EHAA | OTP | –40 to +125, 44-Pin Plastic Leaded Chip Carrier | 16 | SOT187-2 |
| | | P87C575EHFFA | UV | –40 to +125, 40-Pin Ceramic Dual In-line Package | 16 | 0590B |
| P80C575EBBB | P83C575EBBB | P87C575EBBB | OTP | 0 to +70, 44-Pin Plastic Quad Flat Pack | 16 | SOT307-2 |

**NOTE:**
1. OTP - One Time Programmable EPROM. UV - Erasable EPROM

## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

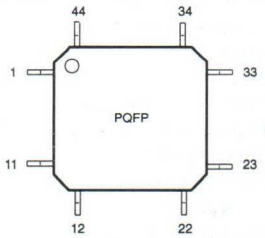



| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC* | 23 | NC* |
| 2 | T2/P1.0/CMP0+ | 24 | P2.0/A8 |
| 3 | T2EX/P1.1/CMP0– | 25 | P2.1/A9 |
| 4 | P1.2/ECI | 26 | P2.2/A10 |
| 5 | P1.3/CMP0/CEX0 | 27 | P2.3/A11 |
| 6 | P1.4/CMP1/CEX1 | 28 | P2.4/A12 |
| 7 | P1.5/CMP2/CEX2 | 29 | P2.5/A13 |
| 8 | P1.6/CMP3/CEX3 | 30 | P2.6/A14 |
| 9 | P1.7/CEX4 | 31 | P2.7/A15 |
| 10 | $\overline{RST}$ | 32 | $\overline{PSEN}$ |
| 11 | RxD/P3.0 | 33 | ALE/$\overline{PROG}$ |
| 12 | NC* | 34 | NC* |
| 13 | TxD/P3.1 | 35 | $\overline{EA}$/$V_{PP}$ |
| 14 | $\overline{INT0}$/P3.2 | 36 | P0.7/AD7 |
| 15 | $\overline{INT1}$/P3.3 | 37 | P0.6/AD6 |
| 16 | T0/P3.4/CMPR– | 38 | P0.5/AD5 |
| 17 | T1/P3.5/CMP1+ | 39 | P0.4/AD4 |
| 18 | $\overline{WR}$/P3.6/CMP2+ | 40 | P0.3/AD3 |
| 19 | $\overline{RD}$/P3.7/CMP3+ | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | $V_{SS}$ | 44 | $V_{CC}$ |

* DO NOT CONNECT

SU00235

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS

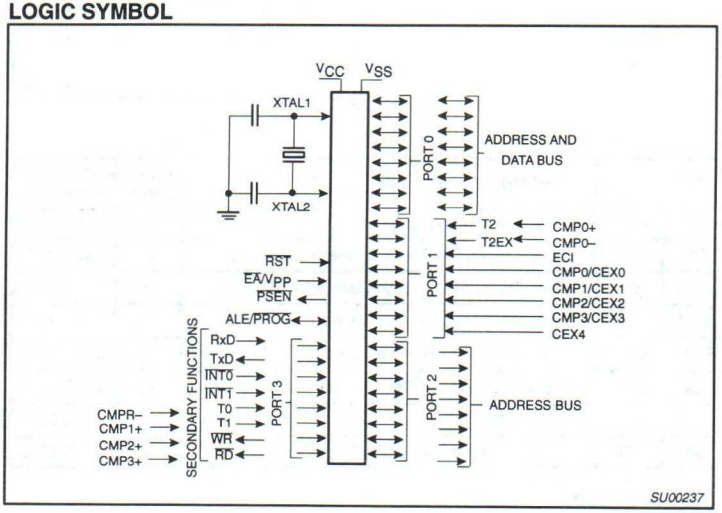



| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5/CMP2/CEX2 | 23 | P2.5/A13 |
| 2 | P1.6/CMP3/CEX3 | 24 | P2.6/A14 |
| 3 | P1.7/CEX4 | 25 | P2.7/A15 |
| 4 | $\overline{RST}$ | 26 | $\overline{PSEN}$ |
| 5 | RxD/P3.0 | 27 | ALE/$\overline{PROG}$ |
| 6 | NC* | 28 | NC* |
| 7 | TxD/P3.1 | 29 | $\overline{EA}$/$V_{PP}$ |
| 8 | $\overline{INT0}$/P3.2 | 30 | P0.7/AD7 |
| 9 | $\overline{INT1}$/P3.3 | 31 | P0.6/AD6 |
| 10 | T0/P3.4/CMPR– | 32 | P0.5/AD5 |
| 11 | T1/P3.5/CMP1+ | 33 | P0.4/AD4 |
| 12 | $\overline{WR}$/P3.6/CMP2+ | 34 | P0.3/AD3 |
| 13 | $\overline{RD}$/P3.7CMP3+ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | $V_{SS}$ | 38 | $V_{CC}$ |
| 17 | NC* | 39 | NC* |
| 18 | P2.0/A8 | 40 | T2/P1.0/CMP0+ |
| 19 | P2.1/A9 | 41 | T2EX/P1.1/CMP0– |
| 20 | P2.2/A10 | 42 | P1.2/ECI |
| 21 | P2.3/A11 | 43 | P1.3/CMP0/CEX0 |
| 22 | P2.4/A12 | 44 | P1.4/CMP1/CEX1 |

* DO NOT CONNECT

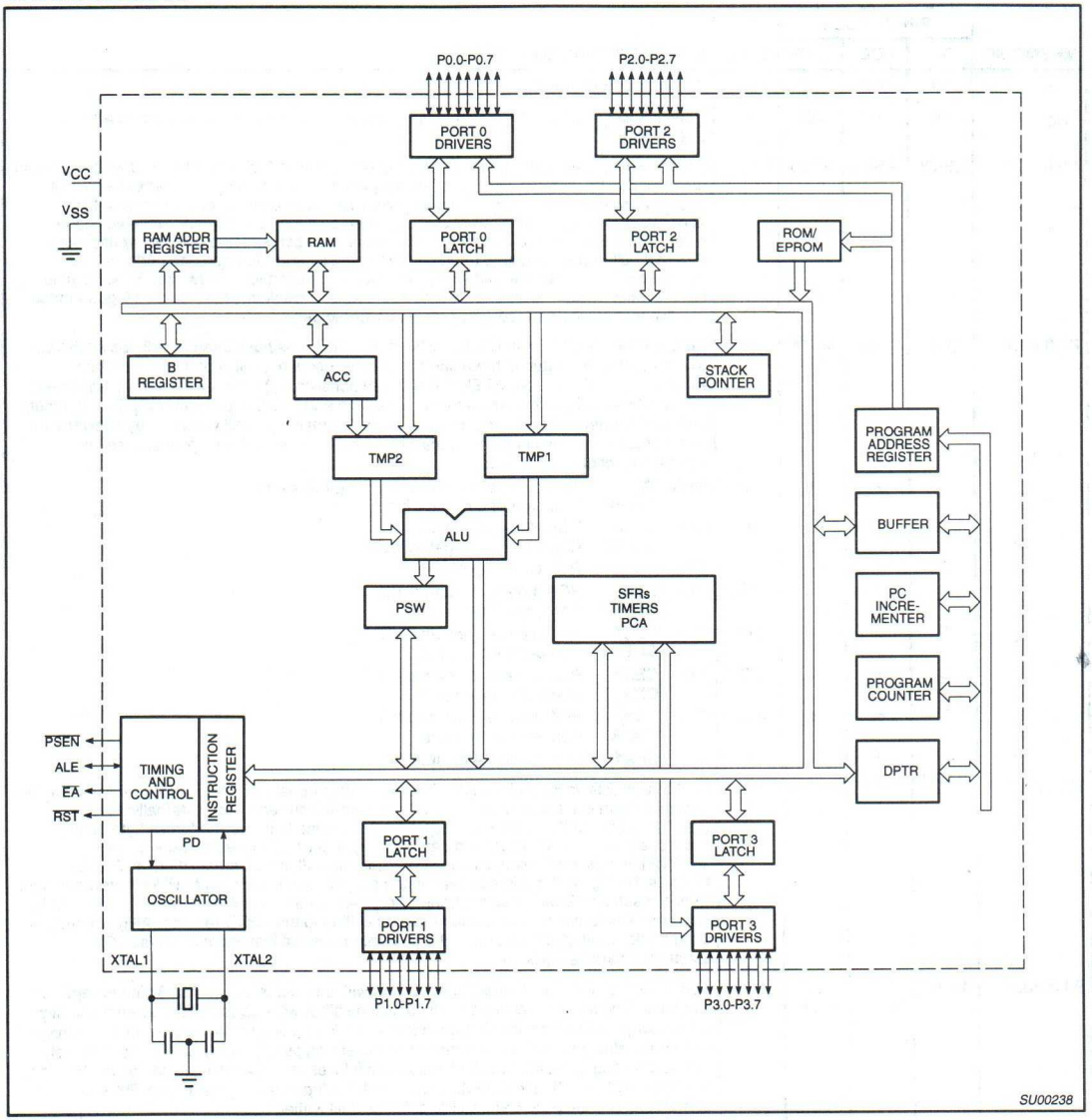
SU00236

## LOGIC SYMBOL




SU00237

## BLOCK DIAGRAM



## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER DIP | LCC | QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0-0.7 | 39-32 | 43-36 | 37-30 | I/O | **Port 0:** Port 0 is an open-drain bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also receives code bytes during EPROM programming and outputs code bytes during program verification. External pull-ups are required during program verification. During reset, port 0 will be asynchronously driven low and will remain low until written to by software. All port 0 pins have Schmitt trigger inputs with 200mV hysteresis. A weak pulldown on port 0 guarantees positive leakage current (see DC Electrical Characteristics: $I_{L1}$). |
| P1.0-P1.7 | 1-8 | 2-9 | 40-44 1-3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port. Port 1 pins have internal pull-ups such that pins that have 1s written to them can be used as inputs but will source current when externally pulled low (see DC Electrical Characteristics: $I_{IL}$). Port 1 receives the low-order address byte during program memory verification and EPROM programming. During reset, port 1 will be asynchronously driven low and will remain low until written to by software. All port 1 pins have Schmitt trigger inputs with 50mV hysteresis. Port 1 pins also serve alternate functions as follows: |
| | 1 | 2 | 40 | I/O | **P1.0 T2** Timer 2 external I/O – clockout (programmable)<br>**CMP0+** Comparator 0 positive input |
| | 2 | 3 | 41 | I | **P1.1 T2EX** Timer 2 capture input<br>**CMP0-** Comparator 0 negative input |
| | 3 | 4 | 42 | I | **P1.2 ECI** PCA count input |
| | 4 | 5 | 43 | I/O | **P1.3 CEX0** PCA module 0 external I/O<br>**CMP0** Comparator 0 output |
| | 5 | 6 | 44 | I/O | **P1.4 CEX1** PCA module 1 external I/O<br>**CMP1** Comparator 1 output |
| | 6 | 7 | 1 | I/O | **P1.5 CEX2** PCA module 2 external I/O<br>**CMP2** Comparator 2 output |
| | 7 | 8 | 2 | I/O | **P1.6 CEX3** PCA module 3 external I/O<br>**CMP3** Comparator 3 output |
| | 8 | 9 | 3 | I/O | **P1.7 CEX4** PCA module 4 external I/O |
| P2.0-P2.7 | 21-28 | 24-31 | 18-25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them can be used as inputs, but will source current when externally pulled low (see DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during accesses to external program and data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. Port 2 receives the high-order address byte during program verification and EPROM programming. During reset, port 2 will be asynchronously driven low and will remain low until written to by software. Port 2 can be made open drain by writing to the P2OD register (AIH). In open drain mode, weak pulldowns on port 2 guarantee positive leakage current (see DC Electrical Characteristics $I_{L1}$). |
| P3.0-P3.7 | 10-17 | 11, 13-19 | 5, 7-13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins except P3.1 that have 1s written to them can be used as inputs but will source current when externally pulled low (see DC Electrical Characteristics: $I_{IL}$). P3.1 will be a high impedance pin except while transmitting serial data, in which case the strong pull-up will remain on continuously when outputting a 1 level. The P3.1 output drive level when transmitting can be set to one of two levels by the writing to the P3.1 register bit. During reset all pins (except P3.1) will be asynchronously driven low and will remain low until written to by software. All port 3 pins have Schmitt trigger inputs with 200mV hysteresis, except P3.2 and P3.3, which have 50mV hysteresis. Port 3 pins serve alternate functions as follows: |

**PIN DESCRIPTIONS** (Continued)

| MNEMONIC | PIN NUMBER DIP | LCC | QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| | | | | | **Port 3:** (continued) |
| | 10 | 11 | 5 | I | **P3.0 RxD** Serial receive port |
| | 11 | 13 | 7 | O | **P3.1 TxD** Serial transmit port enabled only when transmitting serial data |
| | 12 | 14 | 8 | I | **P3.2 $\overline{INT0}$** External interrupt 0 |
| | 13 | 15 | 9 | I | **P3.3 $\overline{INT1}$** External interrupt 1 |
| | 14 | 16 | 10 | I | **P3.4 T0** Timer/counter 0 input<br>**CMPR-** Common - reference to comparators 1, 2, 3 |
| | 15 | 17 | 11 | I | **P3.5 T1** Timer/counter 1 input<br>**CMP1+** Comparator 1 positive input |
| | 16 | 18 | 12 | O | **P3.6 $\overline{WR}$** External data memory write strobe<br>**CMP2+** Comparator 2 positive input |
| | 17 | 19 | 13 | O | **P3.7 $\overline{RD}$** External data memory read strobe<br>**CMP3+** Comparator 3 positive input |
| $\overline{RST}$ | 9 | 10 | 4 | I | **Reset:** A low on this pin asynchronously resets all port pins to a low state except P3.1. The pin must be held low with the oscillator running for 24 oscillator cycles to initialize the internal registers. An internal diffused resistor to $V_{CC}$ permits a power on reset using only an external capacitor to $V_{SS}$. RST has a Schmitt trigger input stage to provide additional noise immunity with a slow rising input voltage. |
| ALE/$\overline{PROG}$ | 30 | 33 | 27 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. ALE is switched off if the bit 0 in the AUXR register (8EH) is set. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the device is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}$/$V_{PP}$ | 31 | 35 | 29 | I | **External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H to 1FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 1FFFH. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**Table 1. 87C575 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| AUXR# | Auxiliary | 8EH | – | – | – | – | – | – | LO | AO | xxxxxx00B |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| CCAP0H# | Module 0 Capture High | FAH | | | | | | | | | xxxxxxxxB |
| CCAP1H# | Module 1 Capture High | FBH | | | | | | | | | xxxxxxxxB |
| CCAP2H# | Module 2 Capture High | FCH | | | | | | | | | xxxxxxxxB |
| CCAP3H# | Module 3 Capture High | FDH | | | | | | | | | xxxxxxxxB |
| CCAP4H# | Module 4 Capture High | FEH | | | | | | | | | xxxxxxxxB |
| CCAP0L# | Module 0 Capture Low | EAH | | | | | | | | | xxxxxxxxB |
| CCAP1L# | Module 1 Capture Low | EBH | | | | | | | | | xxxxxxxxB |
| CCAP2L# | Module 2 Capture Low | ECH | | | | | | | | | xxxxxxxxB |
| CCAP3L# | Module 3 Capture Low | EDH | | | | | | | | | xxxxxxxxB |
| CCAP4L# | Module 4 Capture Low | EEH | | | | | | | | | xxxxxxxxB |
| CCAPM0# | Module 0 Mode | DAH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM1# | Module 1 Mode | DBH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM2# | Module 2 Mode | DCH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM3# | Module 3 Mode | DDH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM4# | Module 4 Mode | DEH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| CCON*# | PCA Counter Control | D8H | CF | CR | – | CCF4 | CCF3 | CCF2 | CCF1 | CCF0 | 00x00000B |
| CH# | PCA Counter High | F9H | | | | | | | | | 00H |
| CL# | PCA Counter Low | E9H | | | | | | | | | 00H |
| CMOD# | PCA Counter Mode | D9H | CIDL | WDTE | – | – | – | CPS1 | CPS0 | ECF | 00xxx000B |
| | | | EF | EE | ED | EC | EB | EA | E9 | E8 | |
| CMP*# | Comparator | E8H | EC3DP | EC2DP | EC1DP | EC0DP | C3RO | C2RO | C1RO | C0RO | 00H |
| CMPE# | Comparator Enable | 91H | EC3TDC | EC2TDC | EC1TDC | EC0TDC | EC3OD | EC2OD | EC1OD | EC0OD | 00H |
| DPTR: | Data Pointer (2 bytes) | | | | | | | | | | |
| DPH | Data Pointer High | 83H | | | | | | | | | 00H |
| DPL | Data Pointer Low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE* | Interrupt Enable | A8H | EA | EC | ET2 | ES | ET1 | EX1 | ET0 | EX0 | 00H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP* | Interrupt Priority | B8H | – | PPC | PT2 | PS | PT1 | PX1 | PT0 | PX0 | x0000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | 00H |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | CEX4 | CEX3 | CEX2 | CEX1 | CEX0 | EXI | T2EX | T2 | 00H |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | AD15 | AD14 | AD13 | AD12 | AD11 | AD10 | AD9 | AD8 | 00H |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TxD | RxD | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

**Table 1. 87C575 Special Function Registers** (Continued)

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| P2OD# | Port 2 Pullup Disable | A1H | | | | | | | | | 00H |
| PCON# | Power Control | 87H | SMOD1 | SMOD0 | OSF[1] | POF[1] | LVF[1] | GF0 | PD | IDL | 00xxx000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program Status Word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| RACAP2H# | Timer 2 Capture High | CBH | | | | | | | | | 00H |
| RACAP2L# | Timer 2 Capture Low | CAH | | | | | | | | | 00H |
| SADDR# | Slave Address | A9H | | | | | | | | | 00H |
| SADEN# | Slave Address Mask | B9H | | | | | | | | | 00H |
| SBUF | Serial Data Buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial Control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack Pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer Control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| T2CON* | Timer 2 Control | C8H | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/T2 | CP/RL2 | 00H |
| T2MOD# | Timer 2 Mode Control | C9H | – | – | – | – | – | – | T2OE[2] | DCEN | xxxxxxx0B |
| TH0 | Timer High 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer High 1 | 8DH | | | | | | | | | 00H |
| TH2# | Timer High 2 | CDH | | | | | | | | | 00H |
| TL0 | Timer Low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer Low 1 | 8BH | | | | | | | | | 00H |
| TL2# | Timer Low 2 | CCH | | | | | | | | | 00H |
| TMOD | Timer Mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |
| | | | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 | |
| WDCON*# | Watchdog Timer Control | C0H | PRE2 | PRE1 | PRE0 | LVRE | OFRE | WDRUN | WDTOF | WDMOD | 11111101B |
| WDL# | Watchdog Timer Reload | C1H | | | | | | | | | 00H |
| WFEED1# | Watchdog Feed 1 | C2H | | | | | | | | | xxH |
| WFEED2# | Watchdog Feed 2 | C3H | | | | | | | | | xxH |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
1. Reset value depends on reset source.
2. Programmable clock-out.

## POWER ON CLEAR/ POWER ON FLAG

An on-chip Power On Detect Circuit resets the 8XC575 and sets the Power Off Flag (PCON.4) on power up or if $V_{CC}$ drops to zero momentarily. The POF can only be cleared by software. The RST pin is not driven by the power on detect circuit. The POF can be read by software to determine that a power failure has occurred and can also be set by software.

## LOW VOLTAGE DETECT

An on-chip Low Voltage Detect circuit sets the Low Voltage Flag (PCON.3) if $V_{CC}$ drops below $V_{LOW}$ (see DC Electrical Characteristics) and resets the 8XC575 if the Low Voltage Reset Enable bit (WDCON.4) is set. If the LVRE is cleared, the reset is disabled but LVF will still be set if $V_{CC}$ is low. The RST pin is not driven by the low voltage detect circuit. The LVF can be read by software to determine that $V_{CC}$ was low. The LVF can be set or cleared by software.

## OSCILLATOR FAIL DETECT

An on-chip Oscillator Fail Detect circuit sets the Oscillator Fail Flag (PCON.5) if the oscillator frequency drops below OSCF for one or more cycles (see AC Electrical Characteristics: OSCF) and resets the 8XC575 if the Oscillator Fail Reset Enable bit (WDCON.3) is set. If OFRE is cleared, the reset is disabled but OSF will still be set if the oscillator fails. The RST pin is not driven by the oscillator fail detect circuit. The OSF can be read by software to determine that an oscillator failure has occurred. The OSF can be set or cleared by software.

## LOW ACTIVE RESET

One of the most notable features on this part is the low active reset. At this time this is the only 80C51 derivative available that has low active reset. This feature makes it easier to interface the 8XC575 into an application to accommodate the power-on and low voltage conditions that can occur. The low active reset operates exactly the same as high active reset with the exception that the part is put into the reset mode by applying a low level to the reset pin. For power-on reset it is also necessary to invert the power-on reset circuit; connecting the 8.2K resistor from the reset pin to $V_{CC}$ and the 10µf capacitor from the reset pin to ground. Figure 1 shows all of the reset related circuitry.

When reset the port pins on the 87C575 are driven low asynchronously. This is different from all other 80C51 derivatives.

The 8XC575 also has Low voltage detection circuitry that will, if enabled, force the part to reset when $V_{CC}$ (on the part) fails below a set level. Low Voltage Reset is enabled by a normal reset. Low Voltage Reset can be disabled by clearing LVRE (bit 4 in the WDCON SFR) then executing a watchdog feed sequence (A5H to WFEED1 followed immediately by 5A to WFEED2). In addition there is a flag (LVF) that is set if a low voltage condition is detected. The LVF flag is set even if the Low Voltage detection circuitry is disabled. Notice that the Low voltage detection circuitry does not drive the RST# pin so the LVF flag is the only way that the microcontroller can determine if it has been reset due to a low voltage condition.

The 8XC575 has an on-chip power-on detection circuit that sets the POF (PCON.4) flag on power up or if the $V_{CC}$ level momentarily drops to 0V. This flag can be used to determine if the part is being started from a power-on (cold start) or if a reset has occurred due to another condition (warm start).

## TIMERS

The 87C575 has four on-chip timers.

Timers 0 and 1 are identical in every way to Timers 0 and 1 on the 80C51.

Timer 2 on the 8XC575 is identical to the 80C52 Timer 2 (described in detail in the 80C52 overview) with the exception that it is an up or down counter. To configure the Timer to count down the DCEN bit in the T2MOD special function register must be set and a low level must be present on the T2EX pin (P1.1).

The Watchdog timer operation and implementation is the same as that for the 8XC550 (described in the 8XC550 overview) with the exception that the reset values of the WDCON and WDL special function registers have been changed. The changes in these registers cause the watchdog timer to be enabled with a timeout of 98304 × $T_{OSC}$ when the part is reset. The watchdog can be disabled by executing a valid feed sequence and then clearing WDRUN (bit 2 in the WDCON SFR).

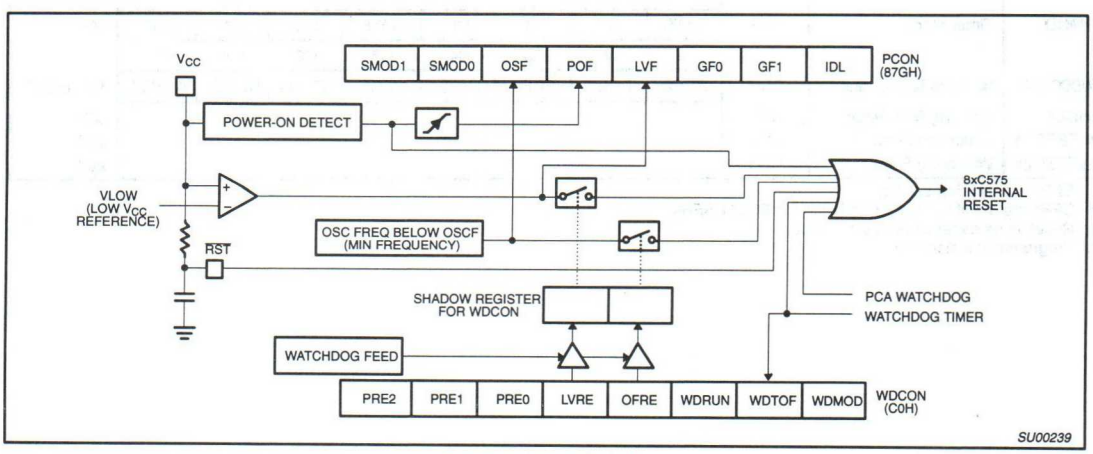



**Figure 1. Reset Circuitry**

## PROGRAMMABLE COUNTER ARRAY (PCA)

The Programmable Counter Array is a special Timer that has five 16-bit capture/compare modules associated with it. Each of the modules can be programmed to operate in one of four modes: rising and/or falling edge capture, software timer, high-speed output, or pulse width modulator. Each module has a pin associated with it in port 1. Module 0 is connected to P1.3(CEX0), module 1 to P1.4(CEX1), etc.. The basic PCA configuration is shown in Figure 2.

The PCA timer is a common time base for all five modules and can be programmed to run at: 1/12 the oscillator frequency, 1/4 the oscillator frequency, the Timer 0 overflow, or the input on the ECI pin (P1.2). The timer count source is determined from the CPS1 and CPS0 bits in the CMOD SFR as follows (see Figure 3):

| CPS1 | CPS0 | PCA Timer Count Source |
|---|---|---|
| 0 | 0 | 1/12 oscillator frequency |
| 0 | 1 | 1/4 oscillator frequency |
| 1 | 0 | Timer 0 overflow |
| 1 | 1 | External Input at ECI pin |

In the CMOD SFR are three additional bits associated with the PCA. They are CIDL which allows the PCA to stop during idle mode, WDTE which enables or disables the watchdog function on module 4, and ECF which when set causes an interrupt and the PCA overflow flag CF (in the CCON SFR) to be set when the PCA timer overflows. These functions are shown in Figure 3.

The watchdog timer function is implemented in module 4 as implemented in other parts that have a PCA that are available on the market. However, if a watchdog timer is required in the target application, it is recommended to use the hardware watchdog timer that is implemented on the 87C575 separately from the PCA (see Figure 14).

The CCON SFR contains the run control bit for the PCA and the flags for the PCA timer (CF) and each module (refer to Figure 6). To run the PCA the CR bit (CCON.6) must be set by software. The PCA is shut off by clearing this bit. The CF bit (CCON.7) is set when the PCA counter overflows and an interrupt will be generated if the ECF bit in the CMOD register is set, The CF bit can only be cleared by software. Bits 0 through 4 of the CCON register are the flags for the modules (bit 0 for module 0, bit 1 for module 1, etc.) and are set by hardware when either a match or a capture occurs. These flags also can only be cleared by software. The PCA interrupt system shown in Figure 4.

Each module in the PCA has a special function register associated with it. These registers are: CCAPM0 for module 0, CCAPM1 for module 1, etc. (see Figure 7). The registers contain the bits that control the mode that each module will operate in. The ECCF bit (CCAPMn.0 where n=0, 1, 2, 3, or 4 depending on the module) enables the CCF flag in the CCON SFR to generate an interrupt when a match or compare occurs in the associated module. PWM (CCAPMn.1) enables the pulse width modulation mode. The TOG bit (CCAPMn.2) when set causes the CEX output associated with the module to toggle when there is a match between the PCA counter and the module's capture/compare register. The match bit MAT (CCAPMn.3) when set will cause the CCFn bit in the CCON register to be set when there is a match between the PCA counter and the module's capture/compare register.

The next two bits CAPN (CCAPMn.4) and CAPP (CCAPMn.5) determine the edge that a capture input will be active on. The CAPN bit enables the negative edge, and the CAPP bit enables the positive edge. If both bits are set both edges will be enabled and a capture will occur for either transition. The last bit in the register ECOM (CCAPMn.6) when set enables the comparator function. Figure 8 shows the CCAPMn settings for the various PCA functions.

There are two additional registers associated with each of the PCA modules. They are CCAPnH and CCAPnL and these are the registers that store the 16-bit count when a capture occurs or a compare should occur. When a module is used in the PWM mode these registers are used to control the duty cycle of the output.

### PCA Capture Mode

To use one of the PCA modules in the capture mode either one or both of the CCAPM bits CAPN and CAPP for that module must be set. The external CEX input for the module (on port 1) is sampled for a transition. When a valid transition occurs the PCA hardware loads the value of the PCA counter registers (CH and CL) into the module's capture registers (CCAPnL and CCAPnH). If the CCFn bit for the module in the CCON SFR and the ECCFn bit in the CCAPMn SFR are set then an interrupt will be generated. Refer to Figure 9.

### 16-bit Software Timer Mode

The PCA modules can be used as software timers by setting both the ECOM and MAT bits in the modules CCAPMn register. The PCA timer will be compared to the module's capture registers and when a match occurs an interrupt will occur if the CCFn (CCON SFR) and the ECCFn (CCAPMn SFR) bits for the module are both set (see Figure 10).

### High Speed Output Mode

In this mode the CEX output (on port 1) associated with the PCA module will toggle each time a match occurs between the PCA counter and the module's capture registers. To activate this mode the TOG, MAT, and ECOM bits in the module's CCAPMn SFR must be set (see Figure 11).

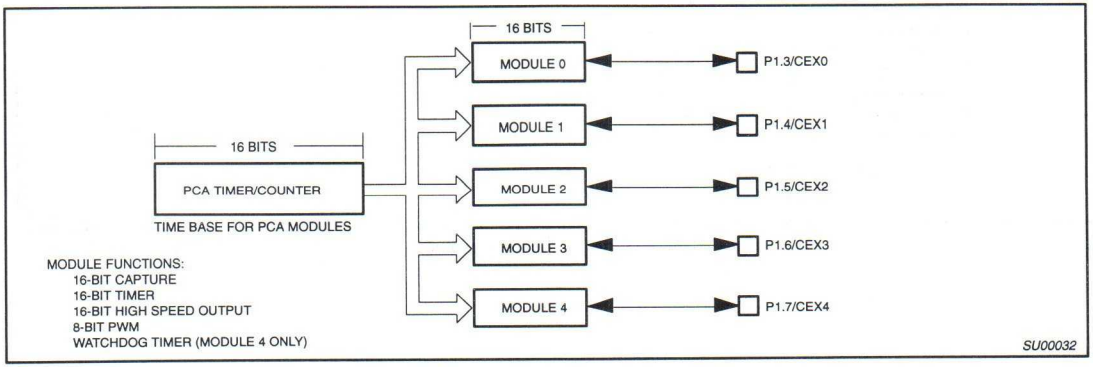



**Figure 2. Programmable Counter Array (PCA)**

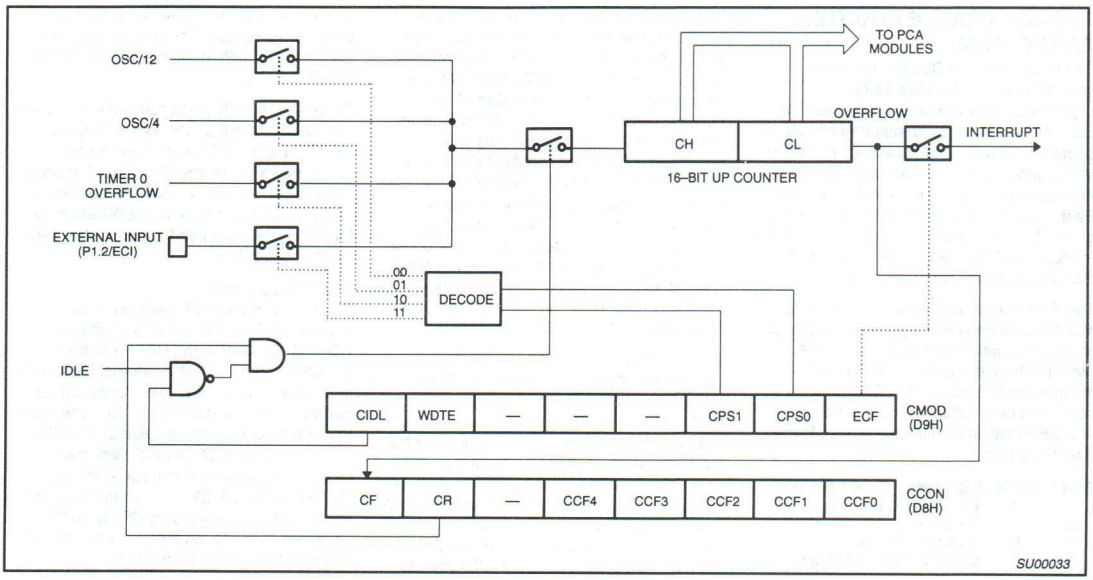



**Figure 3. PCA Timer/Counter**

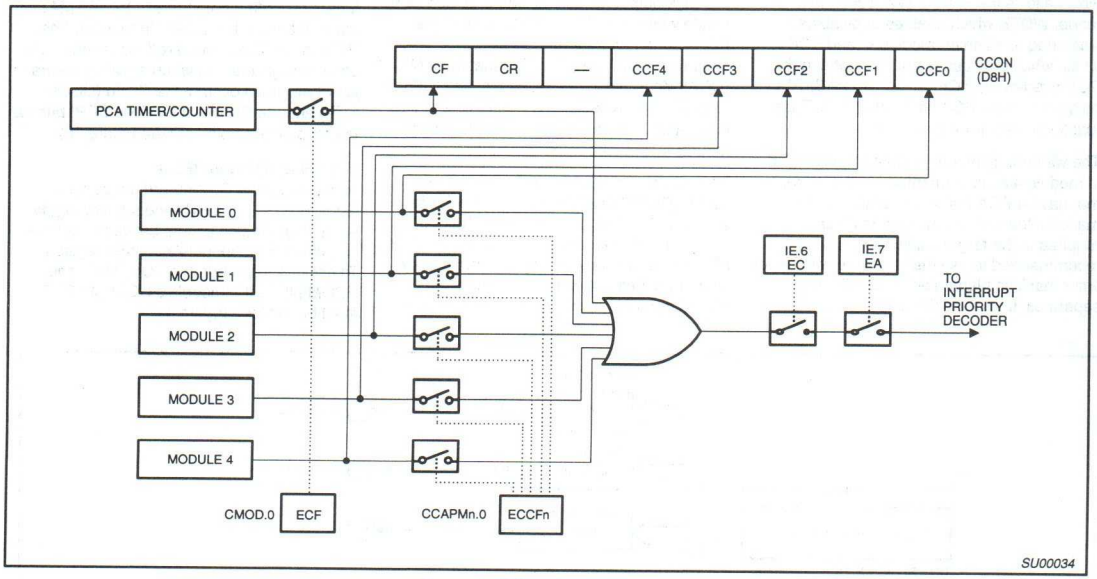



**Figure 4. PCA Interrupt System**

**CMOD** Address = OD9H Reset Value = 00XX X000B

Bit Addressable

| CIDL | WDTE | – | – | – | CPS1 | CPS0 | ECF |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Bit:

| Symbol | Function |
|---|---|
| **CIDL** | Counter Idle control: CIDL = 0 programs the PCA Counter to continue functioning during idle Mode. CIDL = 1 programs it to be gated off during idle. |
| **WDTE** | Watchdog Timer Enable: WDTE = 0 disables Watchdog Timer function on PCA Module 4. WDTE = 1 enables it. |
| – | Not implemented, reserved for future use.* |
| **CPS1** | PCA Count Pulse Select bit 1. |
| **CPS0** | PCA Count Pulse Select bit 0. |

| CPS1 | CPS0 | Selected PCA Input** | |
|---|---|---|---|
| 0 | 0 | 0 | Internal clock, $f_{OSC} \div 12$ |
| 0 | 1 | 1 | Internal clock, $f_{OSC} \div 4$ |
| 1 | 0 | 2 | Timer 0 overflow |
| 1 | 1 | 3 | External clock at ECI/P1.2 pin (max. rate = $f_{OSC} \div 8$) |

| | |
|---|---|
| **ECF** | PCA Enable Counter Overflow interrupt: ECF = 1 enables CF bit in CCON to generate an interrupt. ECF = 0 disables that function of CF. |

**NOTE:**
* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.
** $f_{OSC}$ = oscillator frequency

*SU00035*

**Figure 5. CMOD: PCA Counter Mode Register**

**CCON** Address = OD8H Reset Value = 00X0 0000B

Bit Addressable

| CF | CR | – | CCF4 | CCF3 | CCF2 | CCF1 | CCF0 |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Bit:

| Symbol | Function |
|---|---|
| **CF** | PCA Counter Overflow flag. Set by hardware when the counter rolls over. CF flags an interrupt if bit ECF in CMOD is set. CF may be set by either hardware or software but can only be cleared by software. |
| **CR** | PCA Counter Run control bit. Set by software to turn the PCA counter on. Must be cleared by software to turn the PCA counter off. |
| – | Not implemented, reserved for future use*. |
| **CCF4** | PCA Module 4 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF3** | PCA Module 3 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF2** | PCA Module 2 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF1** | PCA Module 1 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF0** | PCA Module 0 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |

**NOTE:**
* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

*SU00036*

**Figure 6. CCON: PCA Counter Control Register**

**CCAPMn** Address CCAPM0 0DAH
CCAPM1 0DBH
CCAPM2 0DCH
CCAPM3 0DDH
CCAPM4 0DEH

Reset Value = X000 0000B

Not Bit Addressable

| | – | ECOMn | CAPPn | CAPNn | MATn | TOGn | PWMn | ECCFn |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Symbol | Function |
|---|---|
| – | Not implemented, reserved for future use*. |
| **ECOMn** | Enable Comparator. ECOMn = 1 enables the comparator function. |
| **CAPPn** | Capture Positive, CAPPn = 1 enables positive edge capture. |
| **CAPNn** | Capture Negative, CAPNn = 1 enables negative edge capture. |
| **MATn** | Match. When MATn = 1, a match of the PCA counter with this module's compare/capture register causes the CCFn bit in CCON to be set, flagging an interrupt. |
| **TOGn** | Toggle. When TOGn = 1, a match of the PCA counter with this module's compare/capture register causes the CEXn pin to toggle. |
| **PWMn** | Pulse Width Modulation Mode. PWMn = 1 enables the CEXn pin to be used as a pulse width modulated output. |
| **ECCFn** | Enable CCF interrupt. Enables compare/capture flag CCFn in the CCON register to generate an interrupt. |

**NOTE:**
*User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

*SU00037*

**Figure 7. CCAPMn: PCA Modules Compare/Capture Registers**

| – | ECOMn | CAPPn | CAPNn | MATn | TOGn | PWMn | ECCFn | MODULE FUNCTION |
|---|---|---|---|---|---|---|---|---|
| X | 0 | 0 | 0 | 0 | 0 | 0 | 0 | No operation |
| X | X | 1 | 0 | 0 | 0 | 0 | X | 16-bit capture by a positive-edge trigger on CEXn |
| X | X | 0 | 1 | 0 | 0 | 0 | X | 16-bit capture by a negative trigger on CEXn |
| X | X | 1 | 1 | 0 | 0 | 0 | X | 16-bit capture by a transition on CEXn |
| X | 1 | 0 | 0 | 1 | 0 | 0 | X | 16-bit Software Timer |
| X | 1 | 0 | 0 | 1 | 1 | 0 | X | 16-bit High Speed Output |
| X | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 8-bit PWM |
| X | 1 | 0 | 0 | 1 | X | 0 | X | Watchdog Timer |

**Figure 8. PCA Module Modes (CCAPMn Register)**

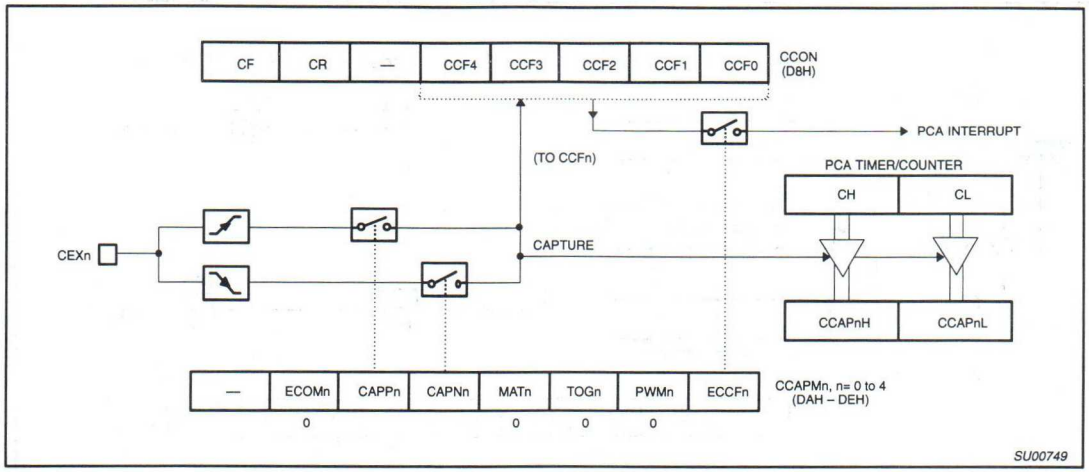



**Figure 9. PCA Capture Mode**

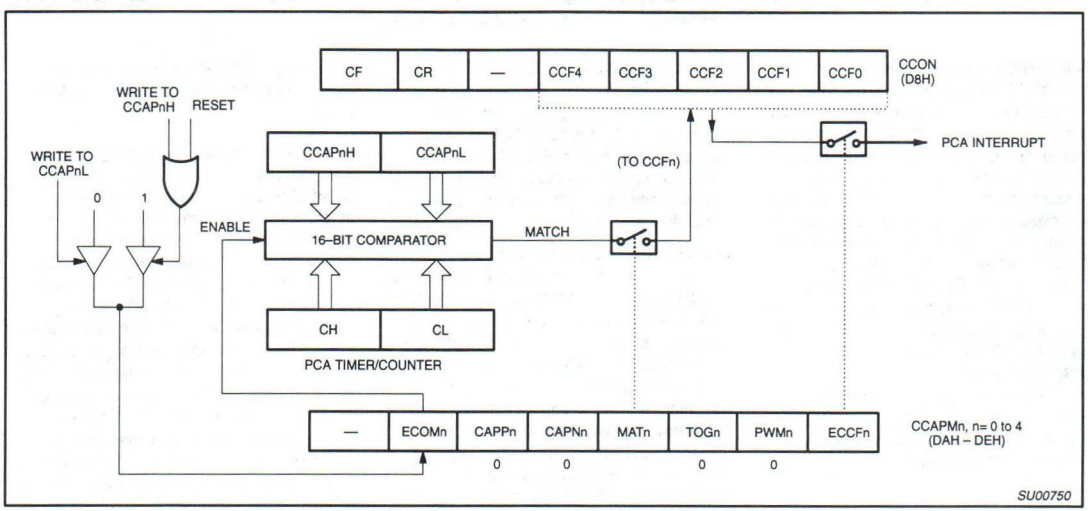



**Figure 10. PCA Compare Mode**

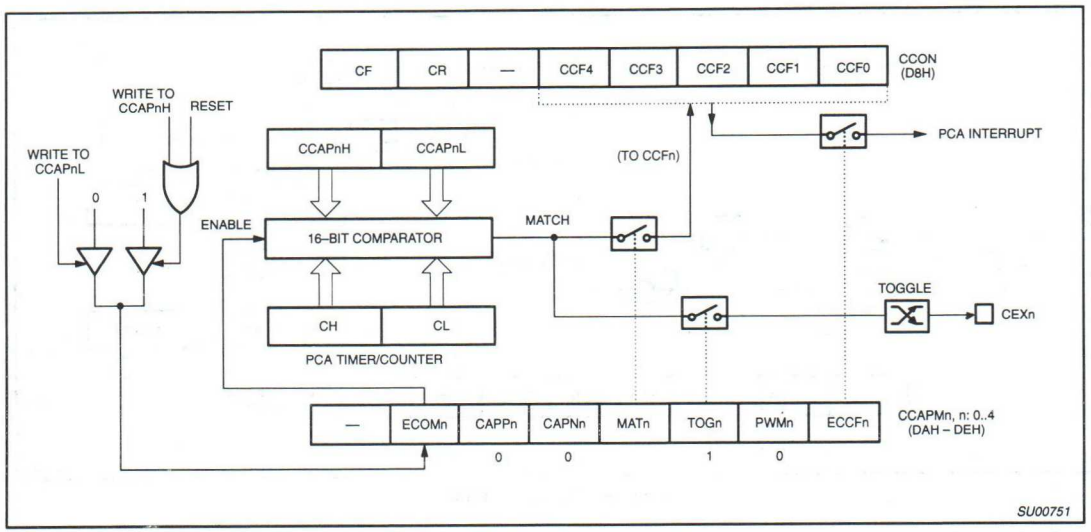



**Figure 11. PCA High Speed Output Mode**

**Pulse Width Modulator Mode**

All of the PCA modules can be used as PWM outputs. Figure 12 shows the PWM function. The frequency of the output depends on the source for the PCA timer. All of the modules will have the same frequency of output because they all share the PCA timer. The duty cycle of each module is independently variable using the module's capture register CCAPLn. When the value of the PCA CL SFR is less than the value in the module's CCAPLn SFR the output will be low, when it is equal to or greater than the output will be high. When CL overflows from FF to 00, CCAPLn is reloaded with the value in CCAPHn. the allows updating the PWM without glitches. The PWM and ECOM bits in the module's CCAPMn register must be set to enable the PWM mode.

## WATCHDOG TIMER

The watchdog timer is not directly loadable by the user. Instead, the value to be loaded into the main timer is held in an autoload register or is part of the mask ROM programming. In order to cause the main timer to be loaded with the appropriate value, a special sequence of software action must take place. This operation is referred to as feeding the watchdog timer.

To feed the watchdog, two instructions must be sequentially executed successfully. No intervening instruction fetches are allowed, so interrupts should be disabled before feeding the watchdog. The instructions should move A5H to the WFEED1 register and then 5AH to the WFEED2 register. If WFEED1 is correctly loaded and WFEED2 is not correctly loaded, then an immediate underflow will occur.

The watchdog timer subsystem has two modes of operation. Its principal function is a watchdog timer. In this mode it protects the system from incorrect code execution by causing a system reset when the watchdog timer underflows as a result of a failure of software to feed the timer prior to the timer reaching its terminal count. If the user does not employ the watchdog function, the watchdog subsystem can be used as a timer. In this mode, reaching the terminal count sets a flag. In most other respects, the timer mode possesses the characteristics of the watchdog mode. This is done to protect the integrity of the watchdog function.

The watchdog timer subsystem consists of a prescaler and a main counter. The prescaler has 8 selectable taps off the final stages and the output of a selected tap provides the clock to the main counter. The main counter is the section that is loaded as a result of the software feeding the watchdog and it is the section that causes the system reset (watchdog mode) or time-out flag to be set (timer mode) if allowed to reach its terminal count.

### Programming the Watchdog Timer

Both the EPROM and ROM devices have a set of SFRs for holding the watchdog autoload values and the control bits. The watchdog time-out flag is present in the watchdog control register and operates the same in all versions. In the EPROM device, the watchdog parameters (autoload value and control) are always taken from the SFRs. In the ROM device, the watchdog parameters can be mask programmed or taken from the SFRs. The selection to take the watchdog parameters from the SFRs or from the mask programmed values is controlled by EA (external access). When EA is high (internal ROM access), the watchdog parameters are taken from the mask programmed values. If the watchdog is mask programmed to the timer mode, then the autoload values and the pre-scaler taps are taken from the SFRs. When EA is low (external access), the watchdog parameters are taken from the SFRs. The user should be able to leave code in his program which initializes the watchdog SFRs even though he has migrated to the mask ROM part. This allows no code changes from EPROM prototyping to ROM coded production parts.

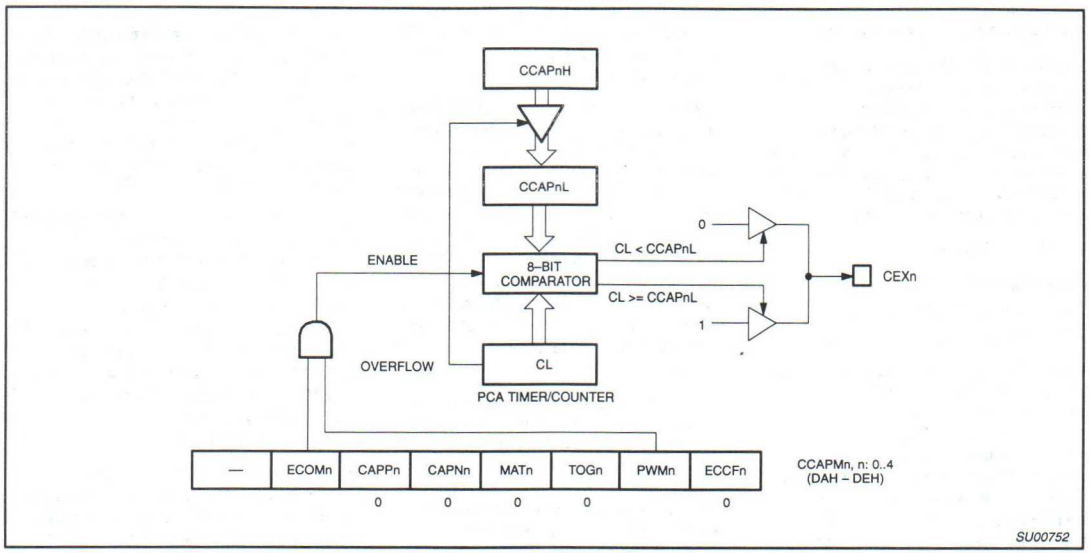



**Figure 12. PCA PWM Mode**

## Watchdog Detailed Operation

**EPROM Device (and ROMless Operation: $\overline{EA}$ = 0)**

In the ROMless operation (ROM part, $\overline{EA}$ = 0) and in the EPROM device, the watchdog operates in the following manner (see Figure 14).

Whether the watchdog is in the watchdog or timer mode, when external RESET is applied, the following takes place:

- Watchdog mode bit set to watchdog mode.
- Watchdog run control bit set to ON.
- Autoload register set to 00 (min. count).
- Watchdog time-out flag cleared.
- Prescaler is cleared.
- Prescaler tap set to the highest divide.
- Autoload takes place.

The watchdog can be fed even though it is in the timer mode.

Note that the operational concept is for the watchdog mode of operation, when coming out of a hardware reset, the software should load the autoload registers, set the mode to watchdog, and then feed the watchdog (cause an autoload). The watchdog will now be starting at a known point.

If the watchdog is in the watchdog mode and running and happens to underflow at the time the external RESET is applied, the watchdog time-out flag will be cleared.

When the watchdog is in the watchdog mode and the watchdog underflows, the following action takes place (see Figure 16):

- Autoload takes place.
- Watchdog time-out flag is set
- Mode bit unchanged.
- Watchdog run bit unchanged.
- Autoload register unchanged.
- Prescaler tap unchanged.
- All other device action same as external reset.

Note that if the watchdog underflows, the program counter will start from 00H as in the case of an external reset. The watchdog time-out flag can be examined to determine if the watchdog has caused the reset condition. The watchdog time-out flag bit can be cleared by software.

When the watchdog is in the timer mode and the timer software underflows, the following action takes place:

- Autoload takes place.
- Watchdog time-out flag is set
- Mode bit unchanged.
- Watchdog run bit unchanged.
- Autoload register unchanged.
- Prescaler tap unchanged.

**Mask ROM Device ($\overline{EA}$ = 1)**

In the mask ROM device, the watchdog mode bit (WDMOD) is mask programmed and the bit in the watchdog command register is read only and reflects the mask programmed selection. If the mask programmed mode bit selects the timer mode, then the watchdog run bit (WDRUN) operates as described under EPROM Device. If the mask programmed bit selects the watchdog mode, then the watchdog run bit has no effect on the timer operation (see Figure 15).

**Watchdog Function**

The watchdog consists of a programmable prescaler and the main timer. The prescaler derives its clock from the on-chip oscillator. The prescaler consists of a divide by 12 followed by a 13 stage counter with taps from stage 6 through stage 13. This is shown in Figure 17. The tap selection is programmable. The watchdog main counter is a down counter clocked (decremented) each time the programmable prescaler underflows. The watchdog generates an underflow signal (and is autoloaded) when the watchdog is at count 0 and the clock to decrement the watchdog occurs. The watchdog is 8 bits long and the autoload value can range from 0 to FFH. (The

autoload value of 0 is permissible since the prescaler is cleared upon autoload).

This leads to the following user design equations. Definitions :$t_{OSC}$ is the oscillator period, N is the selected prescaler tap value, W is the main counter autoload value, $t_{MIN}$ is the minimum watchdog time-out value (when the autoload value is 0), $t_{MAX}$ is the maximum time-out value (when the autoload value is FFH), $t_D$ is the design time-out value.

$t_{MIN} = t_{OSC} \times 12 \times 64$

$t_{MAX} = t_{MIN} \times 128 \times 256$

$t_D = t_{MIN} \times 2^{PRESCALER} \times (W + 1)$
(where prescaler = 0, 1, 2, 3, 4, 5, 6, or 7)

Note that the design procedure is anticipated to be as follows. A $t_{MAX}$ will be chosen either from equipment or operation considerations and will most likely be the next convenient value higher than $t_D$. (If the watchdog were inadvertently to start from FFH, an overflow would be guaranteed, barring other anomalies, to occur within $t_{MAX}$). Then the value for the prescaler would be chosen from:

prescaler = log2 ($t_{MAX}$ / ($t_{OSC} \times 12 \times 256$)) - 6

This then also fixes $t_{MIN}$. An autoload value would then be chosen from:

$W = t_D / t_{MIN} - 1$

The software must be written so that a feed operation takes place every $t_D$ seconds from the last feed operation. Some tradeoffs may need to be made. It is not advisable to include feed operations in minor loops or in subroutines unless the feed operation is a specific subroutine.

**Watchdog Control Register (WDCON) (Bit Addressable) Address C0**

The following bits of this register are read only in the ROM part when $\overline{EA}$ is high: WDMOD, PRE0, PRE1, and PRE2. That is, the register will reflect the mask programmed values. In the ROM part with $\overline{EA}$ high, these bits are taken from mask coded bits and are not readable by the program. WDRUN is read only in the ROM part when $\overline{EA}$ is high and WDMOD is in the watchdog mode. When WDMOD is in the timer mode, WDRUN functions normally.

The parameters written into WDMOD, PRE0, PRE1, and PRE2 by the program are not applied directly to the watchdog timer subsystem. The watchdog timer subsystem is directly controlled by a second register which stores these bits. The transfer of these bits from the user register (WDMOD) to the second control register takes place when the watchdog is fed. This prevents random code execution from directly foiling the watchdog function. This does not affect the operation where these bits are taken from mask coded values.

The reset values of the WDCON and WDL registers will be such that the timer resets to the watchdog mode with a timeout period of $12 \times 64 \times 128 \times t_{OSC}$. The watchdog timer will not generate an interrupt. Additional bits in WDCON are used to disable reset generation by the oscillator fail and low voltage detect circuits. WDCON can be written by software only by executing a valid watchdog feed sequence.

**WDCON Register Bit Definitions**

| | | |
|---|---|---|
| WDCON.7 | PRE2 | Prescaler Select 2, reset to 1 |
| WDCON.6 | PRE1 | Prescaler Select 1, reset to 1 |
| WDCON.5 | PRE0 | Prescaler Select 0, reset to 1 |
| WDCON.4 | LVRE | Low Voltage Reset Enable, reset to 1 (enabled) |
| WDCON.3 | OFRE | Oscillator Fail Reset Enable, reset to 1 (enabled) |
| WDCON.2 | WDRUN | Watchdog Run, reset to 1 (enabled) |
| WDCON.1 | WDTOF | Watchdog Timeout Flag, reset to 0 |
| WDCON.0 | WDMOD | Watchdog Mode, reset to 1 (watchdog mode) |

## Enhanced UART

The UART operates in all of the usual modes that are described in the first section of this book for the 80C51. In addition the UART can perform framing error detect by looking for missing stop bits, and automatic address recognition. The 87C575 UART also fully supports multiprocessor communication as does the standard 80C51 UART.

When used for framing error detect the UART looks for missing stop bits in the communication. A missing bit will set the FE bit in the SCON register. The FE bit shares the SCON.7 bit with SM0 and the function of SCON.7 is determined by PCON.6 (SMOD0) (see Figure 19). If SMOD0 is set then SCON.7 functions as FE. SCON.7 functions as SM0 when SMOD0 is cleared. When used as FE SCON.7 can only be cleared by software. Refer to Figure 18.

**Automatic Address Recognition**

Automatic Address Recognition is a feature which allows the UART to recognize certain addresses in the serial bit stream by using hardware to make the comparisons. This feature saves a great deal of software overhead by eliminating the need for the software to examine every serial address which passes by the serial port. This feature is enabled by setting the SM2 bit in SCON. In the 9 bit UART modes, mode 2 and mode 3, the Receive Interrupt flag (RI) will be automatically set when the received byte contains either the "Given" address or the "Broadcast" address. The 9 bit mode requires that the 9th information bit is a 1 to indicate that the received information is an address and not data. Automatic address recognition is shown in Figure 20.

The 8 bit mode is called Mode 1. In this mode the RI flag will be set if SM2 is enabled and the information received has a valid stop bit following the 8 address bits and the information is either a Given or Broadcast address.

Mode 0 is the Shift Register mode and SM2 is ignored.

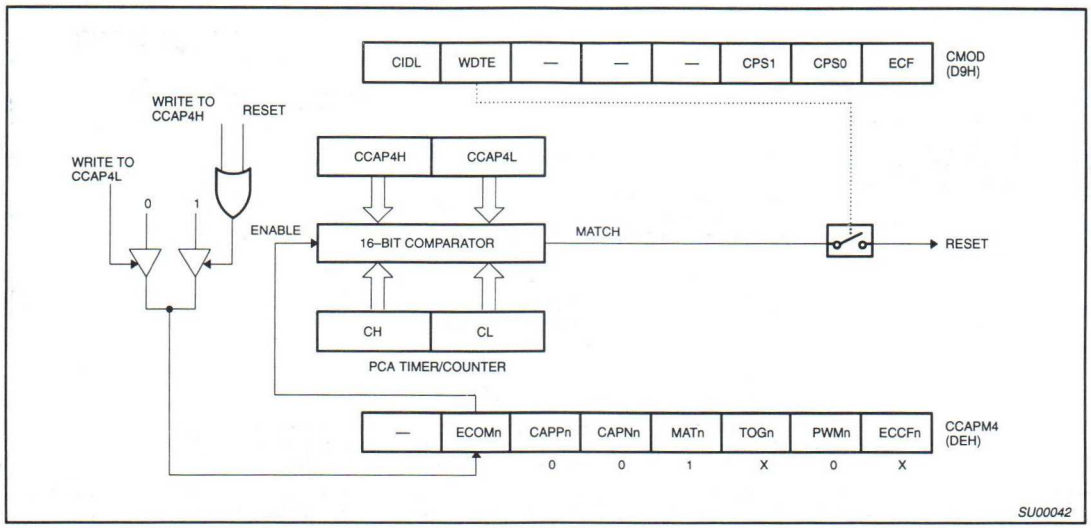



**Figure 13. PCA Watchdog Timer**

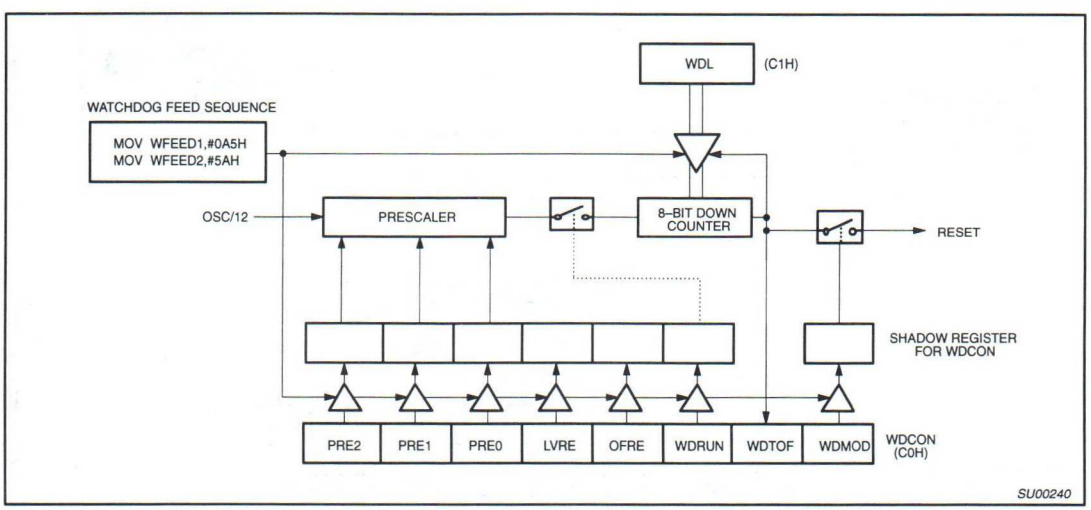



**Figure 14. Watchdog Timer in 87C575 and 80C575 / 83C575 (EA = 0)**

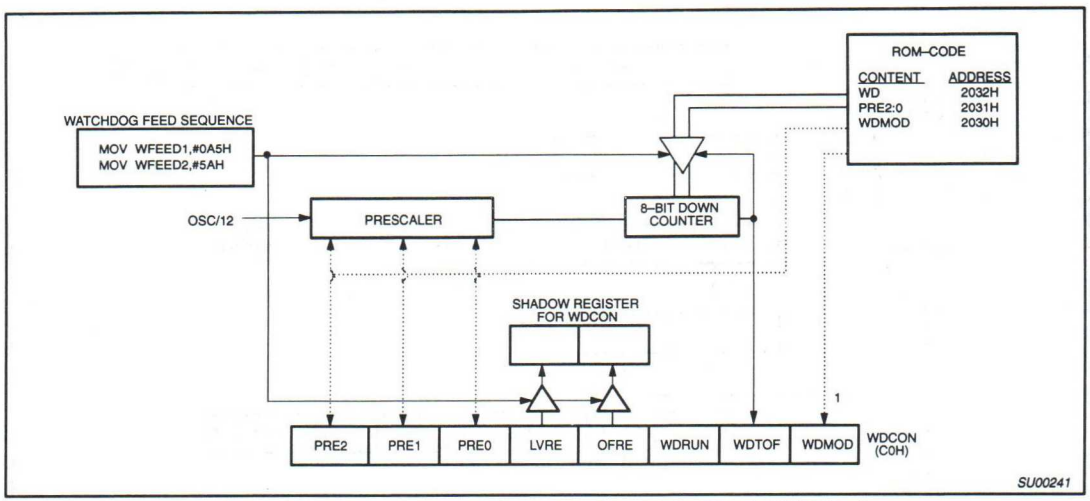



**Figure 15. Watchdog Timer of 83C575 in Watchdog Mode (EA = 1, WDMOD = 1)**

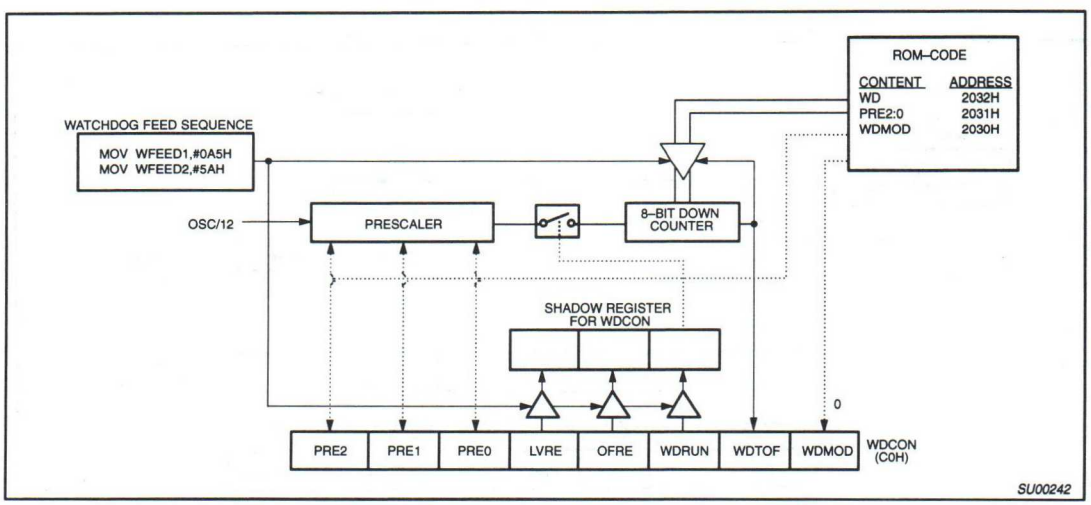



**Figure 16. Watchdog Timer of 83C575 in Timer Mode (EA = 1, WDMOD = 0)**

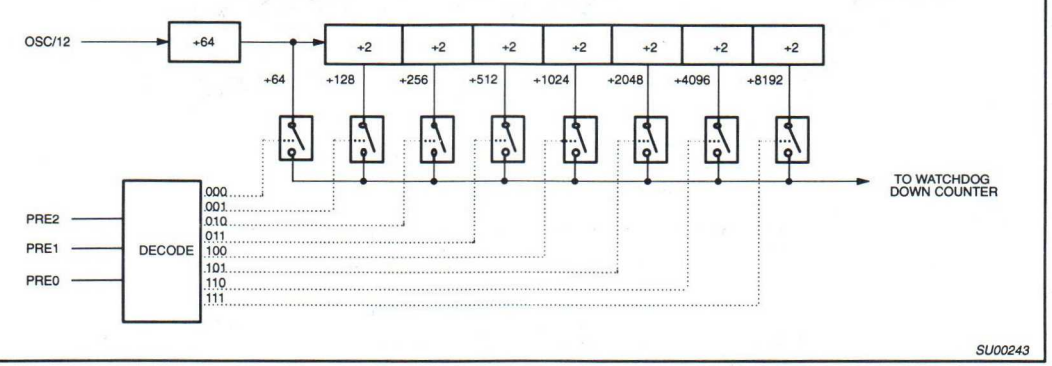



**Figure 17. Watchdog Prescaler**

**SCON** Address = 98H Reset Value = 0000 0000B

Bit Addressable

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | **SM0/FE** | **SM1** | **SM2** | **REN** | **TB8** | **RB8** | **TI** | **RI** |

(SMOD0 = 0/1)*

| Symbol | Function |
|---|---|
| **FE** | Framing Error bit. This bit is set by the receiver when an invalid stop bit is detected. The FE bit is not cleared by valid frames but should be cleared by software. The SMOD0 bit must be set to enable access to the FE bit. |
| **SM0** | Serial Port Mode Bit 0, (SMOD0 must = 0 to access bit SM0) |
| **SM1** | Serial Port Mode Bit 1 |

| **SM0** | **SM1** | **Mode** | **Description** | **Baud Rate**** |
|---|---|---|---|---|
| 0 | 0 | 0 | shift register | $f_{OSC}/12$ |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{OSC}/64$ or $f_{OSC}/32$ |
| 1 | 1 | 3 | 9-bit UART | variable |

| | |
|---|---|
| **SM2** | Enables the Automatic Address Recognition feature in Modes 2 or 3. If SM2 = 1 then RI will not be set unless the received 9th data bit (RB8) is 1, indicating an address, and the received byte is a Given or Broadcast Address. In Mode 1, if SM2 = 1 then RI will not be activated unless a valid stop bit was received, and the received byte is a Given or Broadcast Address. In Mode 0, SM2 should be 0. |
| **REN** | Enables serial reception. Set by software to enable reception. Clear by software to disable reception. |
| **TB8** | The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired. |
| **RB8** | In modes 2 and 3, the 9th data bit that was received. In Mode 1, if SM2 = 0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used. |
| **TI** | Transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software. |
| **RI** | Receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software. |

**NOTE:**
*SMOD0 is located at PCON6.
**$f_{OSC}$ = oscillator frequency

SU00043

**Figure 18. SCON: Serial Port Control Register**

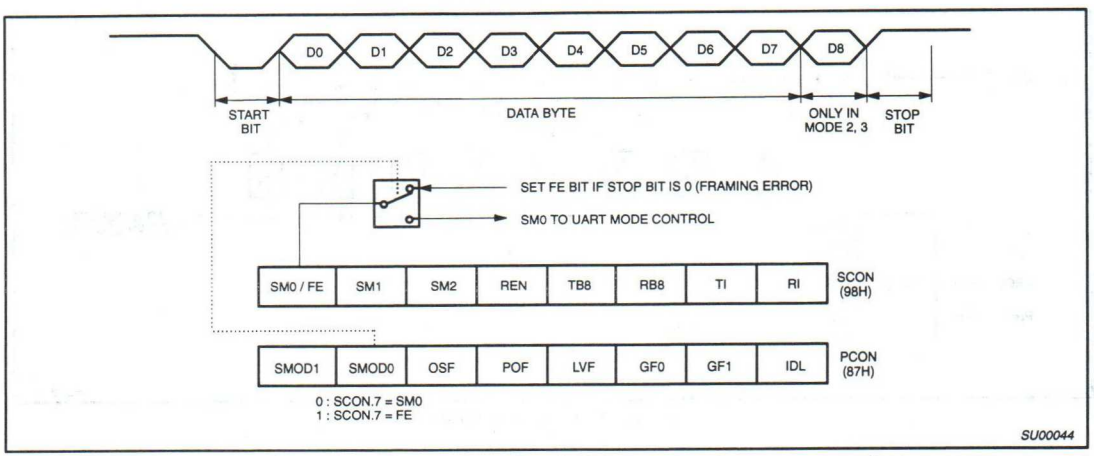



**Figure 19. UART Framing Error Detection**

Using the Automatic Address Recognition feature allows a master to selectively communicate with one or more slaves by invoking the Given slave address or addresses. All of the slaves may be contacted by using the Broadcast address. Two special Function Registers are used to define the slave's address, SADDR, and the address mask, SADEN. SADEN is used to define which bits in the SADDR are to b used and which bits are "don't care". The SADEN mask can be logically ANDed with the SADDR to create the "Given" address which the master will use for addressing each of the slaves. Use of the Given address allows multiple slaves to be recognized while excluding others. The following examples will help to show the versatility of this scheme:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1101 |
| | Given | = | 1100 00X0 |
| Slave 1 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1110 |
| | Given | = | 1100 000X |

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires a 0 in bit 0 and it ignores bit 1. Slave 1 requires a 0 in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010 since slave 1 requires a 0 in bit 1. A unique address for slave 1 would be 1100 0001 since a 1 in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 1100 0000.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1001 |
| | Given | = | 1100 0XX0 |
| Slave 1 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1010 |
| | Given | = | 1110 0X0X |
| Slave 2 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1100 |
| | Given | = | 1110 00XX |

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 and 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary t make bit 2 = 1 to exclude slave 2.

The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are treated as don't-cares. In most cases, interpreting the don't-cares as ones, the broadcast address will be FF hexadecimal.

Upon reset SADDR (SFR address 0A9H) and SADEN (SFR address 0B9H) are loaded with 0s. This produces a given address of all "don't cares" as well as a Broadcast address of all "don't cares". this effectively disables the Automatic Addressing mode and allows the microcontroller to use standard 80C51 type UART drivers which do not make use of this feature.

## Analog Comparators

Four analog comparators are provided on chip. Three comparators have a common negative reference CMPR- and independent positive inputs CMP1+, CMP2+, CMP3+ on port 3. The fourth comparator has independent positive and negative inputs CMP0+ and CMP0- on port 1. The CMP register contains an output and enable bit for each comparator. The CMP register is bit addressable and is located at SFR address E8H. Figure 21 shows the connection of the comparators.

Pullups at the comparator input pins will be disabled by hardware when the comparator is enabled. In addition, to make inputs high impedance, the corresponding port SFR bits must be set by software to disable the pulldowns.

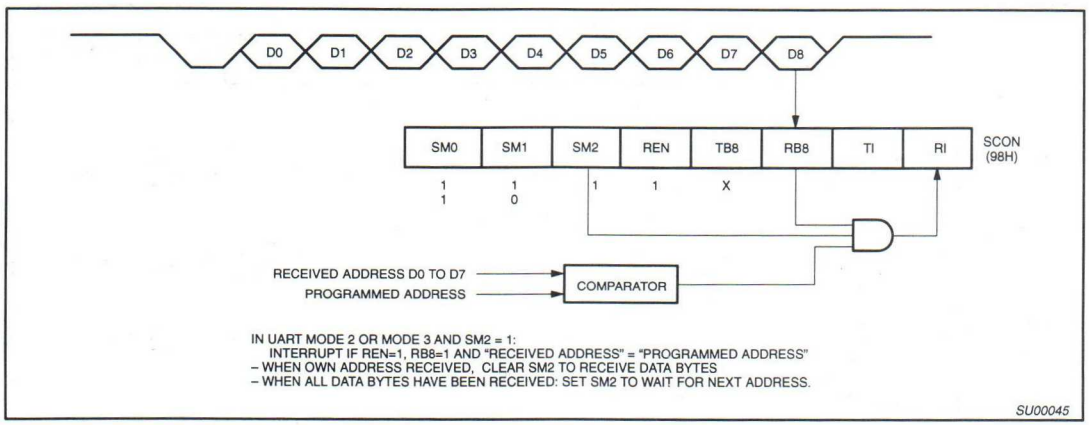



**Figure 20. UART Multiprocessor Communication, Automatic Address Recognition**

**CMP Register Bit Definitions**

CMP.7 enable comparator 3, disable pullups at P3.4, P3.7
CMP.6 enable comparator 2, disable pullups at P3.4, P3.6
CMP.5 enable comparator 1, disable pullups at P3.4, P3.5
CMP.4 enable comparator 0, disable pullups at P1.0, P1.1
CMP.3 comparator 3 output (read only)
CMP.2 comparator 2 output (read only)
CMP.1 comparator 1 output (read only)
CMP.0 comparator 0 output (read only)

All comparators are disabled automatically in power down mode, in idle mode unused comparators should be disabled by software to save power. A comparator can generate an interrupt that will terminate idle mode.

The CMPE register contains bits to enable each comparator to drive external output pins or internal PCA capture inputs. Pullups at the output pins are disabled by hardware when the external comparator output is enabled. The comparator output is wire-ORed with the corresponding port SFR bit, so the SFR bit must also be set by software to enable the output.

**CMPE Register Bit Definitions**

CMPE.7 enables comparator 3 to drive CEX3
CMPE.6 enables comparator 2 to drive CEX2
CMPE.5 enables comparator 1 to drive CEX1
CMPE.4 enables comparator 0 to drive CEX0
CMPE.3 enables comparator 3 output on P1.6 (open drain)
CMPE.2 enables comparator 2 output on P1.5 (open drain)
CMPE.1 enables comparator 1 output on P1.4 (open drain)
CMPE.0 enables comparator 0 output on P1.3 (open drain)

When 1s are written to CMPE bits 7-4, the comparator outputs will drive the corresponding capture input. (This function is not available in the idle or power-down mode.) When 1s are written to CMPE bits 3-0 the comparator output will also drive the corresponding port 1 pin. (This function **is** available in idle mode.) If the comparator's enabled to drive the capture input but not the port pin, then the port pin can be used for general purpose I/O. When a comparator output is enabled, pullups at the output pin are disabled and the output becomes open drain. The comparator output can be used to trigger a capture input in idle mode by programming the CMPE register to drive the pin from the comparator output to have the pin supply the capture trigger.

There are two special function registers associated with the comparators. They are CMP which contains the comparator enables and a bit that can be read by software to determine the state of each comparator's output, and CMPE which controls whether the output from each comparator drives the associated output pin or a capture input associated with one of the PCA modules.

The CMP registers bits 0–3 can be read by software to determine the state of the output of each comparator. To do this the associated comparator must be enabled but the output in port 1 can be disabled. This allows easy polling of the comparator output value without the need to use up a port pin.

The CMPE register allows the comparator to drive the associated PCA module capture input, so that on compare a capture can be generated in the PCA. Bits 0–3 of this register enable the comparator output to drive the associated port 1 output circuitry. Used as a comparator output this circuitry is open drain. To enable the comparator output to drive to port 1, the corresponding port bit must also be set to disable the pulldown. If the comparator is not enabled to drive the port 1 circuitry, the associated port 1 pin can be used for other I/O. This includes when a comparator is enabled to drive the capture input to a PCA module.

## Reduced EMI Mode

There are two bits in the AUXR register that can be set to reduce the internal clock drive and disable the ALE output. AO (AUXR.0) when set turns off the ALE output. LO (AUXR.1) when set reduces the drive of the internal clock circuitry. Both bits are cleared on Reset. With LO set the 87C575 will still operate at 12MHz, but will have reduced EMI in the range above 100MHz.

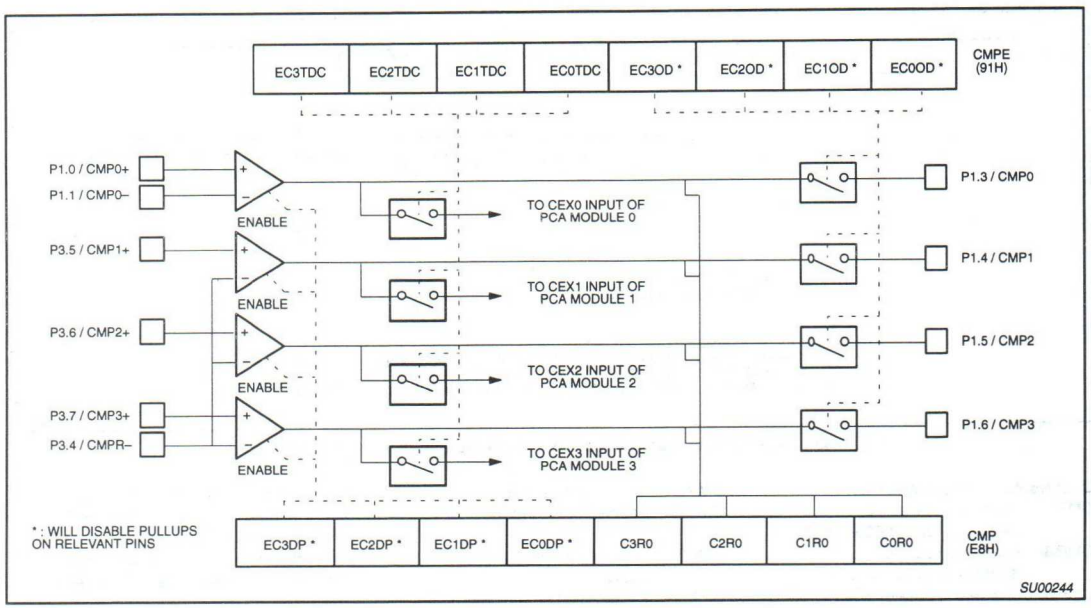



**Figure 21. Analog Comparators**

## 8XC575 Reduced EMI Mode

**AUXR (0X8E)**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | LO | AO |

AO: Turns off ALE output.

LO: Reduces drive of internal clock circuitry. 8XC575 spec'd to 12MHz when LO set.

## INTERNAL RESET

Internal resets generated by the power on, low voltage, and oscillator fail detect circuits are self timed to guarantee proper initialization of the 8XC575. Reset will be held approximately 24 oscillator periods after normal conditions are detected by all enabled detect circuits. Internal resets do not drive $\overline{RST}$ but will cause missing pulses on ALE.

**Interrupt Enable (IE) Register**

| | | |
|---|---|---|
| EA | IE.7 | enable all interrupts |
| EC | IE.6 | enable PCA interrupt |
| ET2 | IE.5 | enable Timer 2 interrupt |
| ES | IE.4 | enable Serial I/O interrupt |
| ET1 | IE.3 | enable Timer 1 interrupt |
| EX1 | IE.2 | enable External interrupt 1 |
| ET0 | IE.1 | enable Timer 0 interrupt |
| EX0 | IE.0 | enable External interrupt 0 |

**Interrupt Priority (IP) Register**

| | | |
|---|---|---|
| | IP.7 | reserved |
| PPC | IP.6 | PCA interrupt priority |
| PT2 | IP.5 | Timer 2 interrupt priority |
| PS | IP.4 | Serial I/O interrupt priority |
| PT1 | IP.3 | Timer 1 interrupt priority |
| PX1 | IP.2 | External interrupt 1 priority |
| PT0 | IP.1 | Timer 0 interrupt priority |
| PX0 | IP.0 | External interrupt 0 priority |

| Priority | Source | Flag | Vector |
|---|---|---|---|
| 1 | INT0 | IE0 | 03H highest priority |
| 2 | Timer 0 | TF0 | 0BH |
| 3 | INT1 | IE1 | 13H |
| 4 | Timer 1 | TF1 | 1BH |
| 5 | PCA | CF,CCFn | 33H |
| 6 | Serial I/O | RI,TI | 23H |
| 7 | Timer 2 | TF2/EXF2 | 2BH lowest priority |

**Power Control (PCON) Register**

| | | |
|---|---|---|
| SMOD1 | PCON.7 | double baud rate bit |
| SMOD0 | PCON.6 | SCON.7 access control |
| OSF | PCON.5 | oscillator fail flag |
| POF | PCON.4 | power off flag |
| LVF | PCON.3 | low voltage flag |
| GF0 | PCON.2 | general purpose flag |
| PD | PCON.1 | power down mode bit |
| IDL | PCON.0 | idle mode bit |

**Auxiliary Register Bit Definitions (AUXR = 8EH)**

| | | |
|---|---|---|
| AO | AUXR.0 | ALE Off, when set turns off ALE |
| LO | AUXR.1 | Low Speed, reduces internal clock drive |

**Port 2 Pullup Disable Register (P2OD = 0A1H)**

Port 2 pullups can be disabled by writing ones to P2OD. Each bit in P2OD controls the corresponding bit in P2. P2OD resets to all zeros enabling Port 2 pullups. Writing one to a P2OD bit disables pullups at the corresponding port 2 bit making the output open drain.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the Logic Symbol, page 3-722.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. The control bits for the reduced power modes are in the special function register PCON. Power-down mode can be terminated with either a hardware reset or external interrupt. With an external interrupt $\overline{INT0}$ or $\overline{INT1}$ must be enabled and configured as level sensitive. Holding the pin low restarts to oscillator and bringing the pin back high completes the exit.

If the watchdog is enabled (WDRUN = 1), then power-down mode is disabled.

## DESIGN CONSIDERATIONS

At power-on, the voltage on $V_{CC}$ must come up with $\overline{RST}$ low for a proper start-up.

Table 2 shows the state of I/O ports during low current operating modes.

**Table 2. External Pin Status During Idle and Power-Down Modes**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

## ROM CODE SUBMISSION

When submitting ROM code for the 83C575, the following must be specified:

1. 8k byte user ROM data
2. 32 byte ROM encryption key
3. ROM security bits
4. The watchdog timer parameters.

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 1FFFH | DATA | 7:0 | User ROM Data |
| 2000H to 201FH | KEY | 7:0 | ROM Encryption Key<br>FFH = no encryption |
| 2020H | Reserved<br>Security Bit 2<br>Security Bit 1 | 2<br>1<br>0 | Must = 1<br>0 = enable, 1 = disable<br>0 = enable, 1 = disable |
| 2030H | Reserved | 7:0 | Must = FFH |
| 2031H | Reserved | 7:0 | Must = FFH |
| 2032H | WDL [1] | 7:0 | Watchdog reload value<br>(see specification) |
| 2033H | WDCON [1] | 7:5 | PRE2:0 |
| 2033H | WDCON [1] | 4 | LVRE |
| 2033H | WDCON [1] | 3 | OFRE |
| 2033H | WDCON [1] | 2 | WDRUN=0, not ROM coded |
| 2033H | WDCON [1] | 1 | WDTOF=0, not ROM coded |
| 2033H | WDCON [1] | 0 | WDMOD |

**NOTES:**

1. See Watchdog Timer Specification for definition of WDL and WDCON bits.

**Security Bit 1:** When programmed, this bit has two effects on masked ROM parts:

1. External MOVC is disabled, and
2. EA# is latched on Reset.

**Security Bit 2:** When programmed, this bit inhibits Verify User ROM.

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | –55 to +125 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Maximum $I_{OL}$ per I/O pin | 15 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.5 | W |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C and –40°C to +125°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $V_{IL}$ | Input low voltage (Ports 0, 2, 3, except 3.2, 3.3) | | –0.5 | | $0.5V_{CC}$–0.6 | V |
| $V_{IL1}$ | Input low voltage (Ports 1, 3.2, 3.3) | | –0.5 | | $0.65V_{CC}$–0.5 | V |
| $V_{IL2}$ | Input low voltage ($\overline{EA}$) | | 0 | | $0.2V_{CC}$–0.45 | V |
| $V_{IL3}$ | Input low voltage (XTAL1, RST) | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IH}$ | Input high voltage (Ports 0, 2, 3, except 3.2, 3.3) | | $0.5V_{CC}$+0.8 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage (Ports 1, 3.2, 3.3) | | $0.8V_{CC}$+0.3 | | $V_{CC}$+0.5 | V |
| $V_{IH2}$ | Input high voltage ($\overline{EA}$) | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH3}$ | Input high voltage (XTAL1, RST) | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| HYS | Hysteresis (Ports 0, 2, 3, except 3.2, 3.3) | | 200 | | | mV |
| HYS1 | Hysteresis (Ports 1, 3.2, 3.3) | | 50 | | | mV |
| $V_{OL}$ | Output voltage low (Ports 1, 2, 3, except 3.1) | $I_{OL}$ = 1.6mA | | | 0.45 | V |
| $V_{OL1}$ | Output voltage low (Ports 0, ALE, $\overline{PSEN}$) | $I_{OL}$ = 3.2mA | | | 0.45 | V |
| $V_{OL2}$ | Output voltage low<br>P3.1 with bit cleared<br>P3.1 with bit set | <br>$I_{OL}$ = 10.0mA<br>$I_{OL}$ = 1.6mA | | | <br>0.50<br>0.45 | <br>V<br>V |
| $V_{OH}$ | Output voltage high (Ports 1, 2, 3, except P3.1) | $I_{OH}$ = –30µA | $V_{CC}$–0.7 | | | V |
| $V_{OH1}$ | Output voltage high (Port 0 in external bus mode, ALE, $\overline{PSEN}$) | $I_{OH}$ = –3.2mA | $V_{CC}$–0.7 | | | V |
| $V_{OH2}$ | Output voltage high<br>P3.1 with bit cleared<br>P3.1 with bit set | <br>$I_{OH}$ = –10.0mA<br>$I_{OH}$ = –1.6mA | <br>$V_{CC}$–1.5<br>$V_{CC}$–1.5 | | | <br>V<br>V |
| $V_{IO}$ | Offset voltage comparator inputs | | –35 | | +35 | mV |
| $V_{CR}$ | Common mode range comparator inputs | | 0 | | $V_{CC}$ | V |
| $I_{IL}$ | Logical 0 input current (Ports 1, 2, 3, except 3.1) | $V_{IN}$ = 0.45V | | | –75 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current (Ports 2, 3, except 3.1, 3.2, 3.3) [4] | See Note 4 | | | –600 | µA |
| $I_{TL1}$ | Logical 1-to-0 transition current (Ports 1, 3.2, 3.3) | See Note 4 | | | –450 | µA |
| $I_{L1}$ | Input leakage current (Port 0, Port2 in open drain mode)[9] | $0.45 < V_{IN} < V_{CC}$ | 2 | | 40 | µA |
| $I_{L2}$ | Input leakage current ($\overline{EA}$, P3.1) | $0.45 < V_{IN} < V_{CC}$ | –10 | | +10 | µA |
| $I_{LC}$ | Input leakage current comparator inputs | $0 < V_{IN} < V_{CC}$ | –1.0 | | +1.0 | µA |
| $I_{CC}$ | Power supply current:[7]<br>Active mode @ 16MHz[5]<br>Idle mode @ 16MHz<br>Power-down mode | See note 6 | | <br>20<br>8<br>5 | <br>30<br>12<br>75 | <br>mA<br>mA<br>µA |
| $R_{RST}$ | Internal reset pull-up resistor | $V_{IN}$ = 0V | 50 | | 200 | kΩ |
| $V_{LOW}$ | Low $V_{CC}$ detect voltage | | 4.0 | | 4.45 | V |
| $C_{IO}$ | Pin capacitance[10] | f = 1MHz | | | 10 | pF |

**NOTES: (SEE NEXT PAGE)**

**NOTES TO THE DC ELECTRICAL CHARACTERISTICS TABLE:**

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is between $V_{IH}$ and $V_{IL}$.
5. $I_{CC}$MAX at other frequencies can be determined from Figure 29.
6. See Figures 30 through 33 for $I_{CC}$ test conditions.
7. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
8. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 10mA
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 71mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
9. Specification applies to Port 2 when P2OD bit is set.
10. 15pF MAX for the $\overline{EA}/V_{PP}$ and P0.0 pins.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C and −40°C to +125°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2]

| SYMBOL | FIGURE | PARAMETER | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $1/t_{CLCL}$ | 22 | Oscillator frequency: **Speed Versions** 8XC575 E | 6 | 16 | MHz |
| OSCF | | Oscillator fail detect frequency | 0.6 | 5.5 | MHz |
| TR | | Comparator response time | | 10 | µs |
| $t_{LHLL}$ | 22 | ALE pulse width | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 22 | Address valid to ALE low | $t_{CLCL}-25$ | | ns |
| $t_{LLAX}$ | 22 | Address hold after ALE low | $t_{CLCL}-25$ | | ns |
| $t_{LLIV}$ | 22 | ALE low to valid instruction in | | $4t_{CLCL}-75$ | ns |
| $t_{LLPL}$ | 22 | ALE low to $\overline{PSEN}$ low | $t_{CLCL}-25$ | | ns |
| $t_{PLPH}$ | 22 | $\overline{PSEN}$ pulse width | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 22 | $\overline{PSEN}$ low to valid instruction in | | $3t_{CLCL}-70$ | ns |
| $t_{PXIX}$ | 22 | Input instruction hold after $\overline{PSEN}$ | 0 | | ns |
| $t_{PXIZ}$ | 22 | Input instruction float after $\overline{PSEN}$ | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 22 | Address to valid instruction in | | $5t_{CLCL}-85$ | ns |
| $t_{PLAZ}$ | 22 | $\overline{PSEN}$ low to address float | | 10 | ns |
| **Data Memory** | | | | | |
| $t_{RLRH}$ | 23, 24 | $\overline{RD}$ pulse width | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 23, 24 | $\overline{WR}$ pulse width | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 23, 24 | $\overline{RD}$ low to valid data in | | $5t_{CLCL}-110$ | ns |
| $t_{RHDX}$ | 23, 24 | Data hold after $\overline{RD}$ | 0 | | ns |
| $t_{RHDZ}$ | 23, 24 | Data float after $\overline{RD}$ | | $2t_{CLCL}-28$ | ns |
| $t_{LLDV}$ | 23, 24 | ALE low to valid data in | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 23, 24 | Address to valid data in | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 23, 24 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 23, 24 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | $4t_{CLCL}-75$ | | ns |
| $t_{QVWX}$ | 23, 24 | Data valid to $\overline{WR}$ transition | $t_{CLCL}-30$ | | ns |
| $t_{WHQX}$ | 23, 24 | Data hold after $\overline{WR}$ | $t_{CLCL}-25$ | | ns |
| $t_{RLAZ}$ | 23, 24 | $\overline{RD}$ low to address float | | 0 | ns |
| $t_{WHLH}$ | 23, 24 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | $t_{CLCL}-25$ | $t_{CLCL}+25$ | ns |
| **External Clock** | | | | | |
| $t_{CHCX}$ | 26 | High time | 12 | | ns |
| $t_{CLCX}$ | 26 | Low time | 12 | | ns |
| $t_{CLCH}$ | 26 | Rise time | | 20 | ns |
| $t_{CHCL}$ | 26 | Fall time | | 20 | ns |
| **Shift Register** | | | | | |
| $t_{XLXL}$ | 25 | Serial port clock cycle time | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 25 | Output data setup to clock rising edge | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 25 | Output data hold after clock rising edge | $2t_{CLCL}-60$ | | ns |
| $t_{XHDX}$ | 25 | Input data hold after clock rising edge | 0 | | ns |
| $t_{XHDV}$ | 25 | Clock rising edge to input data valid | | $10t_{CLCL}-133$ | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 80C32/52 to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{\text{PSEN}}$
Q – Output data
R – $\overline{\text{RD}}$ signal
t – Time
V – Valid
W – $\overline{\text{WR}}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{\text{PSEN}}$ low.

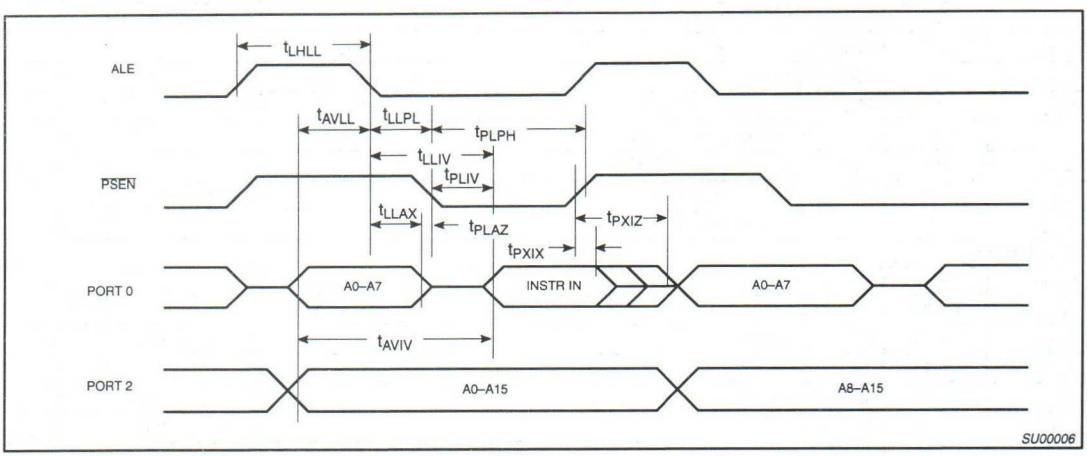



**Figure 22. External Program Memory Read Cycle**

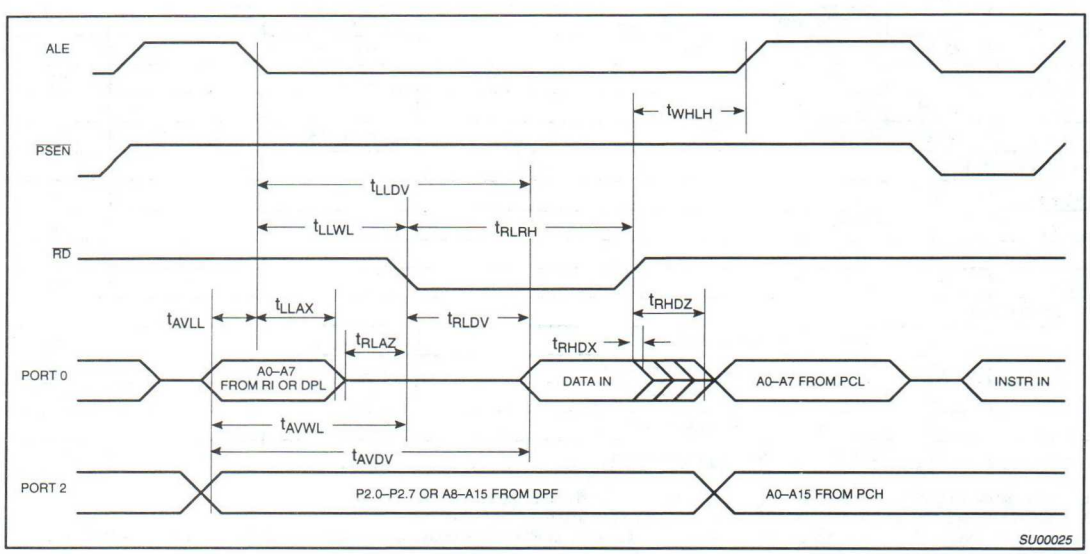



**Figure 23. External Data Memory Read Cycle**

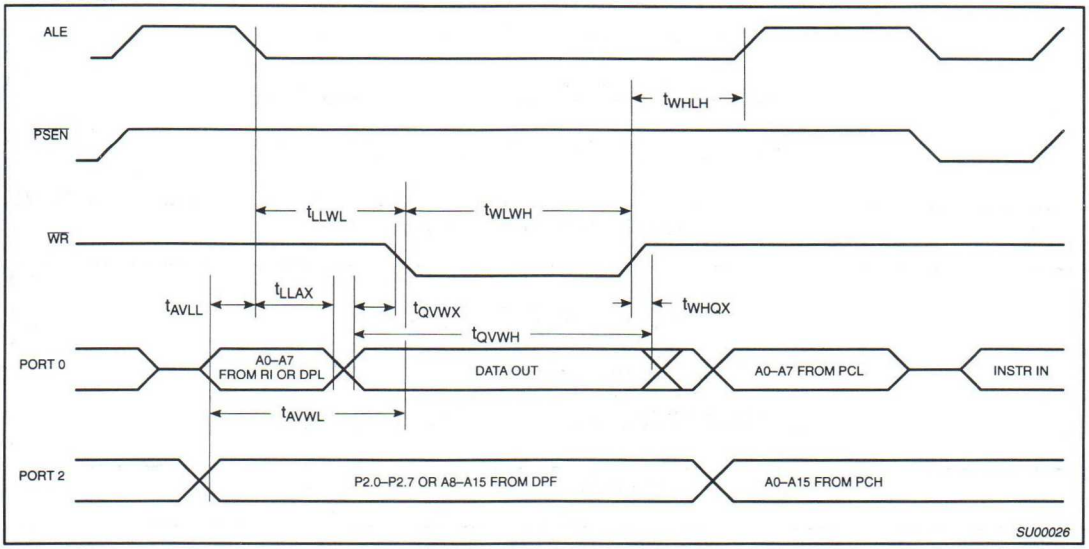



**Figure 24. External Data Memory Write Cycle**

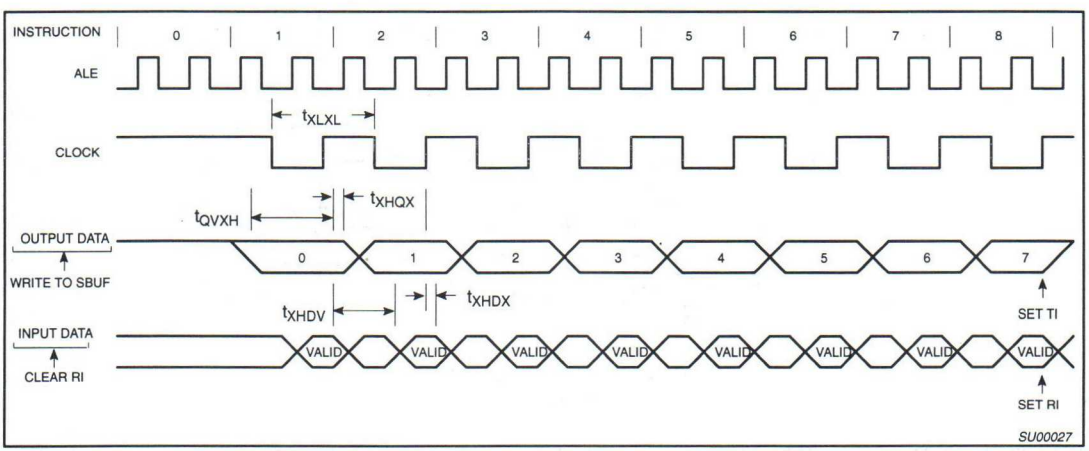



**Figure 25. Shift Register Mode Timing**

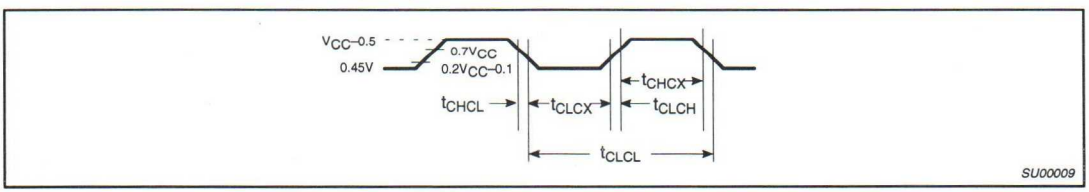
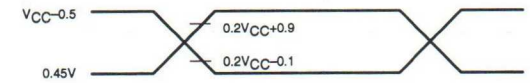


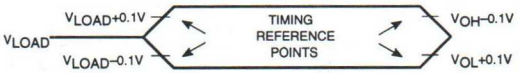

**Figure 26. External Clock Drive**




NOTE:
AC inputs during testing are driven at $V_{CC}$ –0.5 for a logic '1' and 0.45V for a logic '0'.
Timing measurements are made at $V_{IH}$ min for a logic '1' and $V_{IL}$ max for a logic '0'.

SU00010

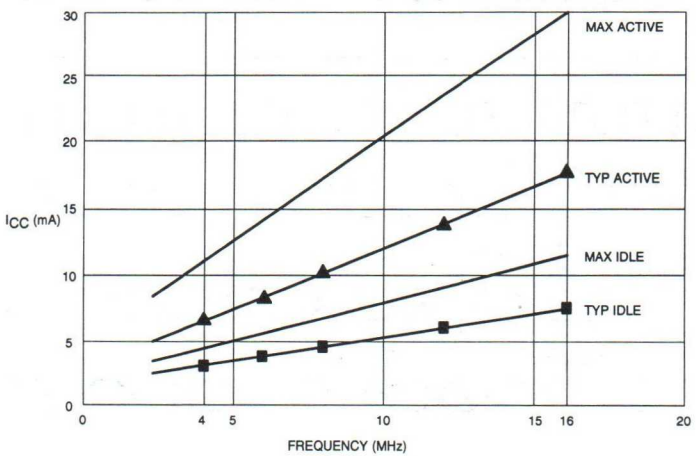
**Figure 27. AC Testing Input/Output**




NOTE:
For timing purposes, a port is no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded $V_{OH}/V_{OL}$ level occurs. $I_{OH}/I_{OL} \geq \pm 20$mA.

SU00011

**Figure 28. Float Waveform**




SU00245

**Figure 29. $I_{CC}$ vs. FREQ**
**Valid only within frequency specifications of the device under test**

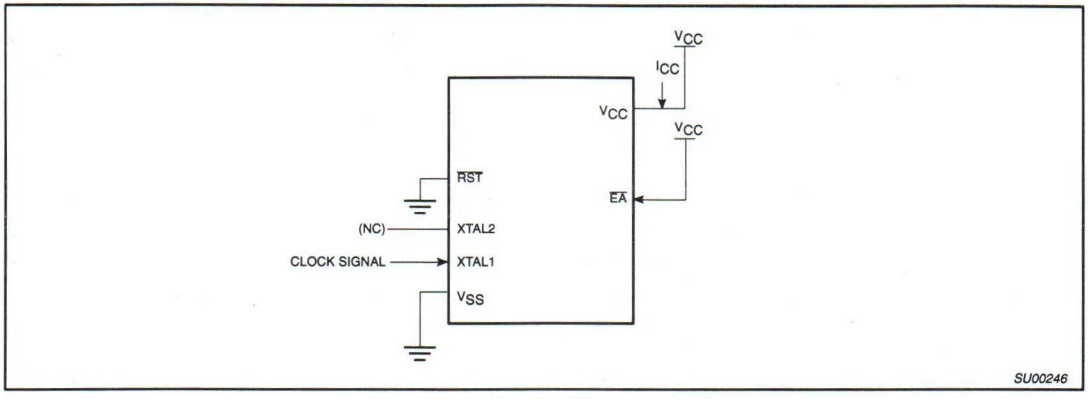



**Figure 30. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

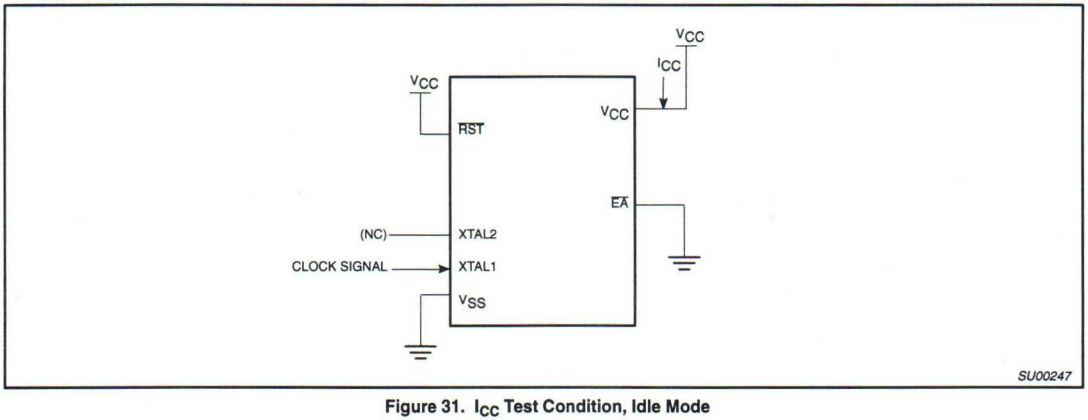



**Figure 31. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

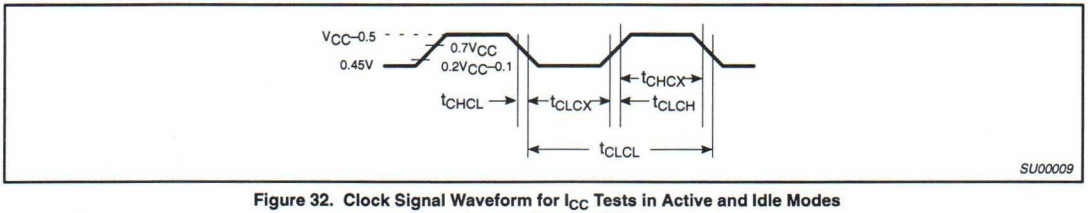



**Figure 32. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

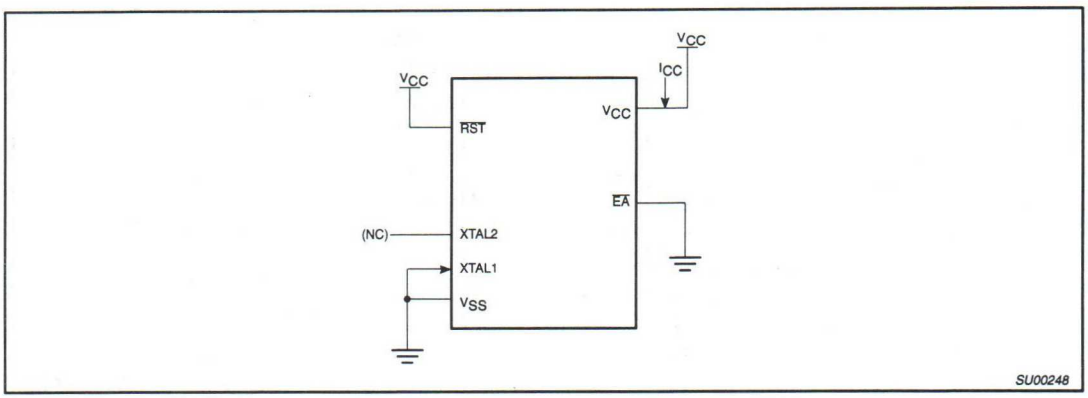



**Figure 33. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2V to 5.5V**

## EPROM CHARACTERISTICS

To put the 87C575 in the EPROM programming mode, PSEN must be held high during power up, then driven low with reset active. The 87C575 is programmed by using a modified Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C575 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C575 manufactured by Philips.

Table 3 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the security bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 34 and 35. Figure 36 shows the circuit configuration for normal program memory verification.

### Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 34. Note that the 87C575 is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1 and 2, as shown in Figure 34. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 3 specified in Table 3 are held at the 'Program Code Data' levels indicated in Table 3. The ALE/$\overline{PROG}$ is pulsed low 25 times as shown in Figure 35.

To program the encryption table, repeat the 25 pulse programming sequence for addresses 0 through 1FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the security bits, repeat the 25 pulse programming sequence using the 'Pgm Security Bit' levels. After one security bit is programmed, further programming of the code memory and encryption table is disabled. However, the other security bit can still be programmed.

Note that the $\overline{EA}$/$V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

#### Program Verification

If security bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 1 and 2 as shown in Figure 36. The other pins are held at the 'Verify Code Data' levels indicated in Table 3. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

#### Reading the Signature Bytes

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:

(030H) = 15H indicates manufactured by Philips
(B0H) = 97H indicates 87C575

### Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 3, and which satisfies the timing specifications, is suitable.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-s/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

**Table 3. EPROM Programming Modes**

| MODE | RST | $\overline{PSEN}$ | ALE/$\overline{PROG}$ | $\overline{EA}$/$V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 0 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 0 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm security bit 1 | 0 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm security bit 2 | 0 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V±10% during programming and verification.

* ALE/$\overline{PROG}$ receives 25 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

™Trademark phrase of Intel Corporation.

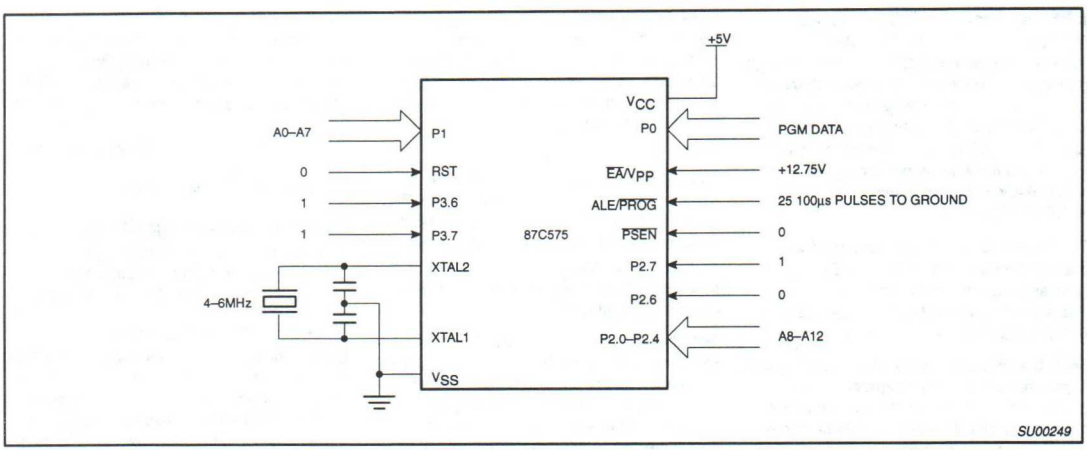



**Figure 34. Programming Configuration**

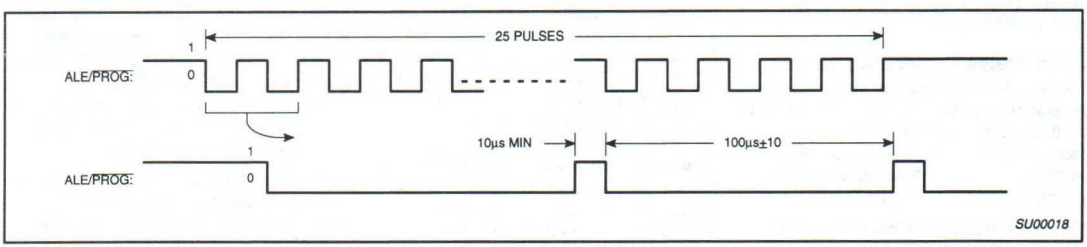



**Figure 35. $\overline{\text{PROG}}$ Waveform**

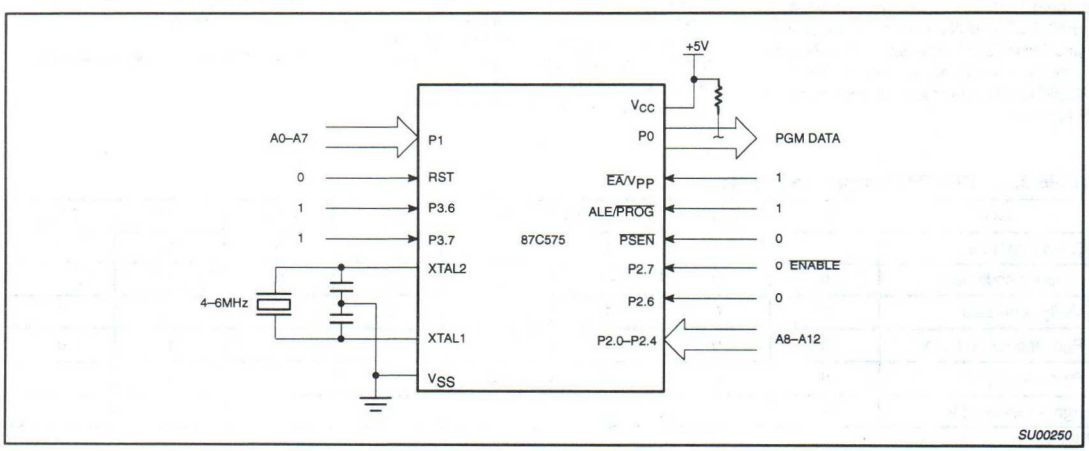
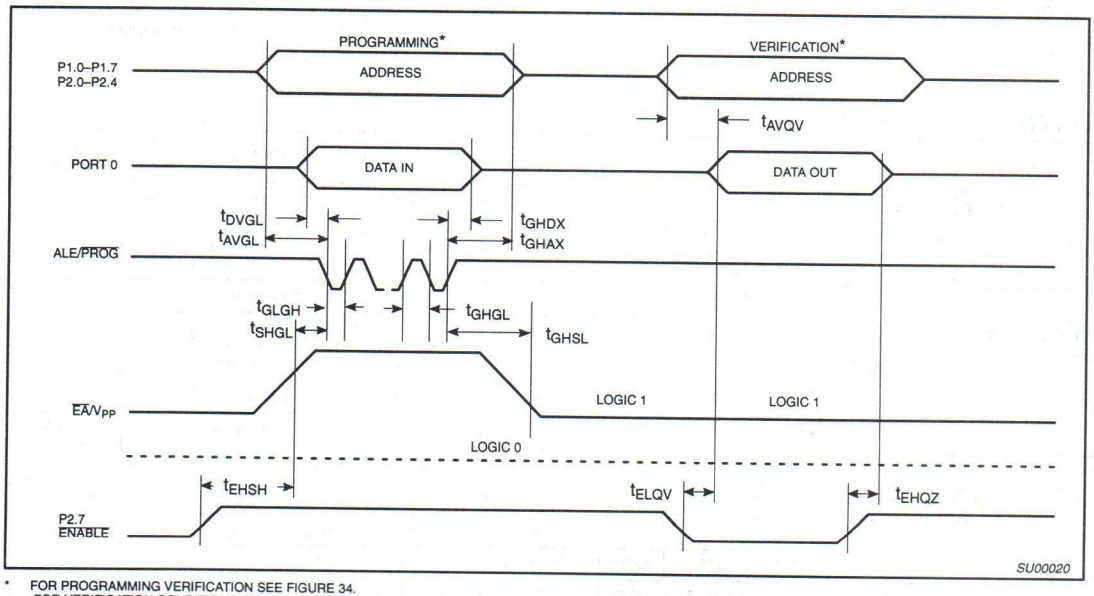



**Figure 36. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (See Figure 37)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




* FOR PROGRAMMING VERIFICATION SEE FIGURE 34.
FOR VERIFICATION CONDITIONS SEE FIGURE 36.

**Figure 37. EPROM Programming and Verification**

# CMOS single-chip 8-bit microcontrollers 83C576/87C576

## FEATURES

- 80C51 based architecture
  - 8k × 8 ROM (83C576)
  - 8k × 8 EPROM (87C576)
  - 256 × 8 RAM
  - 10-bit, 6 channel A/D
  - Three 16-bit counter/timers
  - 2 PWM outputs
  - Programmable Counter Array
  - Universal Peripheral Interface
  - Enhanced UART
  - Oscillator fail detect
  - Low active reset
  - 4 analog comparators
  - Watchdog timer
  - Low $V_{CC}$ detect
  - Power-on detect
- Memory addressing capability
  - 64k ROM and 64k RAM
- Power control modes:
  - Idle mode
  - Power-down mode
- CMOS and TTL compatible
- 6 to 16MHz
- Extended temperature ranges
- OTP package available
- EPROM/OTP versions can be programmed in circuit
- Software Reset
- 15 source, 2 level interrupt structure
- Lower EMI noise
- Programmable I/O pins
- Serial on-board programming
- Schmitt trigger inputs on Port 1

## DESCRIPTION

The Philips 83C576/87C576 is a high-performance microcontroller fabricated with Philips high-density CMOS technology. The Philips CMOS technology combines the high speed and density characteristics of HMOS with the low power attributes of CMOS. Philips epitaxial substrate minimizes latch-up sensitivity.

The 8XC576 contains an 8k × 8 ROM (83C576) EPROM (87C576), a 256 × 8 RAM, 32 I/O lines, three 16-bit counter/timers, a Programmable Counter Array (PCA), a 10-bit, 6 channel A/D, 2 PWM outputs, an 8-bit UPI interface, a fifteen-source, two-priority level nested interrupt structure, an enhanced UART, four analog comparators, power-fail detect and oscillator fail detect circuits, and on-chip oscillator and clock circuits.

In addition, the 8XC576 has a low active reset, and a software reset. There is also a fully configurable watchdog timer, and internal power on clear circuit. The part includes idle mode and power-down mode states for reduced power consumption.

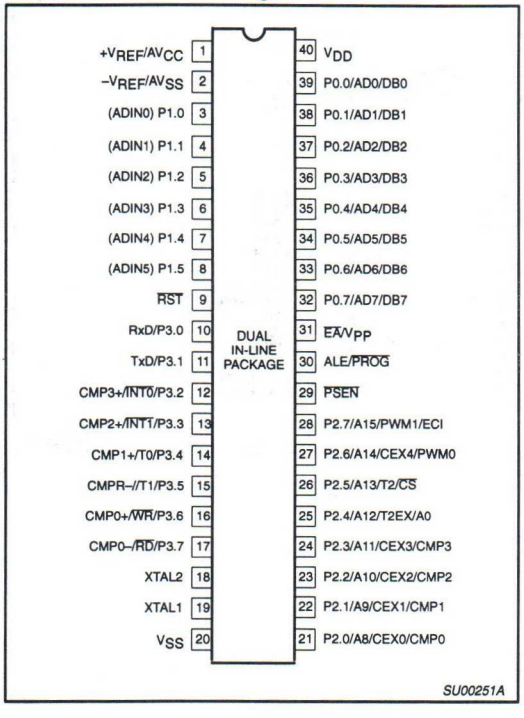
## ORDERING INFORMATION

| ROMless | ROM | EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQ (MHz) | DRAWING NUMBER |
|---|---|---|---|---|---|---|
| P80C576EBPN | P83C576EBPN | P87C576EBPN | OTP | 0 to +70, 40-Pin Plastic Dual In-line Package | 16 | SOT129-1 |
| P80C576EBAA | P83C576EBAA | P87C576EBAA | OTP | 0 to +70, 44-Pin Plastic Leaded Chip Carrier | 16 | SOT187-2 |
| | | P87C576EBFFA | UV | 0 to +70, 40-Pin Ceramic Dual In-line Package | 16 | 0590B |
| | | P87C576EBLKA | UV | 0 to +70, 44-Pin Ceramic Leaded Chip Carrier | 16 | 1472A |
| P80C576EBBB | P83C576EBBB | P87C576EBBB | OTP | 0 to +70, 44-Pin Plastic Quad Flat Pack | 16 | SOT307-2 |
| P80C576EFPN | P83C576EFPN | P87C576EBPN | OTP | –40 to +85, 40-Pin Plastic Dual In-line Package | 16 | SOT129-1 |
| P80C576EFAA | P83C576EFAA | P87C576EFAA | OTP | –40 to +85, 44-Pin Plastic Leaded Chip Carrier | 16 | SOT187-2 |
| | | P87C576EFFFA | UV | –40 to +85, 40-Pin Ceramic Dual In-line Package | 16 | 0590B |
| | | P87C576EFLKA | UV | –40 to +85, 44-Pin Ceramic Leaded Chip Carrier | 16 | 1472A |
| P80C576EFBB | P83C576EFBB | P87C576EFBB | OTP | –40 to +85, 44-Pin Plastic Quad Flat Pack | 16 | SOT307-2 |
| | P83C576EHPN | P87C576EHPN | OTP | –40 to +125, 40-Pin Plastic Dual In-line Package | 16 | SOT129-1 |
| | P83C576EHAA | P87C576EHAA | OTP | –40 to +125, 44-Pin Plastic Leaded Chip Carrier | 16 | SOT187-2 |
| | P83C576EHBB | P87C576EHBB | OTP | –40 to +125, 44-Pin Plastic Quad Flat Pack | 16 | SOT307-2 |

**NOTE:**
1. OTP - One Time Programmable EPROM. UV - Erasable EPROM

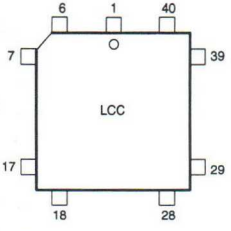
## PIN CONFIGURATIONS

### 40-pin Dual In-line Package




SU00251A

### 44-pin Ceramic and Plastic Leaded Chip Carrier

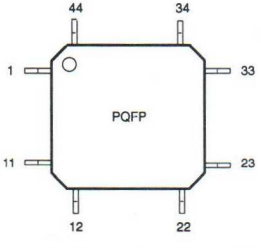



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | NC* | 16 | T0/P3.4/CMP1+ | 31 | P2.7/A15/PWM1/ECI |
| 2 | +$V_{REF}$/$AV_{CC}$ | 17 | T1/P3.5/CMPR– | 32 | PSEN |
| 3 | –$V_{REF}$/$AV_{SS}$ | 18 | WR/P3.6/CMP0+ | 33 | ALE/PROG |
| 4 | ADIN0/P1.0 | 19 | RD/P3.7/CMP0– | 34 | NC* |
| 5 | ADIN1/P1.1 | 20 | XTAL2 | 35 | EA/$V_{PP}$ |
| 6 | ADIN2/P1.2 | 21 | XTAL1 | 36 | P0.7/AD7/DB7 |
| 7 | ADIN3/P1.3 | 22 | $V_{SS}$ | 37 | P0.6/AD6/DB6 |
| 8 | ADIN4/P1.4 | 23 | NC* | 38 | P0.5/AD5/DB5 |
| 9 | ADIN5/P1.5 | 24 | P2.0/A8/CEX0/CMP0 | 39 | P0.4/AD4/DB4 |
| 10 | RST | 25 | P2.1/A9/CEX1/CMP1 | 40 | P0.3/AD3/DB3 |
| 11 | RxD/P3.0 | 26 | P2.2/A10/CEX2/CMP2 | 41 | P0.2/AD2/DB2 |
| 12 | NC* | 27 | P2.3/A11/CEX3/CMP3 | 42 | P0.1/AD1/DB1 |
| 13 | TxD/P3.1 | 28 | P2.4/A12/T2EX/A0 | 43 | P0.0/AD0/DB0 |
| 14 | INT0/P3.2/CMP3+ | 29 | P2.5/A13/T2/CS | 44 | $V_{CC}$ |
| 15 | INT1/P3.3/CMP2+ | 30 | P2.6/A14/CEX4/PWM0 | | |

* DO NOT CONNECT

SU00252A

### 44-pin Plastic Quad Flat Pack

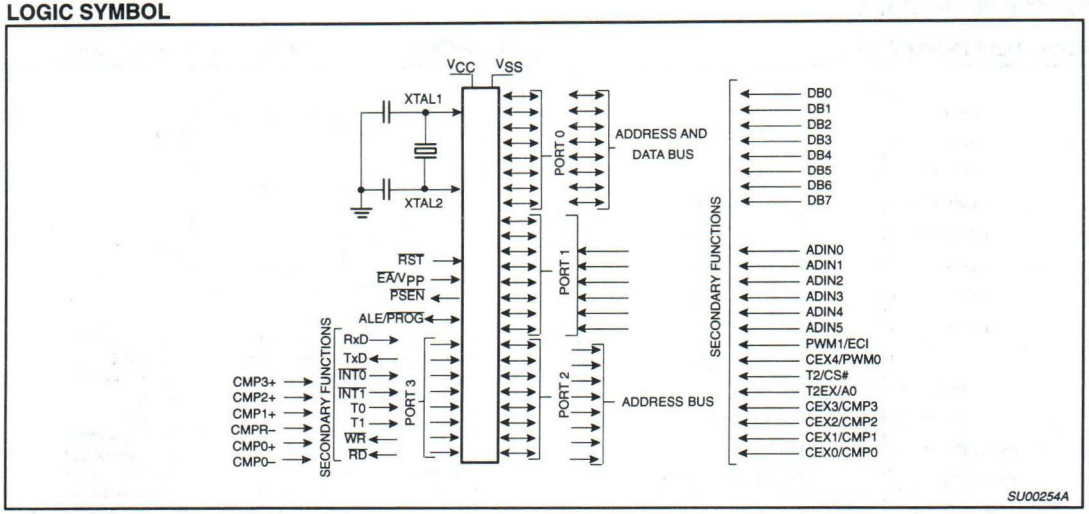



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | ADIN3/P1.3 | 16 | $V_{SS}$ | 31 | P0.6/AD6/DB6 |
| 2 | ADIN4/P1.4 | 17 | NC* | 32 | P0.5/AD5/DB5 |
| 3 | ADIN5/P1.5 | 18 | P2.0/A8/CEX0/CMP0 | 33 | P0.4/AD4/DB4 |
| 4 | RST | 19 | P2.1/A9/CEX1/CMP1 | 34 | P0.3/AD3/DB3 |
| 5 | RxD/P3.0 | 20 | P2.2/A10/CEX2/CMP2 | 35 | P0.2/AD2/DB2 |
| 6 | NC* | 21 | P2.3/A11/CEX3/CMP3 | 36 | P0.1/AD1/DB1 |
| 7 | TXD/P3.1 | 22 | P2.4/A12/T2EX/A0 | 37 | P0.0/AD0/DB0 |
| 8 | INT0/P3.2/CMP3+ | 23 | P2.5/A13/T2/CS | 38 | $V_{CC}$ |
| 9 | INT1/P3.3/CMP2+ | 24 | P2.6/A14/CEX4/PWM0 | 39 | NC* |
| 10 | T0/P3.4/CMP1+ | 25 | P2.7/A15/PWM1/ECI | 40 | +$V_{REF}$/$AV_{CC}$ |
| 11 | T1/P3.5/CMPR– | 26 | PSEN | 41 | –$V_{REF}$/$AV_{SS}$ |
| 12 | WR/P3.6/CMP0+ | 27 | ALE/PROG | 42 | ADIN0/P1.0 |
| 13 | RD/P3.7CMP0– | 28 | NC* | 43 | ADIN1/P1.1 |
| 14 | XTAL2 | 29 | EA/$V_{PP}$ | 44 | ADIN2/P1.2 |
| 15 | XTAL1 | 30 | P0.7/AD7/DB7 | | |

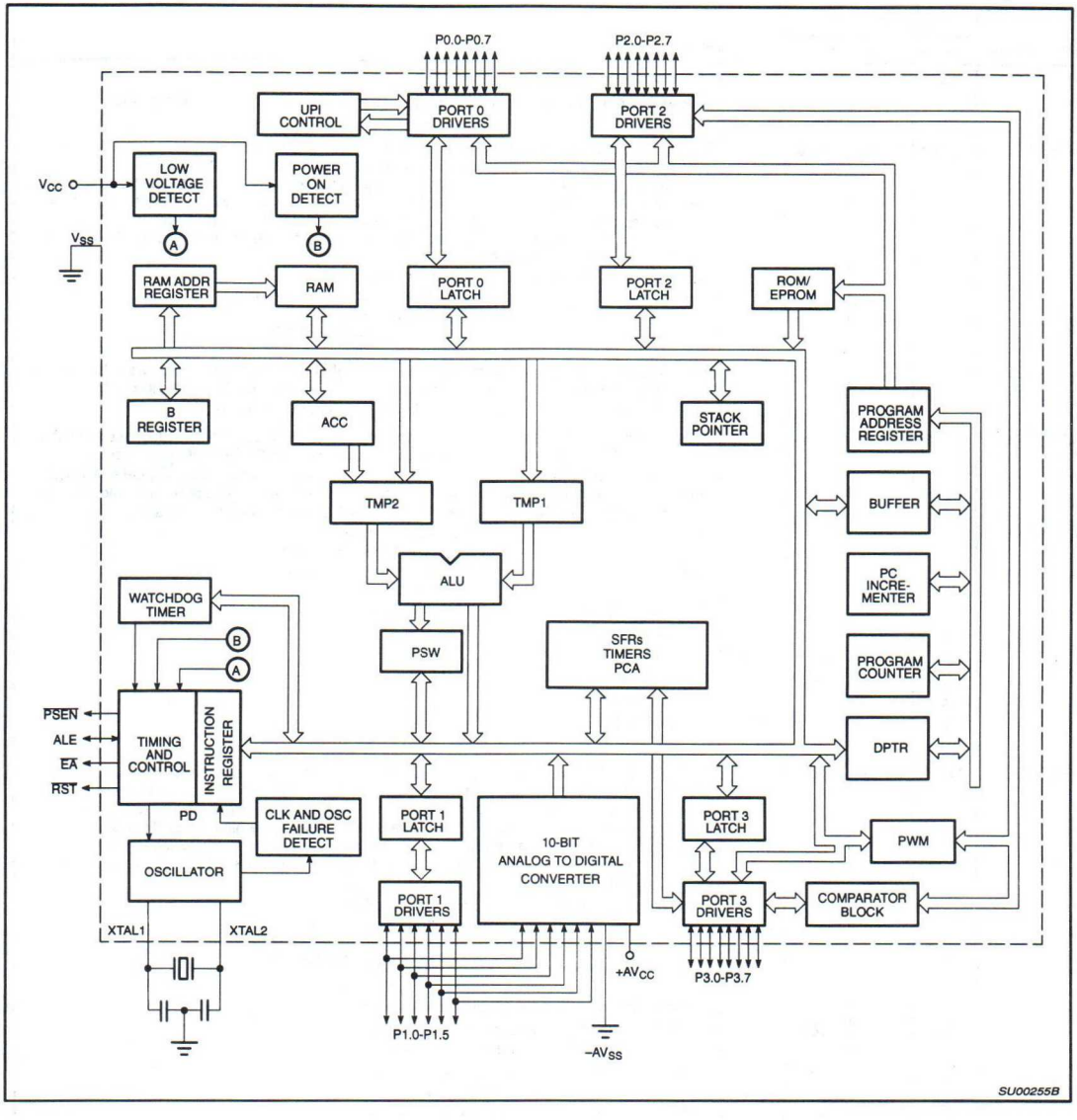
* DO NOT CONNECT

SU00253B

## LOGIC SYMBOL



## BLOCK DIAGRAM



## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER DIP | PIN NUMBER LCC | PIN NUMBER QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0-0.7 | 39-32 | 43-36 | 37-30 | I/O | **Port 0:** Port 0 is a bidirectional I/O port. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory (see Note 5). In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also receives code bytes during parallel EPROM programming and outputs code bytes during verification. External pull-ups are required during program verification. During reset, the port register is loaded with 1's. Port 0 has 4 output modes selected on a per bit basis by writing to the P0M1 and P0M2 Special Function Registers as follows:<br>**P0M1.x** **P0M2.x** **Mode Description**<br>0 0 Open drain (default). See Note 1.<br>0 1 Weak pullup. See Note 2.<br>1 0 High impedance. See Note 3.<br>1 1 Push-pull. See Note 4.<br>Port 0 is also the data I/O port for the Universal Peripheral Interface (UPI). When the UPI is enabled, port 0 must be configured as High-Z by the user. Input/Output through P0 is controlled by pin $\overline{CS}$, $\overline{WR}$, $\overline{RD}$, and A0. Output is push-pull when enabled. |
| P1.0-P1.5 | 3-8 | 5-9 | 42-44<br>1-3 | I/O | **Port 1:** Port 1 is a 6-bit bidirectional I/O port with Schmitt trigger inputs. Port 1 receives the control signals during program memory verification and parallel EPROM programming. During reset, port 1 is configured as a high impedance analog input port. Digital push-pull outputs are enabled by writing 1's to the P1M1 register. The programmer must take care to prevent digital outputs from switching while an A/D conversion is in progress.<br>**P1M1.X** **P1M2.X** **Mode Description**<br>0 0 A/D only. (High impedance)<br>0 1 Digital input only. High impedance (default).<br>1 X Push-pull.<br>Port 1 pins also serve alternate functions as follows: |
| | 3 | 4 | 42 | I/O | P1.0/ADIN0 |
| | 4 | 5 | 43 | I/O | P1.1/ADIN1 |
| | 5 | 6 | 44 | I/O | P1.2/ADIN2 |
| | 6 | 7 | 1 | I/O | P1.3/ADIN3 |
| | 7 | 8 | 2 | I/O | P1.4/ADIN4 |
| | 8 | 9 | 3 | I/O | P1.5/ADIN5 |
| P2.0-P2.7 | 21-28 | 24-31 | 18-25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port. Port 2 emits the high-order address byte during accesses to external program and data memory that use 16-bit addresses (MOVX @DPTR) (see Note 5). In this application, it uses strong internal pull-ups when emitting 1s. Port 2 receives the high-order address byte during program verification and parallel EPROM programming. During reset, the port 2 pullups are turned on synchronously, and the port register is loaded with 1's. Port 2 has the following output modes which can be selected on a per bit basis by writing to P2M1 and P2M0:<br>**P2M1.X** **P2M2.X** **Mode Description**<br>0 0 Open drain. See Note 1.<br>0 1 Weak pullup (default). See Note 2.<br>1 0 High impedance. See Note 3.<br>1 1 Push-pull. See Note 4.<br>Port 2 pins serve alternate functions as follows: |
| | 21 | 24 | 18 | | **P2.0** **CEX0** PCA module 0 external I/O<br>**CMP0** comparator 0 output |
| | 22 | 25 | 19 | | **P2.1** **CEX1** PCA module 1 external I/O<br>**CMP1** comparator 1 output |
| | 23 | 26 | 20 | | **P2.2** **CEX2** PCA module 2 external I/O<br>**CMP2** comparator 2 output |
| | 24 | 27 | 21 | | **P2.3** **CEX3** PCA module 3 external I/O<br>**CMP3** comparator 3 output |
| | 25 | 28 | 22 | | **P2.4** **T2EX** timer 2 capture input<br>**A0** UPI address input |
| | 26 | 29 | 23 | | **P2.5** **T2** timer 2 external I/O — clock-out (programmable)<br>$\overline{CS}$ UPI chip select input |
| | 27 | 30 | 24 | | **P2.6** **CEX4** PCA module 4 external I/O<br>**PWM0** Pulse width modulator 0 output |
| | 28 | 31 | 25 | | **P2.7** **ECI** PCA count input<br>**PWM1** Pulse width modulator 1 output |

## PIN DESCRIPTIONS (Continued)

| MNEMONIC | PIN NUMBER DIP | LCC | QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $+V_{REF}/AV_{CC}$ | 1 | 2 | 40 | I | **A/D positive power supply** |
| $-V_{REF}/AV_{SS}$ | 2 | 3 | 41 | I | **A/D 0V reference** |
| P3.0-P3.7 | 10-17 | 11, 13-19 | 5, 7-13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port. Port 3 pins that have 1s written to them can be used as inputs but will source current when externally pulled low (see DC Electrical Characteristics: $I_{IL}$). During reset all pins will be synchronously driven high and will remain high until written to by software. Port 3 has the following output modes which can be selected on a per bit basis by writing to P3M1 and P3M2: |
| | | | | | **P3M1.X** / **P3M2.X** / **Mode Description**<br>0 / 0 / Open drain. See Note 1.<br>0 / 1 / Weak pullup (default). See Note 2.<br>1 / 0 / High impedance. See Note 3.<br>1 / 1 / Push-pull. See Note 4. |
| | | | | | Port 3 pins serve alternate functions as follows: |
| | 10 | 11 | 5 | I | **P3.0** **RxD** Serial receive port |
| | 11 | 13 | 7 | O | **P3.1** **TxD** Serial transmit port (enabled only when transmitting serial data) |
| | 12 | 14 | 8 | I | **P3.2** **$\overline{INT0}$** External interrupt 0<br>**CMP3+** Comparator 3 positive input |
| | 13 | 15 | 9 | I | **P3.3** **$\overline{INT1}$** External interrupt 1<br>**CMP2+** Comparator 2 positive input |
| | 14 | 16 | 10 | I | **P3.4** **T0** Timer/counter 0 input<br>**CMP1+** Comparator 1 positive input |
| | 15 | 17 | 11 | I | **P3.5** **T1** Timer/counter 1 input<br>**CMPR–** Common reference to comparators 1, 2, 3 |
| | 16 | 18 | 12 | O | **P3.6** **$\overline{WR}$** External data memory write strobe<br>**CMP0+** Comparator 0 positive input |
| | 17 | 19 | 13 | O | **P3.7** **$\overline{RD}$** External data memory read strobe<br>**CMP0–** Comparator 0 negative input |
| $\overline{RST}$ | 9 | 10 | 4 | I | **Reset:** A low on this pin synchronously resets all port pins to a high state. The pin must be held low with the oscillator running for 24 oscillator cycles to initialize the internal registers. An internal diffused resistor to $V_{CC}$ permits a power on reset using only an external capacitor to $V_{SS}$. RST has a Schmitt trigger input stage to provide additional noise immunity with a slow rising input voltage. |
| ALE/$\overline{PROG}$ | 30 | 33 | 27 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. ALE is switched off if the bit 0 in the AUXR register (8EH) is set. This pin is also the program pulse input ($\overline{PROG}$) during parallel EPROM programming. (See also Internal Reset on page 3-779.) |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the device is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}/V_{PP}$ | 31 | 35 | 29 | I | **External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H to 1FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 1FFFH. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. If this pin is at $V_{PP}$ voltage during reset the device enters the in-circuit programming mode. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTES:**
1. When Open Drain mode is selected, ports 0 and 2 have weak pulldowns to guarantee positive leakage current (see DC electrical characteristic $I_{IH}$).
2. When Weak Pullup mode is selected, ports bits that have 1's written to them can be used as inputs but will source current when externally pulled low (see DC electrical characteristic $I_{IL}$).
3. When High Impedance mode is selected, all pullups and pulldowns are turned off. The only current sourced or sunk by the pin is the parasitic leakage current (see DC electrical characteristic $I_{L2}$ or $I_{LC}$, as applicable.
4. When Push-Pull mode is selected, strong pullups are on continuously when emitting 1's (see DC electrical characteristic $V_{OH}$).
5. When Open-Drain, Weak Pull-up, or Push-pull mode is selected.

**Table 1. 87C576 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| ADC0H# | A/D Channel 0 MSB | AAH | | | | | | | | | 00H |
| ADC1H# | A/D Channel 1 MSB | ABH | | | | | | | | | 00H |
| ADC2H# | A/D Channel 2 MSB | ACH | | | | | | | | | 00H |
| ADC3H# | A/D Channel 3 MSB | ADH | | | | | | | | | 00H |
| ADC4H# | A/D Channel 4 MSB | AEH | | | | | | | | | 00H |
| ADC5H# | A/D Channel 5 MSB | AFH | | | | | | | | | 00H |
| ADC0L# | A/D Channel 0 2-LSBits | 9AH | | | | | | | | | 00H |
| ADC1L# | A/D Channel 1 2-LSBits | 9BH | | | | | | | | | 00H |
| ADC2L# | A/D Channel 2 2-LSBits | 9CH | | | | | | | | | 00H |
| ADC3L# | A/D Channel 3 2-LSBits | 9DH | | | | | | | | | 00H |
| ADC4L# | A/D Channel 4 2-LSBits | 9EH | | | | | | | | | 00H |
| ADC5L# | A/D Channel 5 2-LSBits | 9FH | | | | | | | | | 00H |
| ADCON# | A/D Control | B1H | ADF | ADCE | AD8M | AMOD1 | AMOD0 | ASCA2 | ASCA1 | ASCA0 | 00H |
| ADCS# | A/D Channel Select | B2H | | | | | | | | | 00H |
| AUXR# | Auxiliary | 8EH | – | – | – | – | SRST | TXI | LO | AO | xxxx0000B |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| CCAP0H# | Module 0 Capture High | FAH | | | | | | | | | xxxxxxxxB |
| CCAP1H# | Module 1 Capture High | FBH | | | | | | | | | xxxxxxxxB |
| CCAP2H# | Module 2 Capture High | FCH | | | | | | | | | xxxxxxxxB |
| CCAP3H# | Module 3 Capture High | FDH | | | | | | | | | xxxxxxxxB |
| CCAP4H# | Module 4 Capture High | FEH | | | | | | | | | xxxxxxxxB |
| CCAP0L# | Module 0 Capture Low | EAH | | | | | | | | | xxxxxxxxB |
| CCAP1L# | Module 1 Capture Low | EBH | | | | | | | | | xxxxxxxxB |
| CCAP2L# | Module 2 Capture Low | ECH | | | | | | | | | xxxxxxxxB |
| CCAP3L# | Module 3 Capture Low | EDH | | | | | | | | | xxxxxxxxB |
| CCAP4L# | Module 4 Capture Low | EEH | | | | | | | | | xxxxxxxxB |
| CCAPM0# | Module 0 Mode | DAH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM1# | Module 1 Mode | DBH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM2# | Module 2 Mode | DCH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM3# | Module 3 Mode | DDH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| CCAPM4# | Module 4 Mode | DEH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| CCON*# | PCA Counter Control | D8H | CF | CR | – | CCF4 | CCF3 | CCF2 | CCF1 | CCF0 | 00x00000B |
| CH# | PCA Counter High | F9H | | | | | | | | | 00H |
| CL# | PCA Counter Low | E9H | | | | | | | | | 00H |
| CMOD# | PCA Counter Mode | D9H | CIDL | WDTE | – | – | – | CPS1 | CPS0 | ECF | 00xxx000B |
| | | | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 | |
| CMP*# | Comparator | C0H | EC3DP | EC2DP | EC1DP | EC0DP | C3RO | C2RO | C1RO | C0RO | 00H |
| CMPE# | Comparator Enable | 92H | EC3TDC | EC2TDC | EC1TDC | EC0TDC | EC3O | EC2O | EC1O | EC0O | 00H |
| DPTR: | Data Pointer (2 bytes) | | | | | | | | | | |
| DPH | Data Pointer High | 83H | | | | | | | | | 00H |
| DPL | Data Pointer Low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE0*# | Interrupt Enable 0 | A8H | EA | EC | ET2 | ES | ET1 | EX1 | ET0 | EX0 | 00H |
| IE1*# | Interrupt Enable 1 | E8H | EOB | EIB | EAD | EC4 | EC3 | EC2 | EC1 | EC0 | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

**Table 1. 87C576 Special Function Registers** (Continued)

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP0* | Interrupt Priority 0 | B8H | – | PPC | PT2 | PS | PT1 | PX1 | PT0 | PX0 | x0000000B |
| IP1*# | Interrupt Priority 1 | F8H | POB | PIB | PAD | PC4 | PC3 | PC2 | PC1 | PC0 | 00H |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | – | – | ADIN5 | ADIN4 | ADIN3 | ADIN2 | ADIN1 | ADIN0 | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | ECI | CEX4 | T2 | T2EX | CEX3 | CEX2 | CEX1 | CEX0 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TxD | RxD | FFH |
| P0M1# | Port 0 Output Mode 1 | 84H | | | | | | | | | 00H |
| P0M2# | Port 0 Output Mode 2 | 85H | | | | | | | | | 00H |
| P1M1# | Port 1 Output Mode 1 | 94H | | | | | | | | | 00H |
| P1M2# | Port 1 Output Mode 2 | 95H | | | | | | | | | 3FH |
| P2M1# | Port 2 Output Mode 1 | A4H | | | | | | | | | 00H |
| P2M2# | Port 2 Output Mode 2 | A5H | | | | | | | | | FFH |
| P3M1# | Port 3 Output Mode 1 | B4H | | | | | | | | | 00H |
| P3M2# | Port 3 Output Mode 2 | B5H | | | | | | | | | FFH |
| PCON | Power Control | 87H | SMOD1 | SMOD0 | OSF[1] | POF[1] | LVF[1] | WDTOF[1] | PD | IDL | 00xxxx00B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program Status Word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| PWCON# | PWM Control | BCH | – | – | – | – | PWMF | EN/$\overline{CLR}$ | PWE1 | PWE0 | 00H |
| PWMP# | PWM Prescaler | BDH | | | | | | | | | 00H |
| PWM0# | PWM Register 0 | BEH | | | | | | | | | 00H |
| PWM1# | PWM Register 1 | BFH | | | | | | | | | 00H |
| RACAP2H# | Timer 2 Capture High | CBH | | | | | | | | | 00H |
| RACAP2L# | Timer 2 Capture Low | CAH | | | | | | | | | 00H |
| SADDR# | Slave Address | A9H | | | | | | | | | 00H |
| SADEN# | Slave Address Mask | B9H | | | | | | | | | 00H |
| SBUF | Serial Data Buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial Control | 98H | SM0/FE | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack Pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer Control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| | | | CF | CE | CD | CC | CB | CA | C9 | C8 | |
| T2CON* | Timer 2 Control | C8H | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/T2 | CP/RL2 | 00H |
| T2MOD# | Timer 2 Mode Control | C9H | – | – | – | – | – | – | T2OE[2] | DCEN | xxxxxxx0B |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
1. Reset value depends on reset source.
2. Programmable clock-out

**Table 1. 87C576 Special Function Registers** (Continued)

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB LSB | | | | | | | | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| TH0 | Timer High 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer High 1 | 8DH | | | | | | | | | 00H |
| TH2# | Timer High 2 | CDH | | | | | | | | | 00H |
| TL0 | Timer Low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer Low 1 | 8BH | | | | | | | | | 00H |
| TL2# | Timer Low 2 | CCH | | | | | | | | | 00H |
| TMOD | Timer Mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |
| UCS# | UPI Control/Status | 86H | ST7 | ST6 | ST5 | ST4 | UE | AF | IBF | OBE/OBF | 00H |
| | | | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 | |
| WDCON# | Watchdog Timer Control | C4H | PRE2 | PRE1 | PRE0 | LVRE | OFRE | DPD | WDRUN | WDMOD | 11111111B |
| WDL# | Watchdog Timer Reload | C1H | | | | | | | | | 00H |
| WFEED1# | Watchdog Feed 1 | C2H | | | | | | | | | xxH |
| WFEED2# | Watchdog Feed 2 | C3H | | | | | | | | | xxH |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
1. Reset value depends on reset source.

The 8XC576 has a number of failure detect circuits to prevent abnormal operating conditions. these failure detect circuits generate resets as shown in Figure 1.

## POWER ON CLEAR / POWER ON FLAG

An on-chip Power On Detect Circuit resets the 8XC576 and sets the Power Off Flag (PCON.4) on power up or if $V_{CC}$ drops to zero momentarily. The POF can only be cleared by software. The RST pin is not driven by the power on detect circuit. The POF can be read by software to determine that a power failure has occurred and can also be set by software.

## LOW VOLTAGE DETECT

An on-chip Low Voltage Detect circuit sets the Low Voltage Flag (PCON.3) if $V_{CC}$ drops below $V_{LOW}$ (see DC Electrical Characteristics) and resets the 8XC576 if the Low Voltage Reset Enable bit (WDCON.4) is set. If the LVRE is cleared, the reset is disabled but LVF will still be set if $V_{CC}$ is low. The RST pin is not driven by the low voltage detect circuit. The LVF can be read by software to determine that $V_{CC}$ was low. The LVF can be set or cleared by software.

## OSCILLATOR FAIL DETECT

An on-chip Oscillator Fail Detect circuit sets the Oscillator Fail Flag (PCON.5) if the oscillator frequency drops below OSCF for one or more cycles (see AC Electrical Characteristics: OSCF) and resets the 8XC576 if the Oscillator Fail Reset Enable bit (WDCON.3) is set. If OFRE is cleared, the reset is disabled but OSF will still be set if the oscillator fails. The RST pin is not driven by the oscillator fail detect circuit. The OSF can be read by software to determine that an oscillator failure has occurred. The OSF can be set or cleared by software.

## LOW ACTIVE RESET

One of the most notable features on this part is the low active reset. The low active reset operates exactly the same as high active reset with the exception that the part is put into the reset mode by applying a low level to the reset pin. For power-on reset it is also necessary to invert the power-on reset circuit; connecting the 8.2K resistor from the reset pin to $V_{CC}$ and the 10µf capacitor from the reset pin to ground. Figure 1 shows the reset related circuitry.

When reset the port pins on the 8XC576 are driven high synchronously.

The 8XC576 also has Low voltage detection circuitry that will, if enabled, force the part to reset when $V_{CC}$ (on the part) fails below a set level. Low Voltage Reset is enabled by a normal reset. Low Voltage Reset can be disabled by clearing LVRE (bit 4 in the WDCON SFR) then executing a watchdog feed sequence (A5H to WFEED1 followed immediately by 5AH to WFEED2). In addition there is a flag (LVF) that is set if a low voltage condition is detected. The LVF flag is set even if the Low Voltage detection circuitry is disabled. Notice that the Low voltage detection circuitry does not drive the RST# pin so the LVF flag is the only way that the microcontroller can determine if it has been reset due to a low voltage condition.

The 8XC576 has an on-chip power-on detection circuit that sets the POF (PCON.4) flag on power up or if the $V_{CC}$ level momentarily drops to 0V. This flag can be used to determine if the part is being started from a power-on (cold start) or if a reset has occurred due to another condition (warm start).

The 8XC576 can be reset in software by setting the RST bit of the AUXR register (AUXR.3). See Figure 1 for reset diagram.

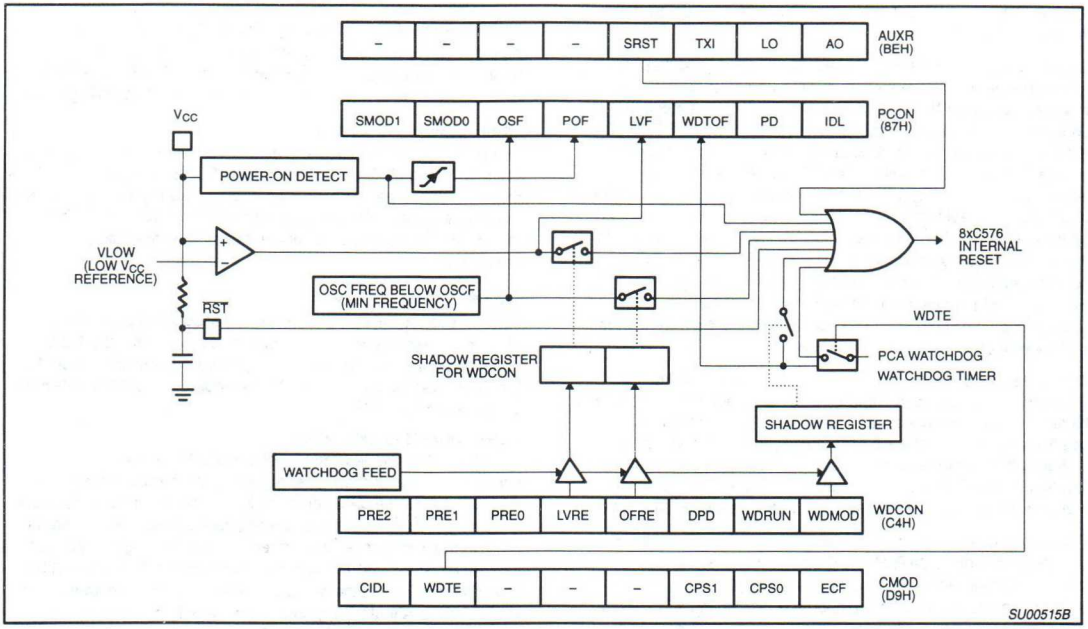



**Figure 1. Reset Circuitry**

## TIMERS

The 8XC576 has four on-chip timers.

Timers 0 and 1 are identical in every way to Timers 0 and 1 on the 80C51.

Timer 2 on the 8XC576 is identical to the 80C52 Timer 2 (described in detail in the 80C52 overview) with the exception that it is an up or down counter. To configure the Timer to count down the DCEN bit in the T2MOD special function register must be set and a low level must be present on the T2EX pin (P1.1).

The Pulse Width Modulator (PWM) system can be used as a timer by disabling its outputs and monitoring its counter overflow flag, the PWMF bit in the PWCON register (see the PWM section for details).

The Watchdog timer operation and implementation is similar to the 8XC550 (for additional information see the 8XC550 datasheet) with the exception that the reset values of the WDCON and WDL special function registers have been changed. The changes in these registers cause the watchdog timer to be enabled with a timeout of $16384 \times T_{OSC}$ when the part is reset. The watchdog can be disabled by executing a valid feed sequence and then clearing WDRUN (bit 2 in the WDCON SFR). In timer mode, the timer is controlled by toggling the WDRUN bit. The timeout flag, WDTOF, is set when the timer overflows and must be cleared in software.

## PROGRAMMABLE COUNTER ARRAY (PCA)

The Programmable Counter Array is a special Timer that has five 16-bit capture/compare modules associated with it. Each of the modules can be programmed to operate in one of four modes: rising and/or falling edge capture, software timer, high-speed output, or pulse width modulator. Each module has a pin associated with it in port 2. Module 0 is connected to P2.0(CEX0), module 1 to P2.1(CEX1), etc. The basic PCA configuration is shown in Figure 2.

The PCA timer is a common time base for all five modules and can be programmed to run at: 1/12 the oscillator frequency, 1/4 the oscillator frequency, the Timer 0 overflow, or the input on the ECI pin (P2.7). The timer count source is determined from the CPS1 and CPS0 bits in the CMOD SFR as follows (see Figure 3):

| CPS1 | CPS0 | PCA Timer Count Source |
|---|---|---|
| 0 | 0 | 1/12 oscillator frequency |
| 0 | 1 | 1/4 oscillator frequency |
| 1 | 0 | Timer 0 overflow |
| 1 | 1 | External Input at ECI pin (P2.7) |

In the CMOD SFR are three additional bits associated with the PCA. They are CIDL which allows the PCA to stop during idle mode, WDTE which enables or disables the watchdog function on module 4, and ECF which when set causes an interrupt and the PCA overflow flag CF (in the CCON SFR) to be set when the PCA timer overflows. These functions are shown in Figure 3.

The watchdog timer function is implemented in module 4 as implemented in other parts that have a PCA that are available on the market. However, if a watchdog timer is required in the target application, it is recommended to use the hardware watchdog timer that is implemented on the 87C576 separately from the PCA (see Figure 15).

The CCON SFR contains the run control bit for the PCA and the flags for the PCA timer (CF) and each module (refer to Figure 6). To run the PCA the CR bit (CCON.6) must be set by software. The PCA is shut off by clearing this bit. The CF bit (CCON.7) is set when the PCA counter overflows and an interrupt will be generated if the ECF bit in the CMOD register is set, The CF bit can only be cleared by software. Bits 0 through 4 of the CCON register are the flags for the modules (bit 0 for module 0, bit 1 for module 1, etc.) and are set by hardware when either a match or a capture occurs. These flags

also can only be cleared by software. The PCA interrupt system shown in Figure 4.

Each module in the PCA has a special function register associated with it. These registers are: CCAPM0 for module 0, CCAPM1 for module 1, etc. (see Figure 7). The registers contain the bits that control the mode that each module will operate in. The ECCF bit (CCAPMn.0 where n=0, 1, 2, 3, or 4 depending on the module) enables the CCF flag in the CCON SFR to generate an interrupt when a match or compare occurs in the associated module. PWM (CCAPMn.1) enables the pulse width modulation mode. The TOG bit (CCAPMn.2) when set causes the CEX output associated with the module to toggle when there is a match between the PCA counter and the module's capture/compare register. The match bit MAT (CCAPMn.3) when set will cause the CCFn bit in the CCON register to be set when there is a match between the PCA counter and the module's capture/compare register.

The next two bits CAPN (CCAPMn.4) and CAPP (CCAPMn.5) determine the edge that a capture input will be active on. The CAPN bit enables the negative edge, and the CAPP bit enables the positive edge. If both bits are set both edges will be enabled and a capture will occur for either transition. The last bit in the register ECOM (CCAPMn.6) when set enables the comparator function. Figure 8 shows the CCAPMn settings for the various PCA functions.

There are two additional registers associated with each of the PCA modules. They are CCAPnH and CCAPnL and these are the registers that store the 16-bit count when a capture occurs or a compare should occur. When a module is used in the PWM mode these registers are used to control the duty cycle of the output.

**PCA Capture Mode**

To use one of the PCA modules in the capture mode either one or both of the CCAPM bits CAPN and CAPP for that module must be set. The external CEX input for the module (on port 2) is sampled for a transition. When a valid transition occurs the PCA hardware loads the value of the PCA counter registers (CH and CL) into the module's capture registers (CCAPnL and CCAPnH). If the CCFn bit for the module in the CCON SFR and the ECCFn bit in the CCAPMn SFR are set then an interrupt will be generated. Refer to Figure 9.

**16-bit Software Timer Mode**

The PCA modules can be used as software timers by setting both the ECOM and MAT bits in the modules CCAPMn register. The PCA timer will be compared to the module's capture registers and when a match occurs an interrupt will occur if the CCFn (CCON SFR) and the ECCFn (CCAPMn SFR) bits for the module are both set (see Figure 10).

**High Speed Output Mode**

In this mode the CEX output (on port 2) associated with the PCA module will toggle each time a match occurs between the PCA counter and the module's capture registers. To activate this mode the TOG, MAT, and ECOM bits in the module's CCAPMn SFR must be set (see Figure 11).

**Pulse Width Modulator Mode**

All of the PCA modules can be used as PWM outputs. Figure 12 shows the PWM function. The frequency of the output depends on the source for the PCA timer. All of the modules will have the same frequency of output because they all share the PCA timer. The duty cycle of each module is independently variable using the module's capture register CCAPLn. When the value of the PCA CL SFR is less than the value in the module's CCAPLn SFR the output will be low, when it is equal to or greater than the output will be high. When CL overflows from FF to 00, CCAPLn is reloaded with the value in CCAPHn. the allows updating the PWM without glitches. The PWM and ECOM bits in the module's CCAPMn register must be set to enable the PWM mode.

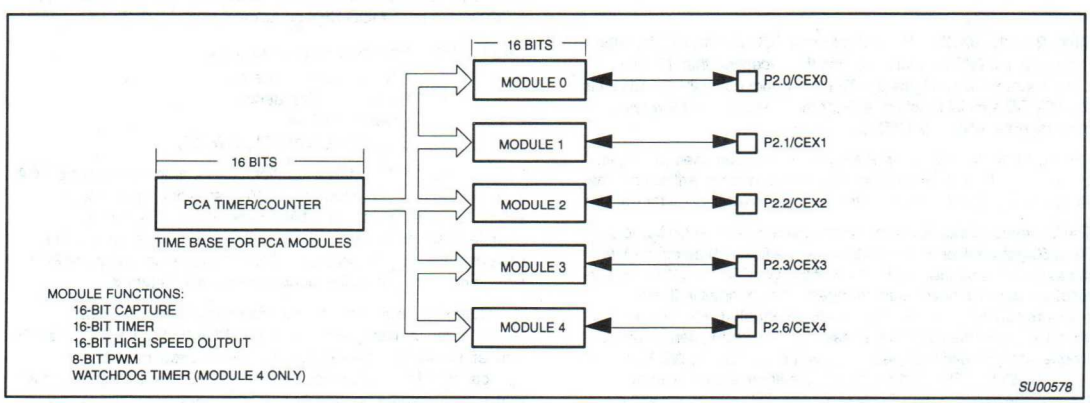



**Figure 2. Programmable Counter Array (PCA)**

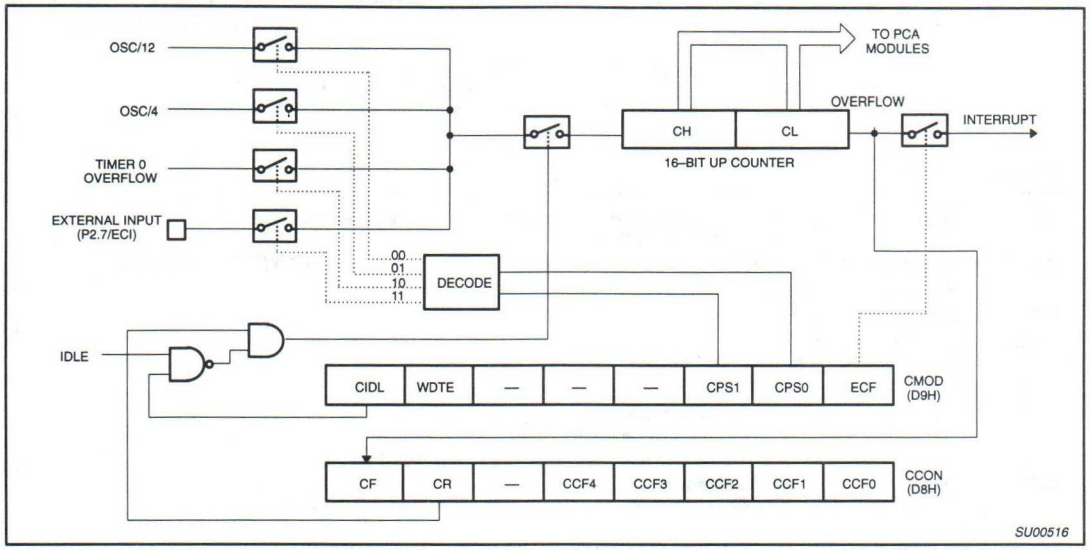



**Figure 3. PCA Timer/Counter**

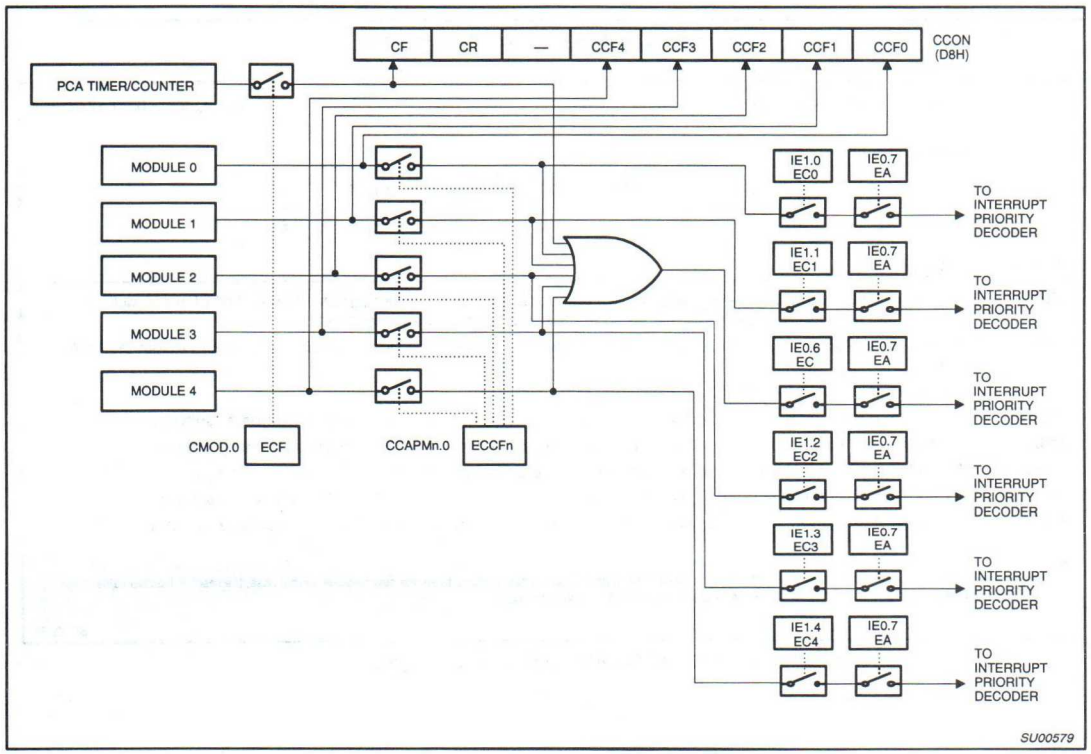



**Figure 4. PCA Interrupt System**

**CMOD** Address = OD9H Reset Value = 00XX X000B

| | CIDL | WDTE | – | – | – | CPS1 | CPS0 | ECF |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Symbol | Function |
|---|---|
| **CIDL** | Counter Idle control: CIDL = 0 programs the PCA Counter to continue functioning during idle Mode. CIDL = 1 programs it to be gated off during idle. |
| **WDTE** | Watchdog Timer Enable: WDTE = 0 disables Watchdog Timer function on PCA Module 4. WDTE = 1 enables it. |
| – | Not implemented, reserved for future use.* |
| **CPS1** | PCA Count Pulse Select bit 1. |
| **CPS0** | PCA Count Pulse Select bit 0. |

| CPS1 | CPS0 | Selected PCA Input** | |
|---|---|---|---|
| 0 | 0 | 0 | Internal clock, $f_{OSC} \div 12$ |
| 0 | 1 | 1 | Internal clock, $f_{OSC} \div 4$ |
| 1 | 0 | 2 | Timer 0 overflow |
| 1 | 1 | 3 | External clock at ECI/P2.7 pin (max. rate = $f_{OSC} \div 8$) |

| | |
|---|---|
| **ECF** | PCA Enable Counter Overflow interrupt: ECF = 1 enables CF bit in CCON to generate an interrupt. ECF = 0 disables that function of CF. |

NOTE:
* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.
** $-f_{OSC}$ = oscillator frequency

SU00686A

**Figure 5. CMOD: PCA Counter Mode Register**

**CCON** Address = OD8H Reset Value = 00X0 0000B

Bit Addressable

| | CF | CR | – | CCF4 | CCF3 | CCF2 | CCF1 | CCF0 |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Symbol | Function |
|---|---|
| **CF** | PCA Counter Overflow flag. Set by hardware when the counter rolls over. CF flags an interrupt if bit ECF in CMOD is set. CF may be set by either hardware or software but can only be cleared by software. |
| **CR** | PCA Counter Run control bit. Set by software to turn the PCA counter on. Must be cleared by software to turn the PCA counter off. |
| – | Not implemented, reserved for future use*. |
| **CCF4** | PCA Module 4 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF3** | PCA Module 3 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF2** | PCA Module 2 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF1** | PCA Module 1 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |
| **CCF0** | PCA Module 0 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |

NOTE:
* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

SU00036

**Figure 6. CCON: PCA Counter Control Register**

**CCAPMn** Address CCAPM0 0DAH
CCAPM1 0DBH
CCAPM2 0DCH
CCAPM3 0DDH
CCAPM4 0DEH

Reset Value = X000 0000B

Not Bit Addressable

| – | ECOMn | CAPPn | CAPNn | MATn | TOGn | PWMn | ECCFn |
|---|---|---|---|---|---|---|---|
| Bit: 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Symbol | Function |
|---|---|
| – | Not implemented, reserved for future use*. |
| **ECOMn** | Enable Comparator. ECOMn = 1 enables the comparator function. |
| **CAPPn** | Capture Positive, CAPPn = 1 enables positive edge capture. |
| **CAPNn** | Capture Negative, CAPNn = 1 enables negative edge capture. |
| **MATn** | Match. When MATn = 1, a match of the PCA counter with this module's compare/capture register causes the CCFn bit in CCON to be set, flagging an interrupt. |
| **TOGn** | Toggle. When TOGn = 1, a match of the PCA counter with this module's compare/capture register causes the CEXn pin to toggle. |
| **PWMn** | Pulse Width Modulation Mode. PWMn = 1 enables the CEXn pin to be used as a pulse width modulated output. |
| **ECCFn** | Enable CCF interrupt. Enables compare/capture flag CCFn in the CCON register to generate an interrupt. |

**NOTE:**
*User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

*SU00037*

**Figure 7. CCAPMn: PCA Modules Compare/Capture Registers**

| – | ECOMn | CAPPn | CAPNn | MATn | TOGn | PWMn | ECCFn | MODULE FUNCTION |
|---|---|---|---|---|---|---|---|---|
| X | 0 | 0 | 0 | 0 | 0 | 0 | 0 | No operation |
| X | X | 1 | 0 | 0 | 0 | 0 | X | 16-bit capture by a positive-edge trigger on CEXn |
| X | X | 0 | 1 | 0 | 0 | 0 | X | 16-bit capture by a negative trigger on CEXn |
| X | X | 1 | 1 | 0 | 0 | 0 | X | 16-bit capture by a transition on CEXn |
| X | 1 | 0 | 0 | 1 | 0 | 0 | X | 16-bit Software Timer |
| X | 1 | 0 | 0 | 1 | 1 | 0 | X | 16-bit High Speed Output |
| X | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 8-bit PWM |
| X | 1 | 0 | 0 | 1 | X | 0 | X | Watchdog Timer |

**Figure 8. PCA Module Modes (CCAPMn Register)**

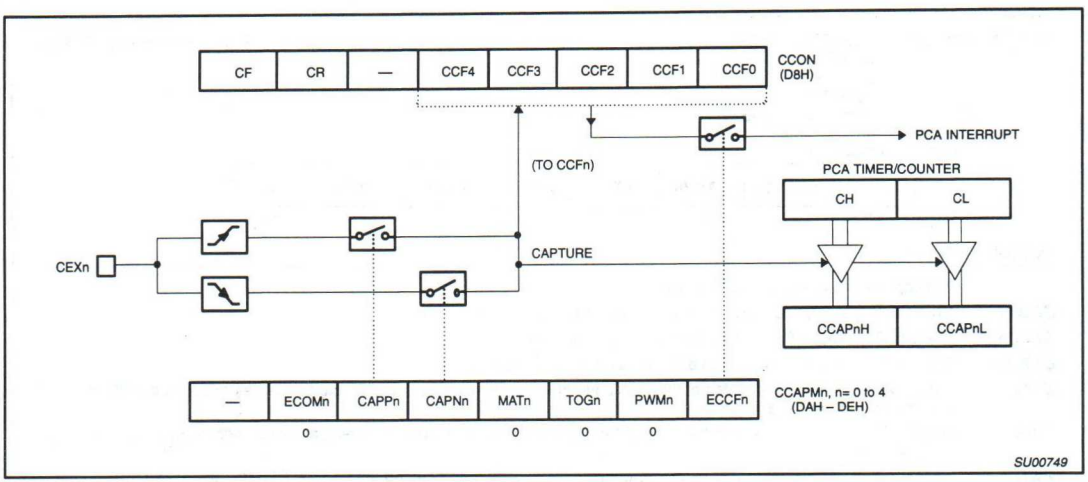



**Figure 9. PCA Capture Mode**

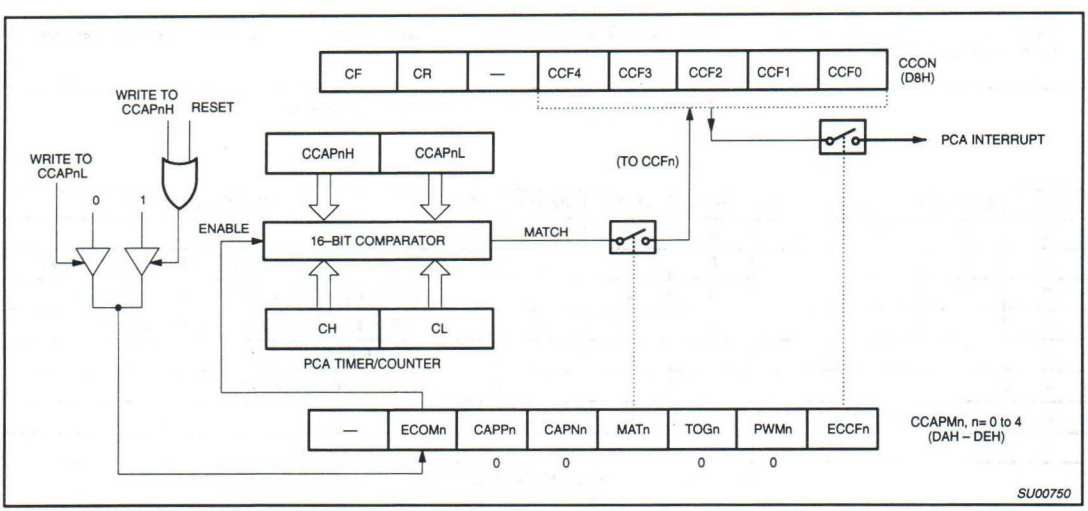



**Figure 10. PCA Compare Mode**

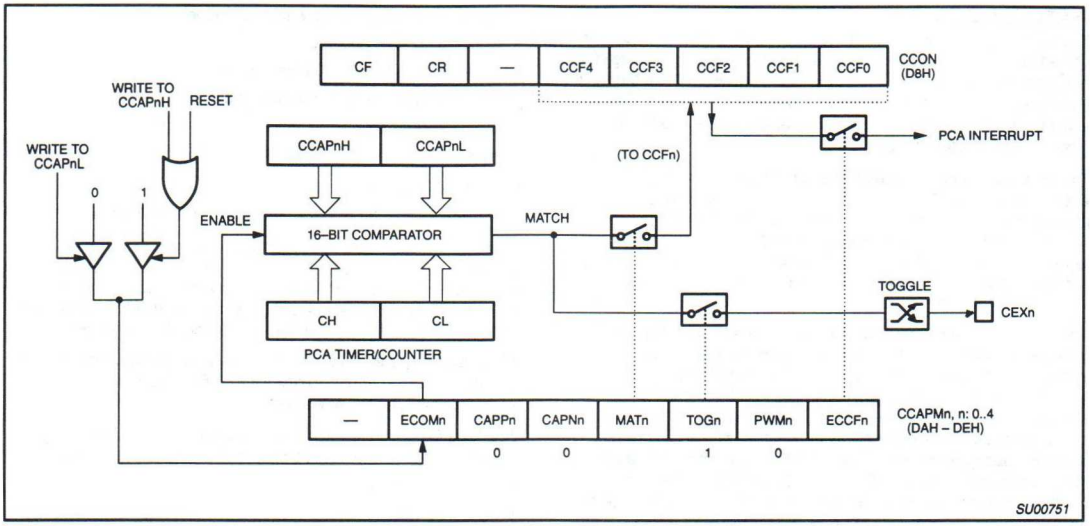



**Figure 11. PCA High Speed Output Mode**

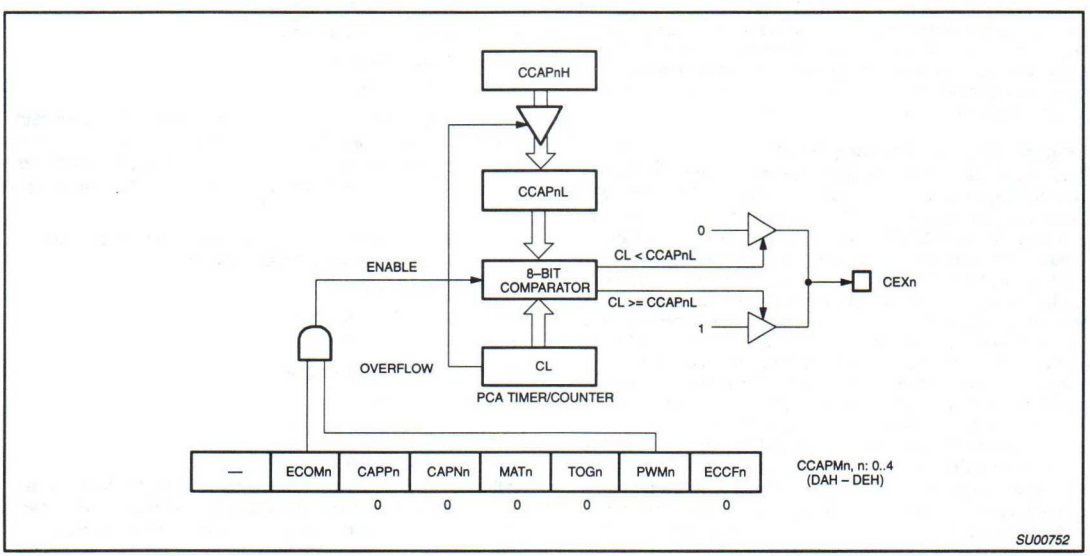



**Figure 12. PCA PWM Mode**

## WATCHDOG TIMER

The watchdog timer is not directly loadable by the user. Instead, the value to be loaded into the main timer is held in an autoload register or is part of the mask ROM programming. In order to cause the main timer to be loaded with the appropriate value, a special sequence of software action must take place. This operation is referred to as feeding the watchdog timer.

To feed the watchdog, two instructions must be sequentially executed successfully. No intervening instruction fetches are allowed, so interrupts should be disabled before feeding the watchdog. The instructions should move A5H to the WFEED1 register and then 5AH to the WFEED2 register. If WFEED1 is correctly loaded and WFEED2 is not correctly loaded, then an immediate underflow will occur.

The watchdog timer subsystem has two modes of operation. Its principal function is a watchdog timer. In this mode it protects the system from incorrect code execution by causing a system reset when the watchdog timer underflows as a result of a failure of software to feed the timer prior to the timer reaching its terminal count. If the user does not employ the watchdog function, the watchdog subsystem can be used as a timer. In this mode, reaching the terminal count sets a flag. In most other respects, the timer mode possesses the characteristics of the watchdog mode. This is done to protect the integrity of the watchdog function.

The watchdog timer subsystem consists of a prescaler and a main counter. The prescaler has 8 selectable taps off the final stages and the output of a selected tap provides the clock to the main counter. The main counter is the section that is loaded as a result of the software feeding the watchdog and it is the section that causes the system reset (watchdog mode) or time-out flag to be set (timer mode) if allowed to reach its terminal count.

### Programming the Watchdog Timer

Both the EPROM and ROM devices have a set of SFRs for holding the watchdog autoload values and the control bits. The watchdog time-out flag is present in the PCON register and operates the same in all versions. In the EPROM device, the watchdog parameters (autoload value and control) are always taken from the SFRs. In the ROM device, the watchdog parameters can be mask programmed or taken from the SFRs. The selection to take the watchdog parameters from the SFRs or from the mask programmed values is controlled by EA (external access). When EA is high (internal ROM access), the watchdog parameters are taken from the mask programmed values. If the watchdog is mask programmed to the timer mode, then the autoload values and the pre-scaler taps are taken from the SFRs. When EA is low (external access), the watchdog parameters are taken from the SFRs. The user should be able to leave code in his program which initializes the watchdog SFRs even though he has migrated to the mask ROM part. This allows no code changes from EPROM prototyping to ROM coded production parts. The run control bit only functions in timer mode and does not require a feed sequence to modify.

### Watchdog Detailed Operation

**EPROM Device (and ROMless Operation: EA = 0)**

In the ROMless operation (ROM part, EA = 0) and in the EPROM device, the watchdog operates in the following manner (see Figure 15).

Whether the watchdog is in the watchdog or timer mode, when external RESET is applied, the following takes place:

- Watchdog mode bit set to watchdog mode.
- Watchdog is running.
- Autoload register set to 00 (min. count).
- Watchdog time-out flag is unchanged.
- Prescaler is cleared.
- Prescaler tap set to the highest divide.
- Autoload takes place.

The watchdog can be fed even though it is in the timer mode.

Note that the operational concept is for the watchdog mode of operation, when coming out of a hardware reset, the software should load the autoload registers, set the mode to watchdog, clear the watchdog timeout flag, and then feed the watchdog (cause an autoload). The watchdog will now be starting at a known point.

If the watchdog is in the watchdog mode and running and happens to underflow at the time the external RESET is applied, the watchdog time-out flag will be set.

When the watchdog is in the watchdog mode and the watchdog underflows, the following action takes place (see Figure 17):

- Autoload takes place.
- Watchdog time-out flag is set
- Mode bit unchanged.
- Watchdog run bit unchanged.
- Autoload register unchanged.
- Prescaler tap unchanged.
- All other device action same as external reset.

Note that if the watchdog underflows, the program counter will start from 00H as in the case of an external reset. The watchdog time-out flag can be examined to determine if the watchdog has caused the reset condition. The watchdog time-out flag bit must be cleared by software.

When the watchdog is in the timer mode and the timer software underflows, the following action takes place:

- Autoload takes place.
- Watchdog time-out flag is set
- Mode bit unchanged.
- Watchdog run bit unchanged.
- Autoload register unchanged.
- Prescaler tap unchanged.

**Mask ROM Device (EA = 1)**

In the mask ROM device, the watchdog mode bit (WDMOD) is mask programmed and the bit in the watchdog command register is read only and reflects the mask programmed selection. If the mask programmed mode bit selects the timer mode, then the watchdog run bit (WDRUN) operates as described under EPROM Device. If the mask programmed bit selects the watchdog mode, then the watchdog run bit has no effect on the timer operation (see Figure 16).

**Watchdog Function**

The watchdog consists of a programmable prescaler and the main timer. The prescaler derives its clock from the on-chip oscillator. The prescaler consists of a divide by 2 followed by a 13 stage upcounter with taps from stage 6 through stage 13. This is shown in Figure 18.

The tap selection is programmable. The watchdog main counter is a down counter clocked (decremented) each time the programmable prescaler overflows. The watchdog generates an underflow signal (and is autoloaded) when the watchdog is at count 0 and the prescaler clock decrements the watchdog. The watchdog is 8 bits long and the autoload value can range from 0 to FFH. (The autoload value of 0 is permissible since the prescaler is cleared upon autoload).

This leads to the following user design equations. Definitions :$t_{OSC}$ is the oscillator period, N is the selected prescaler tap value, W is the main counter autoload value, $t_{MIN}$ is the minimum watchdog time-out value (when the autoload value is 0), $t_{MAX}$ is the maximum time-out value (when the autoload value is FFH), $t_D$ is the design time-out value.

$t_{MIN} = t_{OSC} \times 2 \times 64$

$t_{MAX} = t_{MIN} \times 128 \times 256$

$t_D = t_{MIN} \times 2^{PRESCALER} \times (W + 1)$
(where prescaler = 0, 1, 2, 3, 4, 5, 6, or 7)

Note that the design procedure is anticipated to be as follows. A $t_{MAX}$ will be chosen either from equipment or operation considerations and will most likely be the next convenient value higher than $t_D$. (If the watchdog were inadvertently to start from 00H, an underflow would be guaranteed, barring other anomalies, to occur within $t_{MAX}$).

The software must be written so that a feed operation takes place every $t_D$ seconds from the last feed operation. Some tradeoffs may need to be made. It is not advisable to include feed operations in minor loops or in subroutines unless the feed operation is a specific subroutine.

**Watchdog Control Register (WDCON)**
**Address C4H**
The following bits of this register are read only in the ROM part when $\overline{EA}$ is high: WDMOD, DPD, OFRE, LVRE, PRE0, PRE1, and PRE2. That is, the register will reflect the mask programmed values. In the ROM part with $\overline{EA}$ high, these bits are taken from mask coded bits and are not readable by the program. WDRUN is read only in the ROM part when $\overline{EA}$ is high and WDMOD is in the watchdog mode. When WDMOD is in the timer mode, WDRUN functions normally.

The parameters written into WDMOD, DPD, OFRE, LVRE, PRE0, PRE1, and PRE2 by the program are not applied directly to the watchdog timer subsystem. The watchdog timer subsystem is directly controlled by a second register which stores these bits. The transfer of these bits from the user register to the second control register takes place when the watchdog is fed. This prevents random code execution from directly foiling the watchdog function. This does not affect the operation where these bits are taken from mask coded values.

The reset values of the WDCON and WDL registers will be such that the timer resets to the watchdog mode with a timeout period of $2 \times 64 \times 128 \times t_{OSC}$. The watchdog timer does not generate an interrupt.

Additional bits in WDCON are used to disable reset generation by the oscillator fail and low voltage detect circuits. WDCON can be written by software only by executing a valid watchdog feed sequence.

**WDCON Register Bit Definitions**

| | | |
|---|---|---|
| WDCON.7 | PRE2 | Prescaler Select 2, reset to 1 |
| WDCON.6 | PRE1 | Prescaler Select 1, reset to 1 |
| WDCON.5 | PRE0 | Prescaler Select 0, reset to 1 |
| WDCON.4 | LVRE | Low Voltage Reset Enable, reset to 1 (enabled) |
| WDCON.3 | OFRE | Oscillator Fail Reset Enable, reset to 1 (enabled) |
| WDCON.2 | DPD | Disable Power Down |
| WDCON.1 | WDRUN | Watchdog Run, reset to 1 (enabled) |
| WDCON.0 | WDMOD | Watchdog Mode, reset to 1 (watchdog mode) |

## Enhanced UART

The UART operates in all of the usual modes that are described in the first section of this book for the 80C51. In addition the UART can perform framing error detect by looking for missing stop bits, and automatic address recognition. The 8XC576 UART also fully supports multiprocessor communication as does the standard 80C51 UART.

When used for framing error detect the UART looks for missing stop bits in the communication. A missing bit will set the FE bit in the SCON register. The FE bit shares the SCON.7 bit with SM0 and the function of SCON.7 is determined by PCON.6 (SMOD0) (see Figure 20). If SMOD0 is set then SCON.7 functions as FE. SCON.7 functions as SM0 when SMOD0 is cleared. When used as FE SCON.7 can only be cleared by software. Refer to Figure 19.

The serial port transmitter data can be inverted by setting the TXI (AUXR.2) bit. For normal operation, the TXI bit should be cleared.

**Automatic Address Recognition**
Automatic Address Recognition is a feature which allows the UART to recognize certain addresses in the serial bit stream by using hardware to make the comparisons. This feature saves a great deal of software overhead by eliminating the need for the software to examine every serial address which passes by the serial port. This feature is enabled by setting the SM2 bit in SCON. In the 9 bit UART modes, mode 2 and mode 3, the Receive Interrupt flag (RI) will be automatically set when the received byte contains either the "Given" address or the "Broadcast" address. The 9 bit mode requires that the 9th information bit is a 1 to indicate that the received information is an address and not data. Automatic address recognition is shown in Figure 21.

The 8 bit mode is called Mode 1. In this mode the RI flag will be set if SM2 is enabled and the information received has a valid stop bit following the 8 address bits and the information is either a Given or Broadcast address.

Mode 0 is the Shift Register mode and SM2 is ignored.

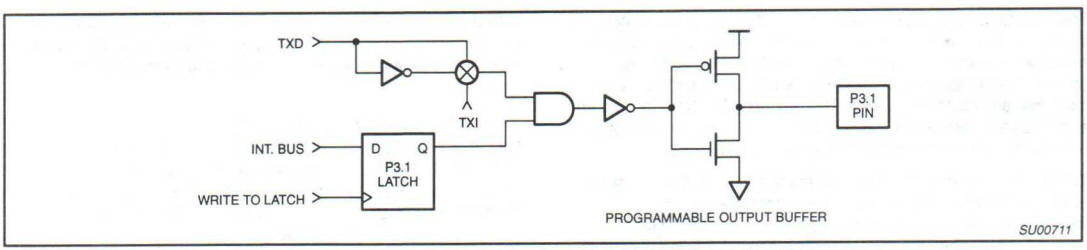



**Figure 13. TXI (AUXR.2) Bit Inverts the TxD Pin (P3.1) When Set**

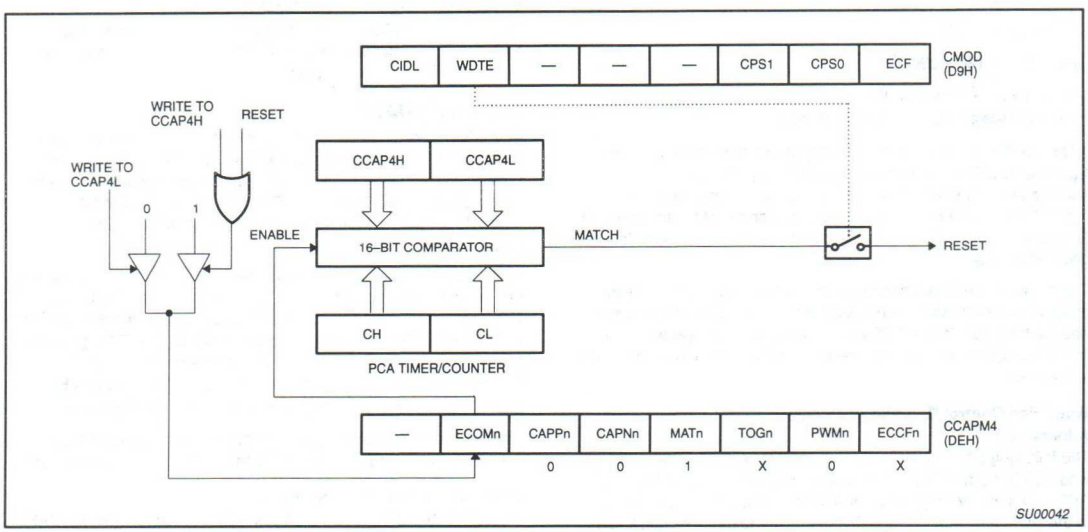



**Figure 14. PCA Watchdog Timer**

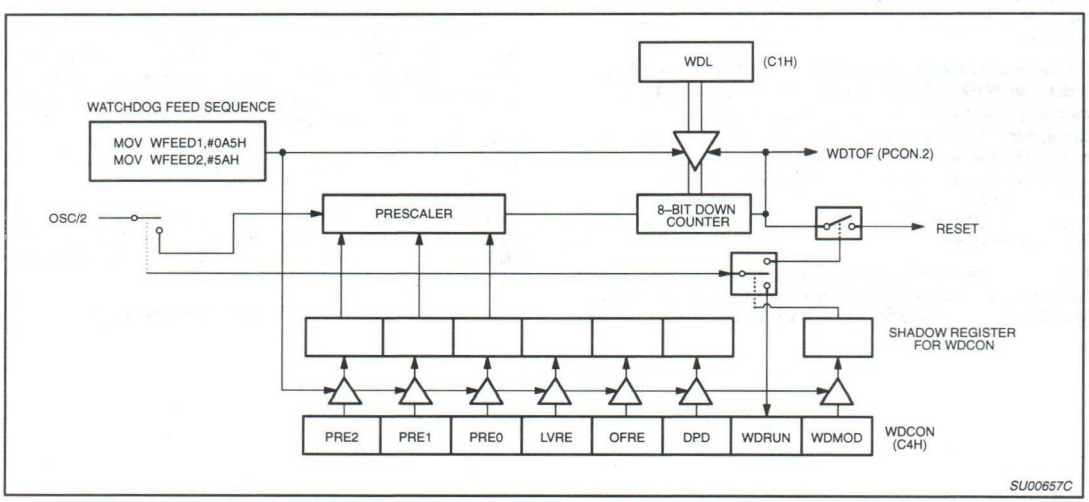



**Figure 15. Watchdog Timer in 87C576 and 80C576 / 83C576 ($\overline{EA}$ = 0)**

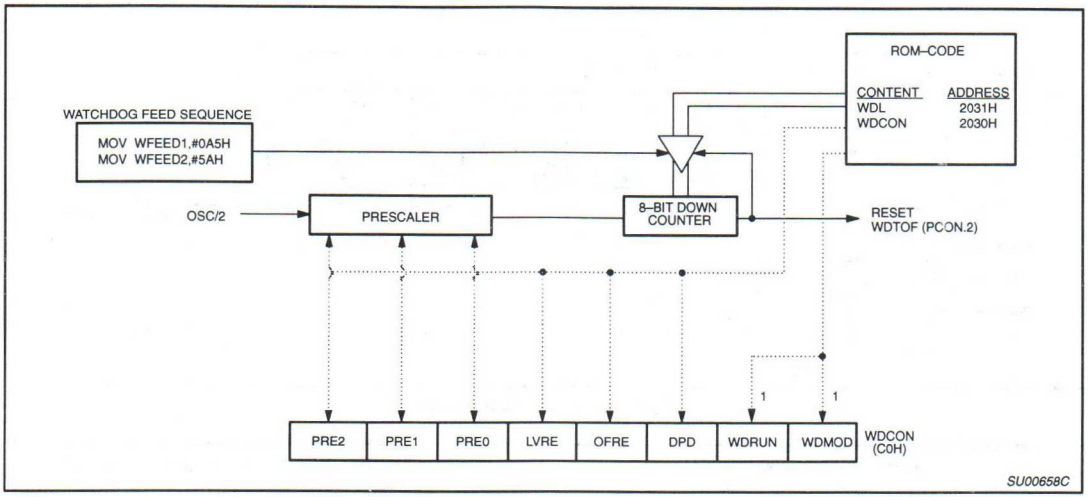



**Figure 16. Watchdog Timer of 83C576 in Watchdog Mode ($\overline{EA}$ = 1, WDMOD = 1)**

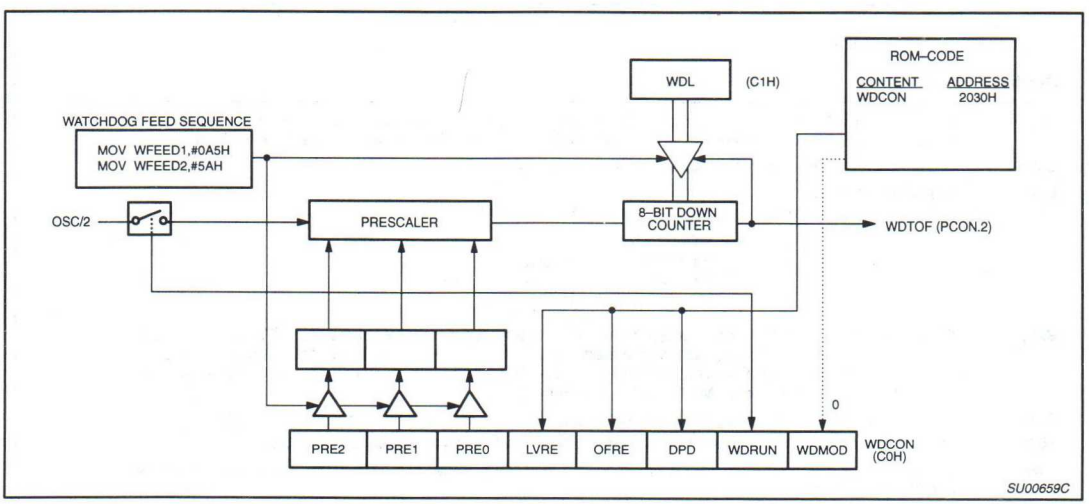



**Figure 17. Watchdog Timer of 83C576 in Timer Mode ($\overline{EA}$ = 1, WDMOD = 0)**

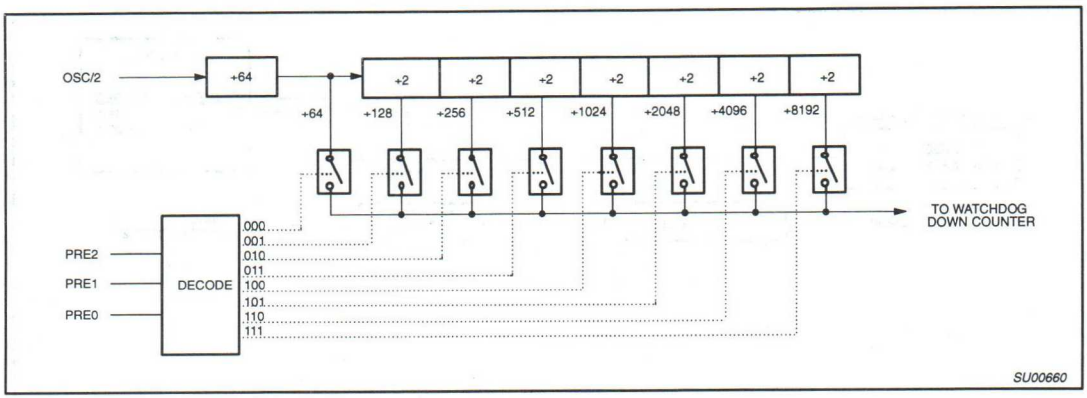



**Figure 18. Watchdog Prescaler**

**SCON** Address = 98H Reset Value = 0000 0000B

Bit Addressable

| | SM0/FE | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | (SMOD0/1)* | | | | | | | |

| Symbol | Function |
|---|---|
| **FE** | Framing Error bit. This bit is set by the receiver when an invalid stop bit is detected. The FE bit is not cleared by valid frames but should be cleared by software. The SMOD0 bit must be set to enable access to the FE bit. |
| **SM0** | Serial Port Mode Bit 0, (SMOD0 must = 0 to access bit SM0) |
| **SM1** | Serial Port Mode Bit 1 |

| SM0 | SM1 | Mode | Description | Baud Rate** |
|---|---|---|---|---|
| 0 | 0 | 0 | shift register | $f_{OSC}/12$ |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{OSC}/64$ or $f_{OSC}/32$ |
| 1 | 1 | 3 | 9-bit UART | variable |

| | |
|---|---|
| **SM2** | Enables the Automatic Address Recognition feature in Modes 2 or 3. If SM2 = 1 then RI will not be set unless the received 9th data bit (RB8) is 1, indicating an address, and the received byte is a Given or Broadcast Address. In Mode 1, if SM2 = 1 then RI will not be activated unless a valid stop bit was received, and the received byte is a Given or Broadcast Address. In Mode 0, SM2 should be 0. |
| **REN** | Enables serial reception. Set by software to enable reception. Clear by software to disable reception. |
| **TB8** | The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired. |
| **RB8** | In modes 2 and 3, the 9th data bit that was received. In Mode 1, if SM2 = 0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used. |
| **TI** | Transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software. |
| **RI** | Receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software. |

**NOTE:**
*SMOD0/1 is located at PCON.6, PCON.7
**$f_{OSC}$ = oscillator frequency

SU00766

**Figure 19. SCON: Serial Port Control Register**

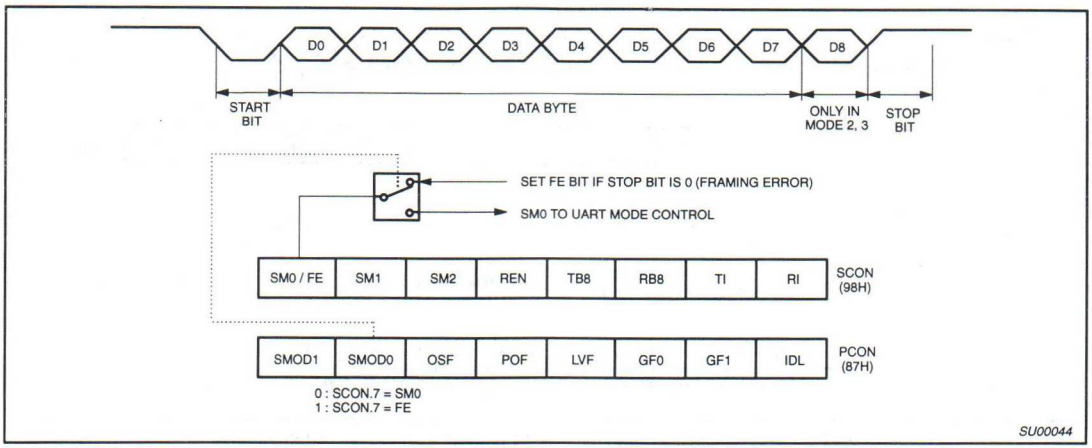



**Figure 20. UART Framing Error Detection**

Using the Automatic Address Recognition feature allows a master to selectively communicate with one or more slaves by invoking the Given slave address or addresses. All of the slaves may be contacted by using the Broadcast address. Two special Function Registers are used to define the slave's address, SADDR, and the address mask, SADEN. SADEN is used to define which bits in the SADDR are to be used and which bits are "don't care". The SADEN mask can be logically ANDed with the SADDR to create the "Given" address which the master will use for addressing each of the slaves. Use of the Given address allows multiple slaves to be recognized while excluding others. The following examples will help to show the versatility of this scheme:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1101 |
| | Given | = | 1100 00X0 |
| Slave 1 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1110 |
| | Given | = | 1100 000X |

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires a 0 in bit 0 and it ignores bit 1. Slave 1 requires a 0 in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010 since slave 1 requires a 0 in bit 1. A unique address for slave 1 would be 1100 0001 since a 1 in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 1100 0000.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

| | | | |
|---|---|---|---|
| Slave 0 | SADDR | = | 1100 0000 |
| | SADEN | = | 1111 1001 |
| | Given | = | 1100 0XX0 |
| Slave 1 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1010 |
| | Given | = | 1110 0X0X |
| Slave 2 | SADDR | = | 1110 0000 |
| | SADEN | = | 1111 1100 |
| | Given | = | 1110 00XX |

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 and 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary t make bit 2 = 1 to exclude slave 2.

The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are treated as don't-cares. In most cases, interpreting the don't-cares as ones, the broadcast address will be FF hexadecimal.

Upon reset SADDR (SFR address 0A9H) and SADEN (SFR address 0B9H) are loaded with 0s. This produces a given address of all "don't cares" as well as a Broadcast address of all "don't cares". this effectively disables the Automatic Addressing mode and allows the microcontroller to use standard 80C51 type UART drivers which do not make use of this feature.

## Analog Comparators

Four analog comparators are provided on chip. Three comparators have a common negative reference CMPR- and independent positive inputs CMP1+, CMP2+, CMP3+ on port 3. The fourth comparator has independent positive and negative inputs CMP0+ and CMP0- on port 2. The CMP register contains an output and enable bit for each comparator. Figure 22 shows the connection of the comparators.

When the comparator is enabled, the port should be configured by the user as high impedance.

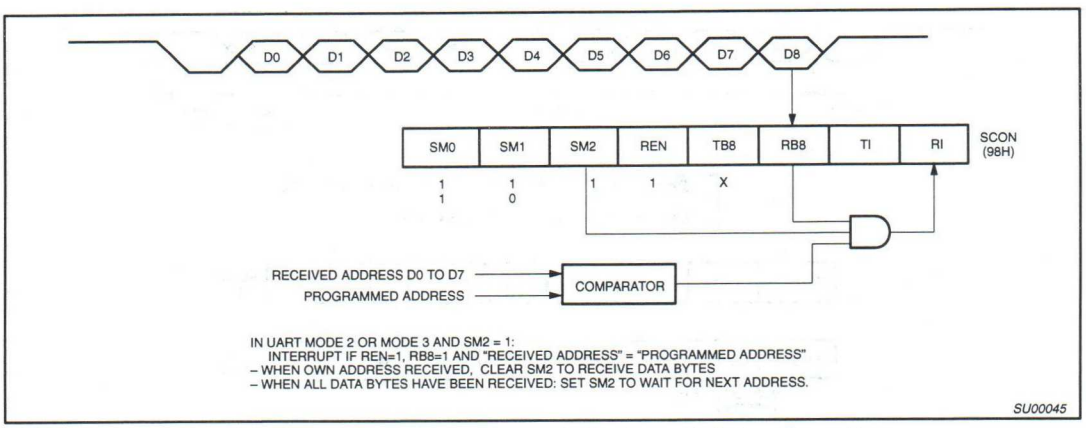



**Figure 21. UART Multiprocessor Communication, Automatic Address Recognition**

**CMP Register Bit Definitions**

CMP.7 enable comparator 3
CMP.6 enable comparator 2
CMP.5 enable comparator 1,
CMP.4 enable comparator 0
CMP.3 comparator 3 output (read only)
CMP.2 comparator 2 output (read only)
CMP.1 comparator 1 output (read only)
CMP.0 comparator 0 output (read only)

All comparators are disabled automatically in power down mode. In idle mode unused comparators should be disabled by software to save power. A comparator can generate an interrupt that will terminate idle mode when used to drive a PCA capture input.

The CMPE register contains bits to enable each comparator to drive external output pins or internal PCA capture inputs. When the comparator is configured for external output, the user must also configure the output port in one of its output modes. The comparator output is wire-ORed with the corresponding port SFR bit, so the SFR bit must also be set by software to enable the output.

**CMPE Register Bit Definitions**

CMPE.7 enables comparator 3 to drive CEX3
CMPE.6 enables comparator 2 to drive CEX2
CMPE.5 enables comparator 1 to drive CEX1
CMPE.4 enables comparator 0 to drive CEX0
CMPE.3 enables comparator 3 output on P2.3
CMPE.2 enables comparator 2 output on P2.2
CMPE.1 enables comparator 1 output on P2.1
CMPE.0 enables comparator 0 output on P2.0

When 1s are written to CMPE bits 7-4, the comparator outputs will drive the corresponding capture input. When 1s are written to CMPE bits 3-0 the comparator output will also drive the corresponding port 2 pin. If the comparator's enabled to drive the capture input but not the port pin, then the port pin can be used for general purpose I/O. When a comparator output is enabled, the user will need to configure the port for one of its output modes.

There are two special function registers associated with the comparators. They are CMP which contains the comparator enables and a bit that can be read by software to determine the state of each comparator's output, and CMPE which controls whether the output from each comparator drives the associated output pin or a capture input associated with one of the PCA modules.

The CMP registers bits 0–3 can be read by software to determine the state of the output of each comparator. To do this the associated comparator must be enabled but the output in port 2 can be disabled. This allows easy polling of the comparator output value without the need to use up a port pin.

The CMPE register allows the comparator to drive the associated PCA module capture input, so that on compare a capture can be generated in the PCA. Bits 0–3 of this register enable the comparator output to drive the associated port 2 output circuitry. Used as a comparator output, the output mode for this port must be configured for output by the user and the port output SFR bit latch must be set. If the comparator is not enabled to drive the port 2 circuitry, the associated port 2 pin can be used for other I/O. This includes when a comparator is enabled to drive the capture input to a PCA module.

## Reduced EMI Mode

There are two bits in the AUXR register that can be set to reduce the internal clock drive and disable the ALE output. AO (AUXR.0) when set turns off the ALE output. LO (AUXR.1) when set reduces the drive of the internal clock circuitry. Both bits are cleared on Reset. With LO set the 8XC576 will still operate at 12MHz, and will have reduced EMI in the range above 100MHz.

## 8XC576 Reduced EMI Mode

**AUXR (0X8E)**

| — | — | — | — | RST | TXI | LO | AO |
|---|---|---|---|---|---|---|---|

AO: Turns off ALE output.
LO: Reduces drive of internal clock circuitry. 8XC576 spec'd to 12MHz when LO set.
TXI: Inverts TxD when set.
RST: Software reset.

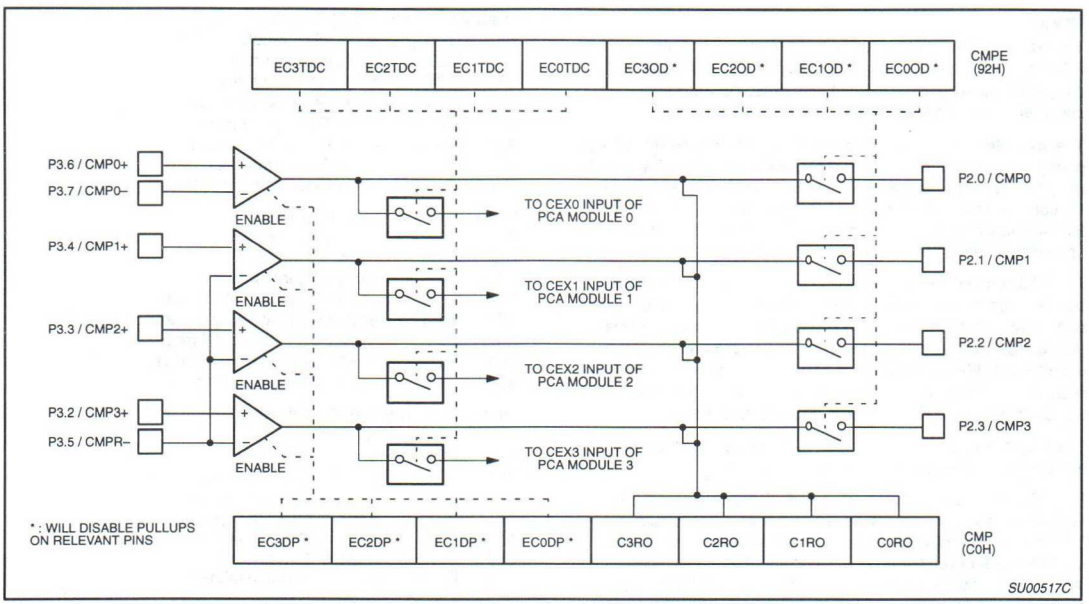



**Figure 22. Analog Comparators**

## INTERNAL RESET

Internal resets (see Figure 1) generated by the power on, low voltage, software (SRST), watchdog and oscillator fail detect circuits are self timed to guarantee proper initialization of the 8XC576. Reset will be held approximately 24 oscillator periods after normal conditions are detected by all enabled detect circuits. Internal resets do not drive $\overline{RST}$ but will cause missing pulses on ALE.

## Analog to Digital Converter

The 8XC576 has a 6 channel10 bit successive approximation A/D converter with separate result registers for each channel. Operating modes are provided for single or multiple channel conversions and multiple conversions of a single channel without software intervention. The ADC can also be operated in 8 bit mode with faster conversion times. Registers ADC0H–ADC5H contain the MSBs and ADC0L–ADC5L bits 6 and 7 contain the 2 LSBs of the conversion result for each channel. The ADCS register determines which channels are converted in multiple channel modes. If the ADCS bit corresponding to a channel is set, that channel is converted, else if the bit is clear the channel is skipped.

**A/D Channel Select (ADCS) Register (Reset Value = 00H)**

ADCS5 ADCS.5 – A/D channel 5 select bit
ADCS4 ADCS.4 – A/D channel 4 select bit
ADCS3 ADCS.3 – A/D channel 3 select bit
ADCS2 ADCS.2 – A/D channel 2 select bit
ADCS1 ADCS.1 – A/D channel 1 select bit
ADCS0 ADCS.0 – A/D channel 0 select bit

**A/D Control (ADCON) Register (Reset Value = 00H)**

ADF ADCON.7 – A/D conversion complete flag
ADCE ADCON.6 – A/D conversion enable
AD8M ADCON.5 – A/D 8-bit mode
AMOD1 ADCON.4 – A/D mode select bit 1
AMOD0 ADCON.3 – A/D mode select bit 0
ASCA2 ADCON.2 – A/D channel address bit 2
ASCA1 ADCON.1 – A/D channel address bit 1
ASCA0 ADCON.0 – A/D channel address bit 0

| AMOD1 | AMOD0 | |
|---|---|---|
| 0 | 0 | Single Conversion Mode – channel selected by bits ASCA2..0 in ADCON is converted, the result placed in the associated result registers; ADF is set on completion. |
| 0 | 1 | Mulitple Channel Scan Mode – all channels selected in the ADCS register are converted starting with the channel addressed by bits ASCA2..0 in ADON, conversion results are placed in the corresponding result registers for each channel. ADF is set when the last conversion is completed. |
| 1 | 0 | Single Channel Multiple Conversion – channel selected by bits ASCA2..0 in ADCON is converted 6 times and all 6 results are saved in ADC0H–ADC5H and ADC0L–ADC5L, ADF is set when all conversions are complete. |
| 1 | 1 | Multiple Channel Continuous – same as Multiple Channel Scan mode but repeats as long as ADCE=1, ADF is set when all channels have been converted once. Hardware will prevent the ADC from wiriting to the result registers while they are being read. |

Flag ADF is set upon completion of a conversion, if the ADC interrupt enable bit EAD is set, the program will vector to the ADC interrupt location when ADF is set.

## PWMs

The pulse width modulator system of the 8XC576 contains two PWM output channels. These channels generate pulses of programmable length and interval. The prescaler and counter are common to both PWM channels.

The prescaler is loaded with the complement of the PWMP register during counter overflow, internal reset, and when EN/CLR# = 0. The repetition frequency is defined by the 8-bit prescaler which clocks the counter. The prescaler division factor = PWMP+1. Reading the PWMP gives the current reload value. The actual count of the prescaler cannot be read.

The 8-bit counter counts from 0–254 inclusive. The value of the counter is compared to the contents of the compare registers PWM0 and PWM1. When the counter compares to the compare register, that register's output goes LOW. When the counter reaches zero the output is set HIGH unless PWMn = 00H. The duty cycle of each channel is defined by the contents of its compare register and is in the range of 0 to 1, programmed in increments of 1/255.

The outputs can be set continuously low by loading PWMn with 00H and continuously high by loading with FFH.

The PWM counter is enabled with bit EN/CLR# of the PWCON register. Output to the port pin is separately enabled by setting the PWEn bits in the PWCON register. The counter remains active if EN/CLR# is set even if both PWEn bits are reset. The PWM function is reset by a chip reset. In idle mode, the PWM will function as configured by PWCON. In power-down the state of the PWM will freeze when the internal clock stops. If the chip is awakened with an external interrupt, the PWM will continue to function from its state when power-down was entered. The EN/CLR# bit of PWCON will clear the counter and load the contents of the PWMP into the prescaler when set LOW. If PWEn is set at this time the output will go HIGH unless PWMn is 00H.

The repetition frequency is given by:

$$f_{PWM} = \frac{f_{OSC}}{(510 \times (1 + PWMP))}$$

An oscillator frequency of 12MHz results in a repetition range of 92Hz to 23.5KHz.

The high/low ratio of PWMn is PWMn/(255–PWMn) for PWMn values except 255. A PWMn value of 255 results in a high PWMn output.

In order for the PWMn output to be used as a standard I/O pin, PWMn must be reset. The PWM counter can still be used as an internal timer by setting EN/CLR#.

**Pulse Width Modulator Control Register Bit Definitions (PWCON = BCH)**

| | | |
|---|---|---|
| PWMF | PWCON.3 | Counter overflow flag, must be cleared by software |
| EN/$\overline{CLR}$ | PWCON.2 | Counter enable and counter/prescaler reset when Low |
| PWE1 | PWCON.1 | PWM1 output to P2.7 pin enable |
| PWE0 | PWCON.0 | PWM0 output to P2.6 pin enable |

**Auxiliary Register Bit Definitions (AUXR =8EH)**

| | | |
|---|---|---|
| RST | AUXR.3 | Software reset bit |
| TXI | AUXR.2 | SIO TxD invert |
| LO | AUXR.1 | Low Speed, reduces internal clock drive |
| AO | AUXR.0 | ALE Off, when set turns off ALE |

**Interrupt Enable 0 (IE0) Register**

| | | |
|---|---|---|
| EA | IE0.7 | Enable all interrupts |
| EC | IE0.6 | Enable PCA interrupt |
| ET2 | IE0.5 | Enable Timer 2 interrupt |
| ES | IE0.4 | Enable Serial I/O interrupt |
| ET1 | IE0.3 | Enable Timer 1 interrupt |
| EX1 | IE0.2 | Enable External interrupt 1 |
| ET0 | IE0.1 | Enable Timer 0 interrupt |
| EX0 | IE0.0 | Enable External interrupt 0 |

**Interrupt Enable 1 (IE1) Register**

| | | |
|---|---|---|
| EOB | IE1.7 | Enable OBE interrupt |
| EIB | IE1.6 | Enable IBF interrupt |
| EAD | IE1.5 | Enable ADC interrupt |
| EC4 | IE1.4 | Enable PCA module 4 interrupt |
| EC3 | IE1.3 | Enable PCA module 3 interrupt |
| EC2 | IE1.2 | Enable PCA module 2 interrupt |
| EC1 | IE1.1 | Enable PCA module 1 interrupt |
| EC0 | IE1.0 | Enable PCA module 0 interrupt |

**Interrupt Priority 0 (IP0) Register**

| | | |
|---|---|---|
| | IP0.7 | (reserved) |
| PPC | IP0.6 | PCA interrupt priority |
| PT2 | IP0.5 | Timer 2 interrupt priority |
| PS | IP0.4 | Serial I/O interrupt priority |
| PT1 | IP0.3 | Timer 1 interrupt priority |
| PX1 | IP0.2 | External interrupt 1 priority |
| PT0 | IP0.1 | Timer 0 interrupt priority |
| PX0 | IP0.0 | External interrupt 0 priority |

**Interrupt Priority 1 (IP1) Register**

| | | |
|---|---|---|
| POB | IP1.7 | OBE interrupt priority |
| PIB | IP1.6 | IBF interrupt priority |
| PAD | IP1.5 | ADC interrupt priority |
| PC4 | IP1.4 | PCA module 4 interrupt priority |
| PC3 | IP1.3 | PCA module 3 interrupt priority |
| PC2 | IP1.2 | PCA module 2 interrupt priority |
| PC1 | IP1.1 | PCA module 1 interrupt priority |
| PC0 | IP1.0 | PCA module 0 interrupt priority |

| Priority | Source | Flag | Vector | |
|---|---|---|---|---|
| 1 | INT0 | IE0 | 03H | highest priority |
| 2 | ADC | ADF | 3BH | |
| 3 | TIMER 0 | TF0 | 0BH | |
| 4 | INT1 | IE1 | 13H | |
| 5 | TIMER 1 | TF1 | 1BH | |
| 6 | SERIAL | RI,TI | 23H | |
| 7 | PCA0 | CC0 | 43H | |
| 8 | PCA1 | CC1 | 4BH | |
| 9 | PCA2 | CC2 | 53H | |
| 10 | PCA3 | CC3 | 5BH | |
| 11 | PCA4 | CC4 | 63H | |
| 12 | PCA | ECF | 33H | |
| 13 | TIMER 2 | TF2/EXF2 | 2BH | |
| 14 | UPI | IBF | 6BH | |
| 15 | UPI | OBE | 73H | lowest priority |

## Power Control (PCON) Register

| | | |
|---|---|---|
| SMOD1 | PCON.7 | double baud rate bit |
| SMOD0 | PCON.6 | SCON.7 access control |
| OSF | PCON.5 | oscillator fail flag |
| POF | PCON.4 | power off flag |
| LVF | PCON.3 | low voltage flag |
| WDTOF | PCON.2 | watchdog timeout flag |
| PD | PCON.1 | power down mode bit |
| IDL | PCON.0 | idle mode bit |

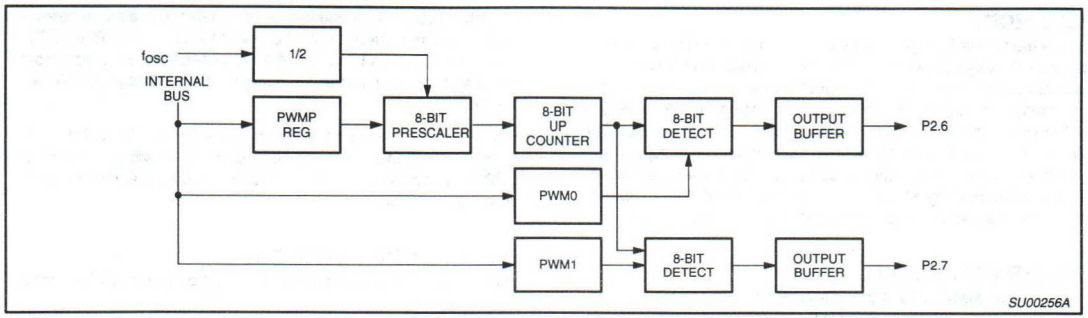



**Figure 23. Block Diagram of PWMs**

## UNIVERSAL PERIPHERAL INTERFACE

UPI mode allows the 8XC576 to function as a slave processor connected to a host CPU bus via port 0. The interface consists of port 0 input and output buffer registers and the UPI control/status register (UCS). UPI mode is enabled by setting the UPI enable bit (UE) in the UCS. When operating in UPI mode, port 0 pins should be programmed to High-Z (P0M1=1 and P0M2=0) by user firmware. Access to port 0 is controlled by inputs $\overline{WR}$, $\overline{RD}$, $\overline{CS}$, and A0. $\overline{RD}$ and $\overline{WR}$ are the external read and write strobes controlled by the host CPU. $\overline{CS}$ is the chip select input, normally a decoded address from the host CPU bus, which qualifies $\overline{RD}$ and $\overline{WR}$ (these pins have no effect when $\overline{CS}$=1). The A0 pin is an address input from the host CPU which selects either the port 0 output buffer or the UCS register to be output during a read operation. During a write operation, the value of the A0 pin is latched in the AF flag in the UCS register. The following is a summary of the UPI data control inputs:

| $\overline{CS}$ | $\overline{RD}$ | $\overline{WR}$ | $\overline{A0}$ | |
|---|---|---|---|---|
| 0 | 0 | 1 | 0 | read port 0 output buffer, clear OBF/set OBE |
| 0 | 0 | 1 | 1 | read UPI control/ status register |
| 0 | 1 | 0 | 0 | write data to input buffer set IBF, clear AF |
| 0 | 1 | 0 | 1 | write command to input buffer set IBF, AF |
| 1 | x | x | x | disable input/output |

**UPI Control Status Register (UCS, Reset value = 00H)**

| | | |
|---|---|---|
| UCS.7 | ST7 | User defined status bit |
| UCS.6 | ST6 | User defined status bit |
| UCS.5 | ST5 | User defined status bit |
| UCS.4 | ST4 | User defined status bit |
| UCS.3 | UE | UPI Enable bit – if UE=1, UPI is enabled (read only AF, IBF, and OBE/OBF), if UE=0, UPI is disabled and port 0 functions normally. |
| UCS.2 | AF | Address Flag – contains status of the A0 (address) pin during the last write. If A0=0, the input buffer should be interpreted as data by the 8XC576 software, if A0=1, the input buffer should be interpreted as a command. |
| USC.1 | IBF | Input Buffer Full flag – set by hardware on trailing (rising) edge of $\overline{WR}$ when $\overline{CS}$=0, cleared by hardware when port 0 SFR is read (by the 8XC576 software). |
| USC.0 | OBE/OBF | Output Buffer Full flag – set by hardware during writes (by 8XC576 software) to the port 0 SFR, set/cleared by hardware on the trailing (rising) edge of $\overline{RD}$ when $\overline{CS}$=0 and A0=0. |

**NOTE:** This bit is defined as OBE (1=empty) when read by the MCU, and, as OBF (—full) when read by the external host.

The IBF and OBF flag bits reflect the status of the input/output buffers. The host CPU writes to the 8XC576 by driving data on the external bus connected to port 0 and strobing the $\overline{WR}$ pin while $\overline{CS}$=0. The $\overline{WR}$ strobe latches port 0 data in the input buffer and sets the IBF flag on the trailing (rising) edge. When the 8XC576 reads from port 0 in UPI mode, it reads from the input buffer and clears the IBF. When the 8XC576 writes to port 0 in UPI mode, it writes to the output buffer which sets the OBF and clears the OBE flag. The host CPU can read the output buffer or the UCS register enabling the port 0 drivers, the OBF flag is cleared and the OBE flag is set when the output buffer is read.

When the UPI is enabled, the AF, IBF, and OBE/OBF flags are read-only, and thus can only be modified by specific hardware events.

The UPI runs in idle mode. It can interrupt the part out of Idle mode for all UPI write and data read operations. It will not interrupt out of idle mode for a UCS register read operation.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the Logic Symbol, page 3-758.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset. Also see UPI section.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. The control bits for the reduced power modes are in the special function register PCON. Power-down mode can be terminated with either a hardware reset or external interrupt. With an external interrupt $\overline{INT0}$ or $\overline{INT1}$ must be enabled and configured as level sensitive. Holding the pin low restarts to oscillator and bringing the pin back high completes the exit.

Power-down mode can be disabled by the DPD bit in the WDCON register. Reset and waking up from power-down will also enable the DPD bit, therefore, the DPD bit must be cleared again before the power-down mode.

## DESIGN CONSIDERATIONS

At power-on, the voltage on $V_{CC}$ must come up with $\overline{RST}$ low for a proper start-up.

Table 2 shows the state of I/O ports during low current operating modes.

**Table 2. External Pin Status During Idle and Power-Down Modes**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

## ROM CODE SUBMISSION

When submitting ROM code for the 83C576, the following must be specified:

1. 8k byte user ROM data
2. 32 byte ROM encryption key
3. ROM security bits
4. The watchdog timer parameters. (See Watchdog Timer Specifications for definition of WDL and WDCON bits.)

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 1FFFH | DATA | 7:0 | User ROM Data |
| 2000H to 201FH | KEY | 7:0 | ROM Encryption Key<br>FFH = no encryption |
| 2020H | SEC | 0 | ROM Security Bit 1<br>0 = enable security<br>1 = disable security |
| | | 1 | ROM Security Bit 2<br>0 = enable security<br>1 = disable security |
| 2030H | WDCON | 7:5 | PRE2:0 |
| | | 4 | LVRE |
| | | 3 | OFRE |
| | | 2 | DPD |
| | | 1 | WDRUN = 0, not ROM coded |
| | | 0 | WDMOD |
| 2031H | WDL | 7:0 | Watchdog autoload value<br>(see specification) |

**Security Bit 1:** When programmed, this bit has two effects on masked ROM parts:

1. External MOVC is disabled, and
2. $\overline{EA}$ is latched on Reset.

**Security Bit 2:** When programmed, this bit inhibits Verify User ROM.

If the ROM code file does not include the options, the following information must be included with the ROM code.

For each of the following, check the appropriate box and send to Philips along with the code:

Security Bit #1: ☐ Enabled ☐ Disabled

Security Bit #2: ☐ Enabled ☐ Disabled

Encryption: ☐ No ☐ Yes If Yes, must send key file.

Watchdog/Timer Modes: ☐ Watchdog Mode ☐ Timer Mode

Prescaler Value: ☐ Value (Value = 64, 128, 256, 512, 1024, 2048, 4096, 8192)

Autoload Value (range 0–255): ☐

Low Voltage Reset (Value 0 or 1): ☐

Oscillator Fail Reset (Value 0 or 1): ☐

Power-Down (Value 0 or 1): ☐

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Operating temperature under bias | –55 to +125 | °C |
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Maximum $I_{OL}$ per I/O pin | 15 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.5 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, –40°C to +85°C, and –40°C to +125°C; $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $V_{IL}$ | Input low voltage (except Port 1, $\overline{EA}$) | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IL1}$ | Input low voltage ($\overline{EA}$) | | –0.5 | | $0.2V_{CC}$–0.45 | V |
| $V_{IL2}$ | Input low voltage (Port 1) | | –0.5 | | $0.3V_{CC}$ | V |
| $V_{IH}$ | Input high voltage (except Port 1, XTAL1, RST) | $I_{IH}$ < 2mA | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage (XTAL1, RST, Port 1) | $I_{IH}$ < 2mA | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| HYS | Hysteresis voltage (Port 1) | | 200 | | | mV |
| $V_{OL}$ | Output voltage low (Ports 1, 2, 3) | $I_{OL}$ = 1.6mA | | | 0.45 | V |
| $V_{OL1}$ | Output voltage low (Ports 0, ALE, $\overline{PSEN}$) | $I_{OL}$ = 3.2mA | | | 0.45 | V |
| $V_{OH}$ | Output voltage high (Ports 1, 2, 3 in push-pull mode) | $I_{OH}$ = –1.6mA | $V_{CC}$–1.0 | | | V |
| $V_{OH1}$ | Output voltage high (Port 0, ALE, $\overline{PSEN}$) | $I_{OH}$ = –3.2mA | $V_{CC}$–0.7 | | | V |
| $V_{OH2}$ | Output voltage high in weak pullup mode (Port 0, 2, 3) | $I_{OH}$ = –10µA | $V_{CC}$–1.0 | | | V |
| $V_{IO}$ | Offset voltage comparator inputs | | –35 | | +35 | mV |
| $V_{CR}$ | Common mode range comparator inputs | | 0 | | $V_{CC}$ | V |
| $I_{IL}$ | Logical 0 input current (Ports 0, 2, 3) (weak pull-up) | $V_{IN}$ = 0.45V | | | –250 | µA |
| $I_{IH}$ | Input pulldown current (Port 0, Port2 in open drain mode) | 0.45 < $V_{IN}$ < $V_{CC}$ | 2 | | 40 | µA |
| $I_{L2}$ | Input leakage current ($\overline{EA}$, P0. 2. 3 High-Z) | 0.45 < $V_{IN}$ < $V_{CC}$ | –10 | | +10 | µA |
| $I_{LA}$ | Input leakage current comparator/ADC inputs | 0 < $V_{IN}$ < $V_{CC}$ | –1.0 | | +1.0 | µA |
| $I_{CC}$ | Power supply current:[7]<br>Active mode @ 16MHz[5]<br>Idle mode @ 16MHz<br>Power-down mode | See note 6 | | <br>20<br>8<br>5 | <br>30<br>12<br>75 | <br>mA<br>mA<br>µA |
| $R_{RST}$ | Internal reset pull-up resistor | $V_{IN}$ = 0V | 50 | | 200 | kΩ |
| $V_{LOW}$ | Low $V_{CC}$ detect voltage | | 3.75 | | 4.25 | V |
| $C_{IO}$ | Pin capacitance[9] | f = 1MHz | | | 15 | pF |

**NOTES:**

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
3. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is between $V_{IH}$ and $V_{IL}$.
5. $I_{CC}$MAX at other frequencies can be determined from Figure 33.
6. See Figures 34 through 37 for $I_{CC}$ test conditions.
7. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
8. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 10mA
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 71mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
9. 20pF MAX for CERDIP package; 15pF MAX for all other packages.

## A/D CONVERTER DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, –40°C to +85°C, and –40°C to +125°C; $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| **Static Characteristics** | | | | | |
| R | Resolution | Monotonic with no missing codes | 10 | | Bits |
| $IL_e$ | Integral non-linearity error [2, 5, 8] | | | ±2 | LSB |
| $DL_e$ | Differential non-linearity error[2, 3, 4, 7, 8] | | | ±1 | LSB |
| $FS_e$ | Full Scale error [2, 8] | | | ±3 | LSB |
| $OS_e$ | Offset error [2, 6, 8] | | | ±2 | LSB |
| **Dynamic Characteristics** | | | | | |
| $t_{ADC}$ | Conversion time (including sampling time) | | | $48t_{CY}$ | µs |
| $t_{ADS}$ | Sampling tme | | | $8t_{CY}$ | µs |
| **Analog Input Characteristics** | | | | | |
| $AV_{IN}$ | Analog input voltage | | $AV_{SS}$ – 0.2 | $AV_{DD}$ + 0.2 | V |
| $C_{IA}$ | Analog input capacitance | | | 15 | pF |
| $M_{CTC}$ | Channel-to-channel matching[7] | | | ±1 | LSB |
| $C_t$ | Crosstalk between inputs of port 1[7] | 0–100kHz | | –60 | dB |
| **Power Requirements** | | | | | |
| $AV_{CC}/V_{REF+}$ | Analog supply and reference voltage | $AV_{CC} = V_{CC} \pm 0.2$ | 4.0 | 6.0 | V |
| $AI_{CC}$ | Analog supply current: operating: (16MHz) | $AV_{CC}$ = 6.0V | | 1.2 | mA |

**NOTES:**

1. The following condition must not be exceeded: $V_{DD} - 0.2V < AV_{DD} < V_{DD} + 0.2V$.
2. Conditions: $AV_{SS}$ = 0V; $AV_{CC}$ = 4.997V; $V_{CC}$ = 5.0V.
3. The differential non-linearity ($DL_e$) is the difference between the actual step width and the ideal step width. (See Figure 24).
4. The ADC is monotonic; there are no missing codes.
5. The integral non-linearity ($IL_e$) is the peak difference between the center of the steps of the actual and the ideal transfer curve after appropriate adjustment of gain and offset error. (See Figure 24).
6. The offset error ($OS_e$) is the absolute difference between the straight line which fits the actual transfer curve (after removing gain error), and a straight line which fits the ideal transfer curve. (See Figure 24).
7. Guaranteed by design.
8. To meet Error Specification, analog input voltage must be less than 1V/ms.

$$\text{Slew Rate}_{MAX} = \frac{(AV_{CC}/1023) \times 1000}{4 \times (12/\text{Osc Freq (MHz)})} \quad (V/ms)$$

For 16MHz @ 5.0V slew rate = 1.6V/ms.

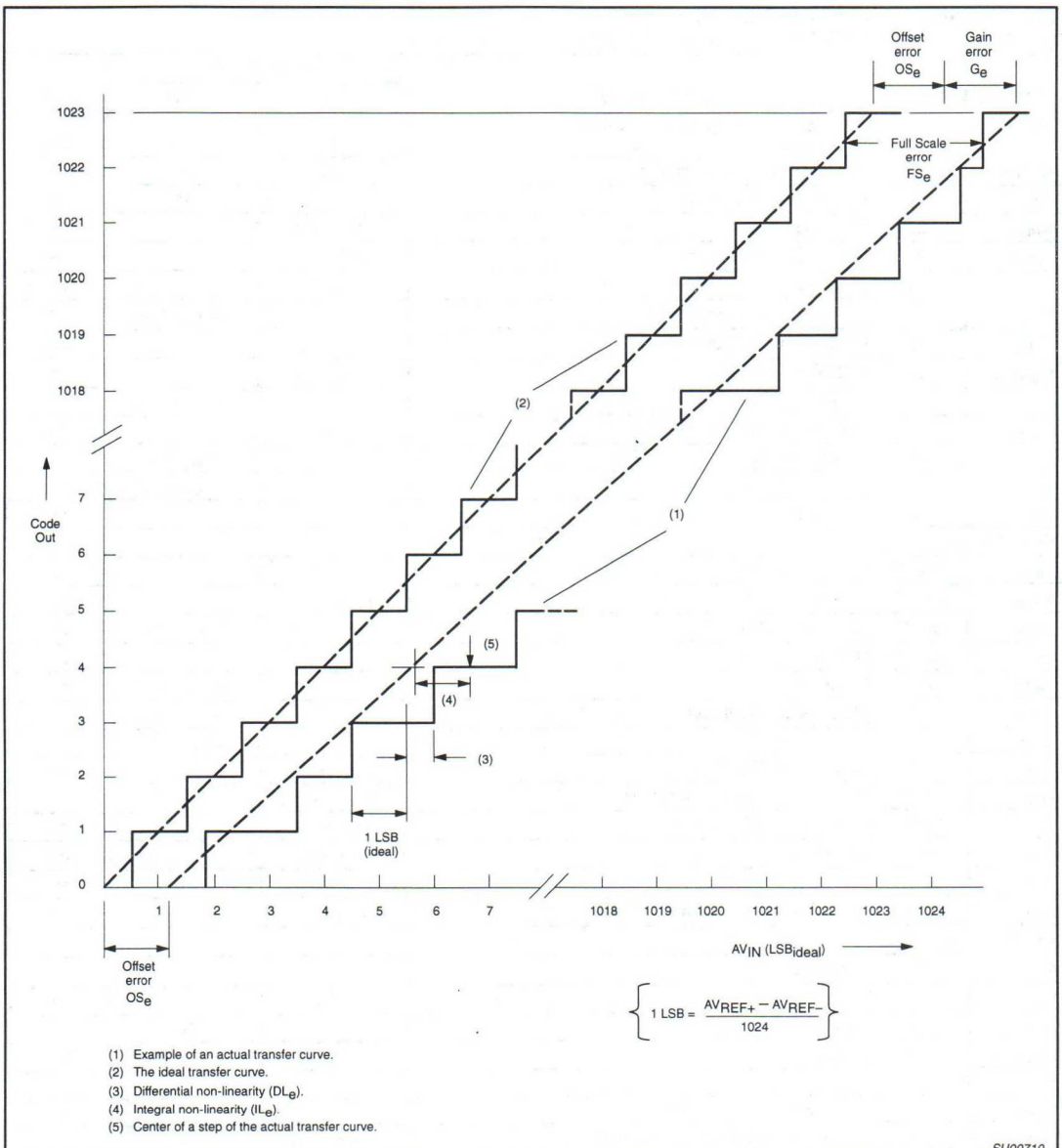



**Figure 24. ADC Conversion Characteristic**

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, –40°C to +85°C, and –40°C to +125°C; $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2]

| SYMBOL | FIGURE | PARAMETER | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $1/t_{CLCL}$ | 25 | Oscillator frequency: **Speed Version** 8XC576 E | 6 | 16 | MHz |
| OSCF | | Oscillator fail detect frequency | 0.6 | 5.5 | MHz |
| TR | | Comparator response time | | 10 | µs |
| $t_{LHLL}$ | 25 | ALE pulse width | $2t_{CLCL}$–40 | | ns |
| $t_{AVLL}$ | 25 | Address valid to ALE low | $t_{CLCL}$–40 | | ns |
| $t_{LLAX}$ | 25 | Address hold after ALE low | $t_{CLCL}$–30 | | ns |
| $t_{LLIV}$ | 25 | ALE low to valid instruction in | | $4t_{CLCL}$–100 | ns |
| $t_{LLPL}$ | 25 | ALE low to $\overline{PSEN}$ low | $t_{CLCL}$–30 | | ns |
| $t_{PLPH}$ | 25 | $\overline{PSEN}$ pulse width | $3t_{CLCL}$–45 | | ns |
| $t_{PLIV}$ | 25 | $\overline{PSEN}$ low to valid instruction in | | $3t_{CLCL}$–105 | ns |
| $t_{PXIX}$ | 25 | Input instruction hold after $\overline{PSEN}$ | 0 | | ns |
| $t_{PXIZ}$ | 25 | Input instruction float after $\overline{PSEN}$ | | $t_{CLCL}$–25 | ns |
| $t_{AVIV}$ | 25 | Address to valid instruction in | | $5t_{CLCL}$–105 | ns |
| $t_{PLAZ}$ | 25 | $\overline{PSEN}$ low to address float | | 10 | ns |
| **Data Memory** | | | | | |
| $t_{RLRH}$ | 26, 27 | $\overline{RD}$ pulse width | $6t_{CLCL}$–100 | | ns |
| $t_{WLWH}$ | 26, 27 | $\overline{WR}$ pulse width | $6t_{CLCL}$–100 | | ns |
| $t_{RLDV}$ | 26, 27 | $\overline{RD}$ low to valid data in | | $5t_{CLCL}$–165 | ns |
| $t_{RHDX}$ | 26, 27 | Data hold after $\overline{RD}$ | 0 | | ns |
| $t_{RHDZ}$ | 26, 27 | Data float after $\overline{RD}$ | | $2t_{CLCL}$–60 | ns |
| $t_{LLDV}$ | 26, 27 | ALE low to valid data in | | $8t_{CLCL}$–150 | ns |
| $t_{AVDV}$ | 26, 27 | Address to valid data in | | $9t_{CLCL}$–165 | ns |
| $t_{LLWL}$ | 26, 27 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | $3t_{CLCL}$–50 | $3t_{CLCL}$+50 | ns |
| $t_{AVWL}$ | 26, 27 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | $4t_{CLCL}$–130 | | ns |
| $t_{QVWX}$ | 26, 27 | Data valid to $\overline{WR}$ transition | $t_{CLCL}$–50 | | ns |
| $t_{WHQX}$ | 26, 27 | Data hold after $\overline{WR}$ | $t_{CLCL}$–50 | | ns |
| $t_{RLAZ}$ | 26, 27 | $\overline{RD}$ low to address float | | 0 | ns |
| $t_{WHLH}$ | 26, 27 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | $t_{CLCL}$–40 | $t_{CLCL}$+40 | ns |
| **External Clock** | | | | | |
| $t_{CHCX}$ | 29 | High time | 20 | | ns |
| $t_{CLCX}$ | 29 | Low time | 20 | | ns |
| $t_{CLCH}$ | 29 | Rise time | | 20 | ns |
| $t_{CHCL}$ | 29 | Fall time | | 20 | ns |
| **Shift Register** | | | | | |
| $t_{XLXL}$ | 28 | Serial port clock cycle time | $12t_{CLCL}$ | | ns |
| $t_{QVXH}$ | 28 | Output data setup to clock rising edge | $10t_{CLCL}$–133 | | ns |
| $t_{XHQX}$ | 28 | Output data hold after clock rising edge | $2t_{CLCL}$–60 | | ns |
| $t_{XHDX}$ | 28 | Input data hold after clock rising edge | 0 | | ns |
| $t_{XHDV}$ | 28 | Clock rising edge to input data valid | | $10t_{CLCL}$–133 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Interfacing the 83C576/87C576 to devices with float times up to 45ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.

## UPI AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, –40°C to +85°C, and –40°C to +125°C; $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $t_{AR}$ | $\overline{CS}$, A setup to $\overline{RD}$ | 0 | | ns |
| $t_{RA}$ | $\overline{CS}$, A hold after $\overline{RD}$ | 35 | | ns |
| $t_{RR}$ | $\overline{RD}$ pulse width | 35 | | ns |
| $t_{AD}$ | $\overline{CS}$, A to data out delay | | 45 | ns |
| $t_{RD}$ | $\overline{RD}$ to data out delay | | 35 | ns |
| $t_{DF}$ | $\overline{RD}$ to data float delay (guaranteed by design) | | 30 | ns |
| $t_{AW}$ | $\overline{CS}$, A setup to $\overline{WR}$ | 0 | | ns |
| $t_{WA}$ | $\overline{CS}$, A hold after $\overline{WR}$ | 15 | | ns |
| $t_{WW}$ | $\overline{WR}$ pulse width | 45 | | ns |
| $t_{DW}$ | Data setup to $\overline{WR}$ | 5 | | ns |
| $t_{WD}$ | Data hold after $\overline{WR}$ | 25 | | ns |

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{PSEN}$ low.

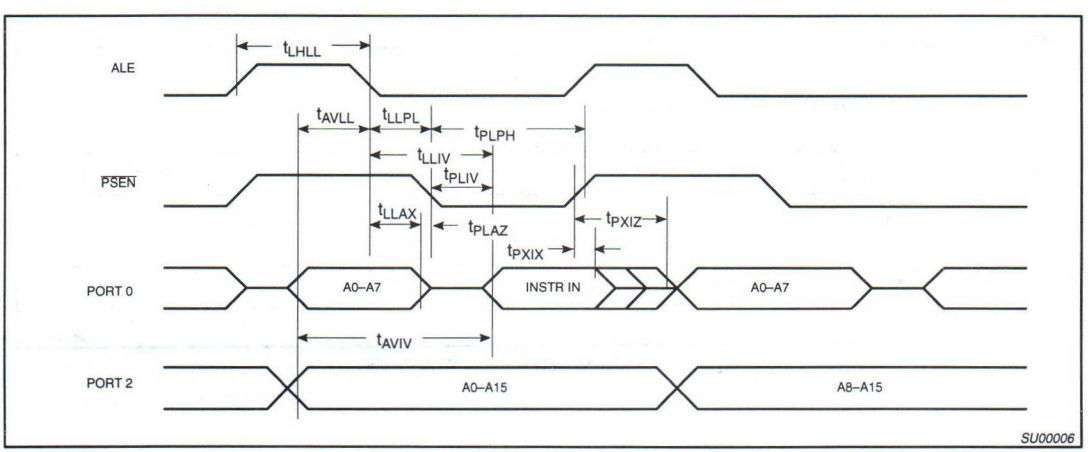



**Figure 25. External Program Memory Read Cycle**

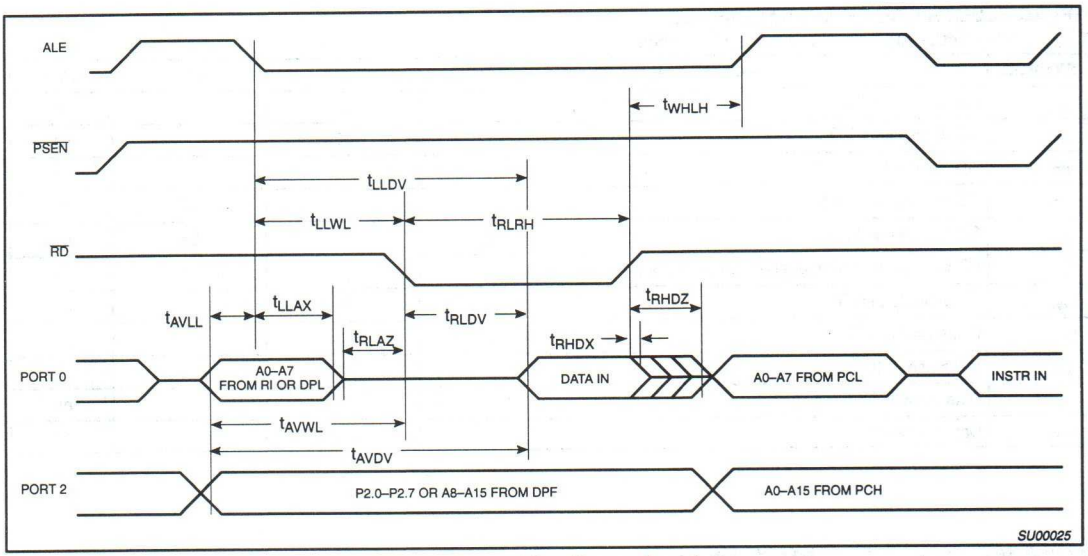



**Figure 26. External Data Memory Read Cycle**

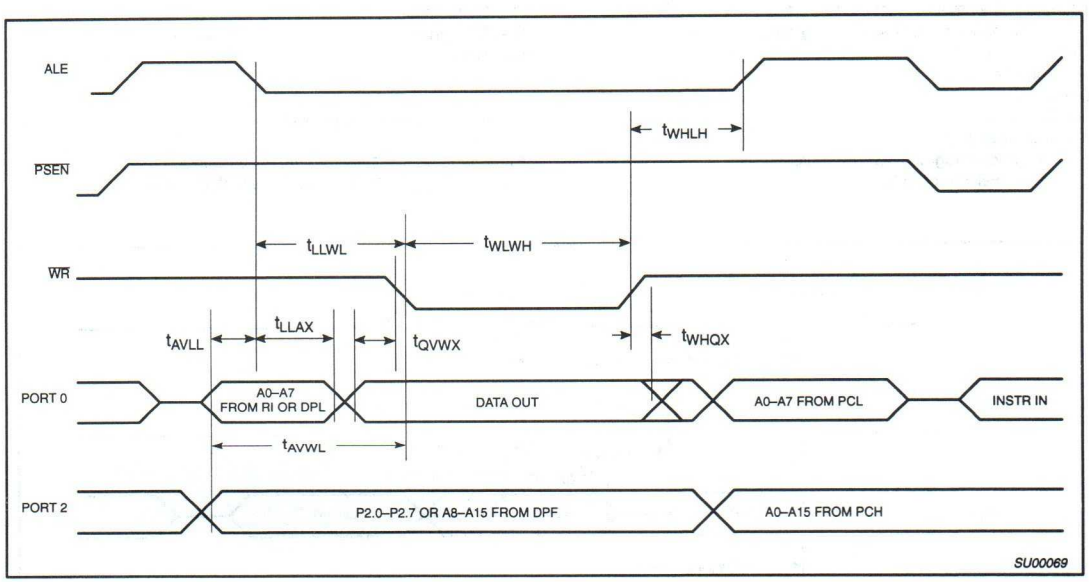



**Figure 27. External Data Memory Write Cycle**

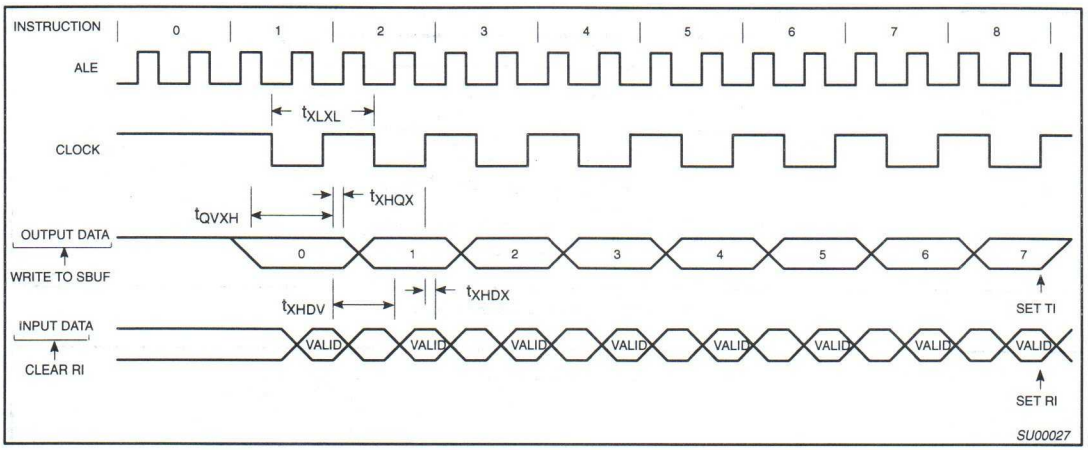



**Figure 28. Shift Register Mode Timing**

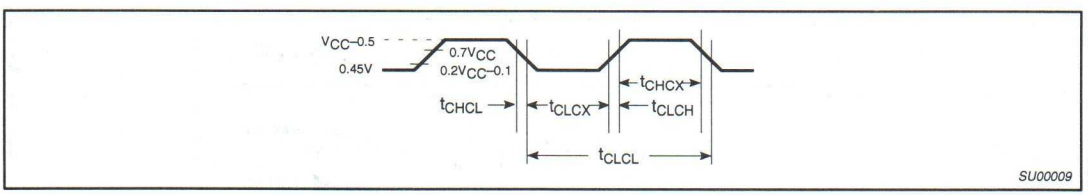



**Figure 29. External Clock Drive**

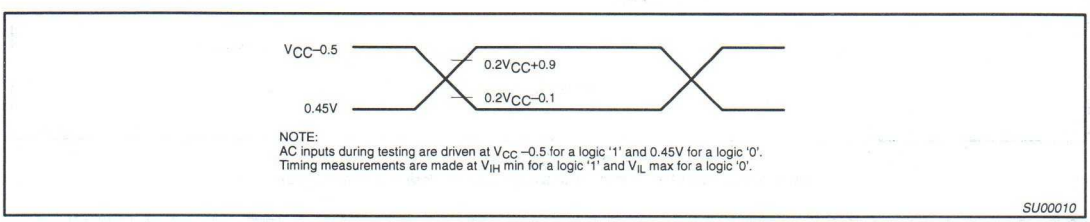



**Figure 30. AC Testing Input/Output**

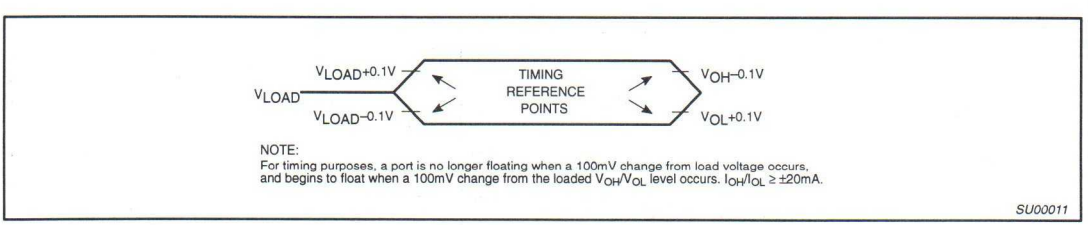



**Figure 31. Float Waveform**

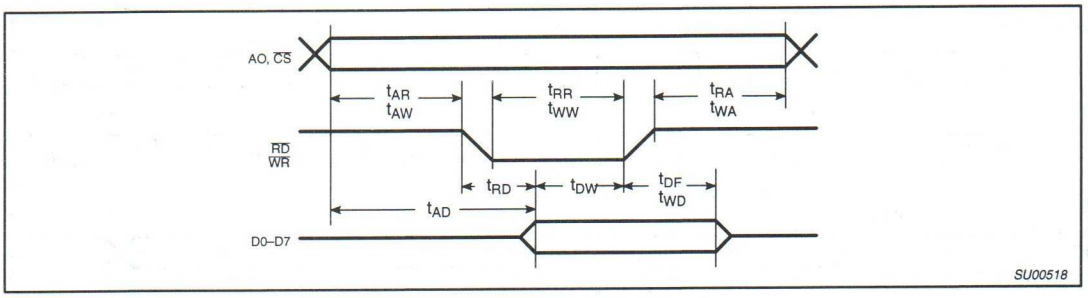



**Figure 32. UPI Read/Write Cycles**

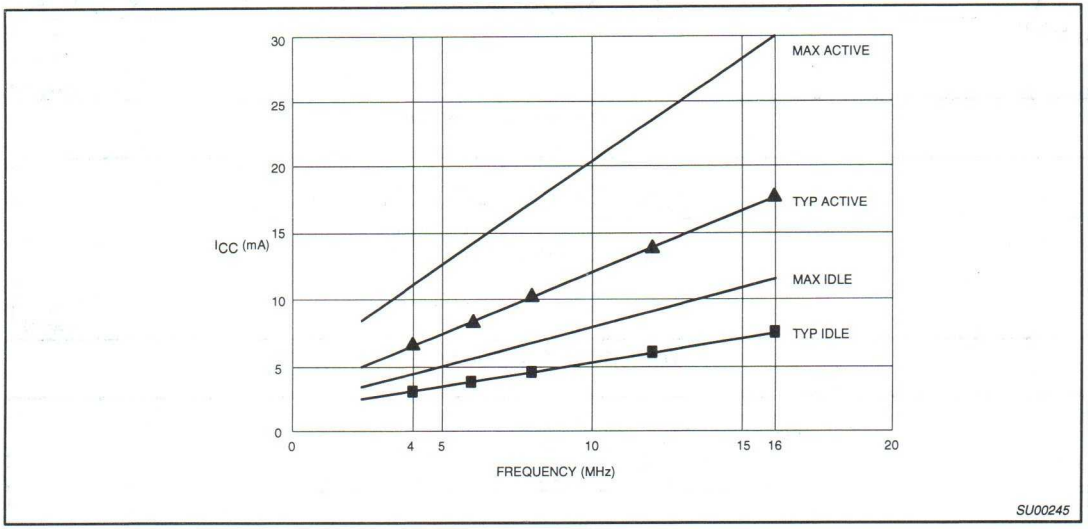



**Figure 33. $I_{CC}$ vs. FREQ**
**Valid only within frequency specifications of the device under test**

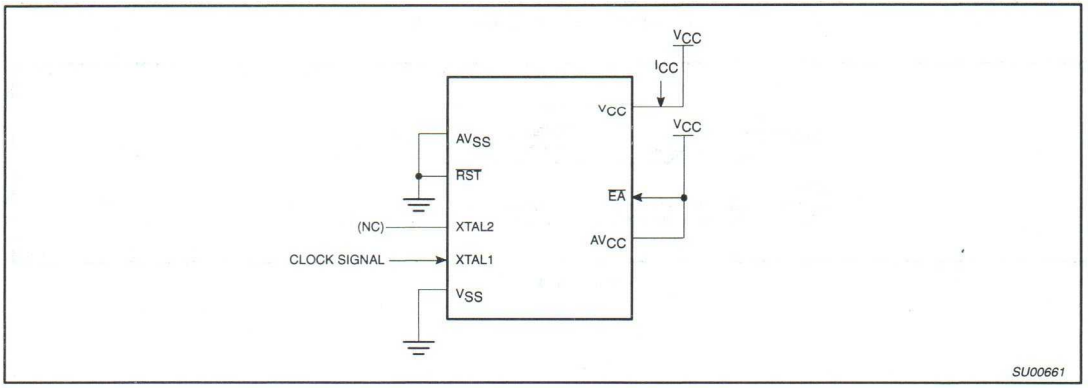



**Figure 34. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

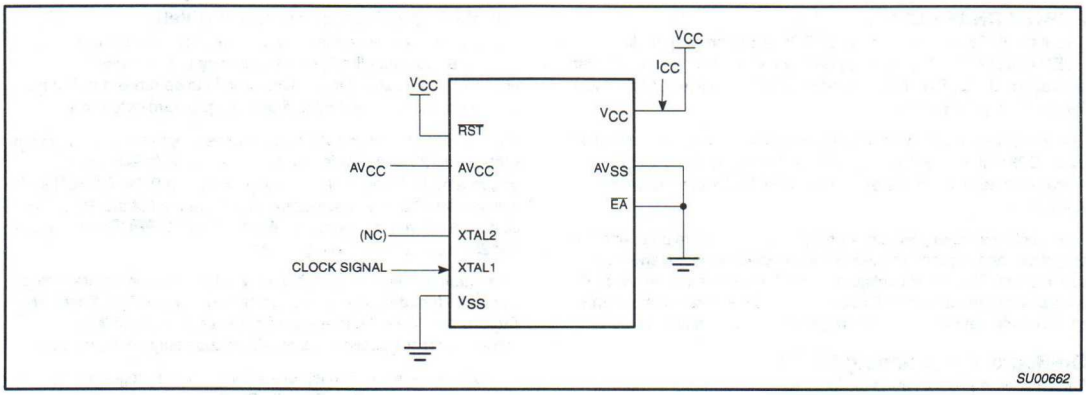



**Figure 35. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

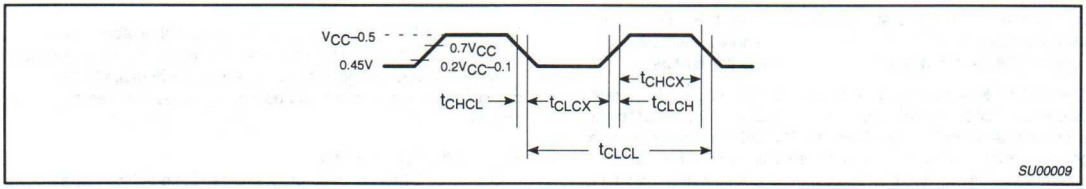



**Figure 36. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

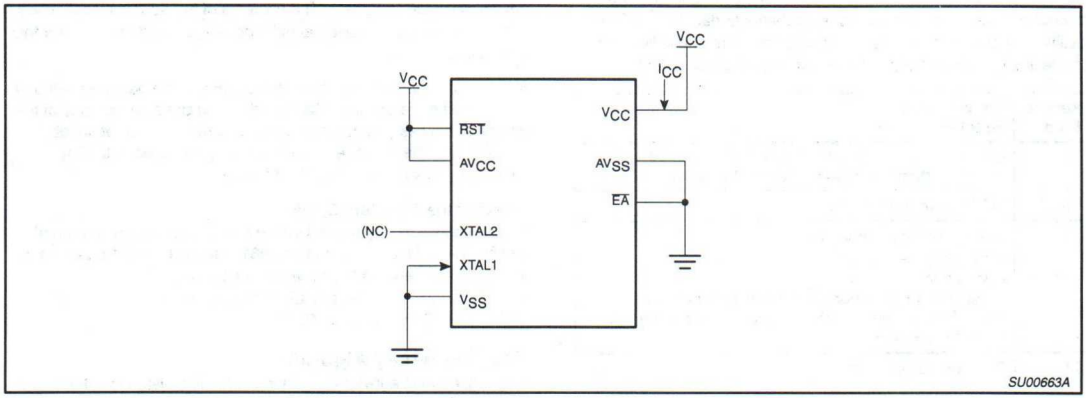



**Figure 37. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2V to 5.5V**

## EPROM CHARACTERISTICS

To put the 87C576 in the parallel EPROM programming mode, PSEN must be held high during power up, then driven low with reset active. The 87C576 is programmed by using a modified Quick-Pulse Programming™ algorithm.

The 87C576 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C576 manufactured by Philips.

Table 3 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the security bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 38 and 39. Figure 40 shows the circuit configuration for normal program memory verification.

## On-Board Programming (OBP)

The On-Board Programming facility consists of a series of internal hardware resources coupled with internal firmware to facilitate remote programming of the 87C576 through the serial port.

The OBP function is invoked by having the $\overline{EA}/V_{PP}$ pin at the $V_{PP}$ voltage level at the time that the part exits reset. The OBP function only requires that the TxD, RxD, $V_{SS}$, $V_{CC}$, and $V_{PP}$ pins be connected to an external circuit in order to use this feature.

The OBP feature provides for the use of a wide range of baud rates independent of the oscillator frequency used. It is also adaptable to a wide range of oscillator frequencies. The OBP facility provides for both auto-echo and no-echo of received characters. The OBP feature requires that an initial character, an uppercase U, be sent to the 87C576 to establish the baud rate to be used.

Once baud rate initialization has been performed, the OBP facility only accepts Intel Hex records. The record-type field of these hex records are used to indicate either commands or data for the OBP facility. The maximum number of data bytes in a record is limited to 16 (decimal). These commands/data are summarized below:

| Record Type | Command/Data Function |
|---|---|
| 00 | Data record, programs the part with data indicated in record starting with load address in the record |
| 01 | EOF record, no operation |
| 02 | Specify timing parameters<br>– rec length = 3 bytes<br>– load address = 0000<br>– 1st byte = timer count for 50µs programming pulse<br>– 2nd byte = timer count for 10µs delay between pulses<br>– 3rd byte = 0AH |
| 03 | Program security bits<br>– rec length = 1 byte<br>– load address = 0000<br>– 1st byte = sec bit values (xxxx xxB2B1) |
| 04 | Display contents of USER EPROM array<br>– rec length = 00<br>– load address = 0000 |
| 05 | Verify security bit status<br>– rec length = 00<br>– load address = 0000 |

## Quick-Pulse Programming (Parallel)

The setup for microcontroller quick-pulse programming is shown in Figure 38. Note that the 87C576 is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 3 and 2, as shown in Figure 38. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 1 specified in Table 3 are held at the 'Program Code Data' levels indicated in Table 3. The ALE/$\overline{PROG}$ is pulsed low 25 times as shown in Figure 39.

To program the encryption table, repeat the 25 pulse programming sequence for addresses 0 through 1FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the security bits, repeat the 25 pulse programming sequence using the 'Pgm Security Bit' levels. After one security bit is programmed, further programming of the code memory and encryption table is disabled. However, the other security bit can still be programmed.

Note that the $\overline{EA}/V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

## Program Verification

If security bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 3 and 2 as shown in Figure 40. The other pins are held at the 'Verify Code Data' levels indicated in Table 3. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

### Reading the Signature Bytes

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P1.0 and P1.1 need to be pulled to a logic low. The values are:
(030H) = 15H indicates manufactured by Philips
(B6H) = B6H indicates 87C576

## Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 3, and which satisfies the timing specifications, is suitable.

™Trademark phrase of Intel Corporation.

## Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-s/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

**Table 3. EPROM Programming Modes**

| MODE | RST | $\overline{PSEN}$ | ALE/$\overline{PROG}$ | $\overline{EA}$/$V_{PP}$ | P2.7 | P2.6 | P1.1 | P1.0 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 0 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 0 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm security bit 1 | 0 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm security bit 2 | 0 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V±10% during programming and verification.

* ALE/$\overline{PROG}$ receives 5 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 50µs (±10µs) and high for a minimum of 10µs.

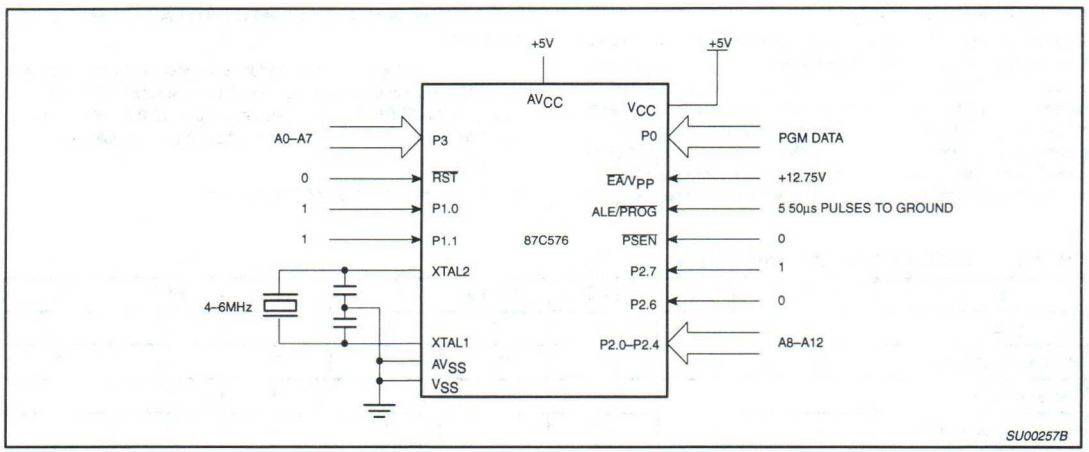



**Figure 38. Programming Configuration**

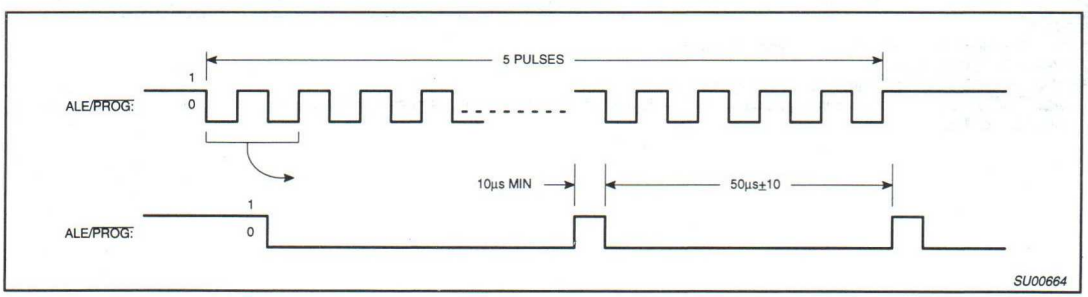



**Figure 39. PROG Waveform**

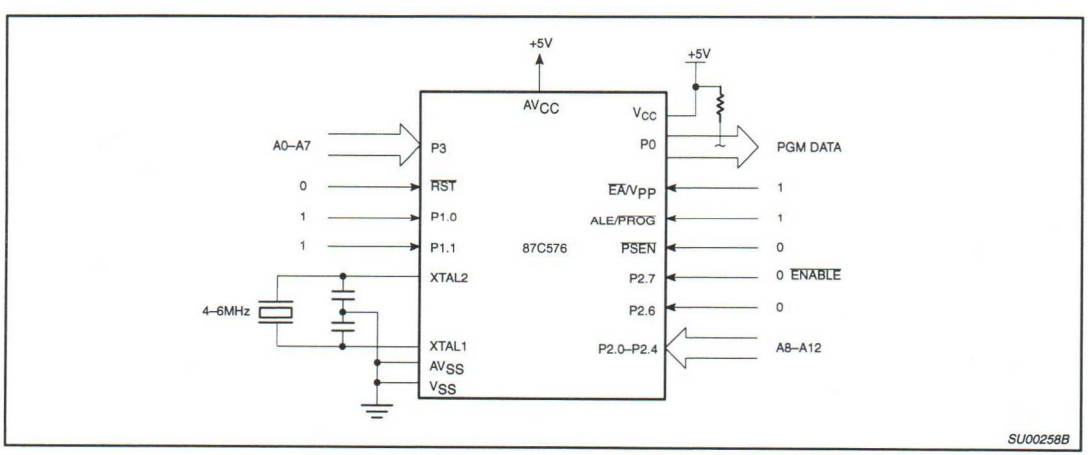
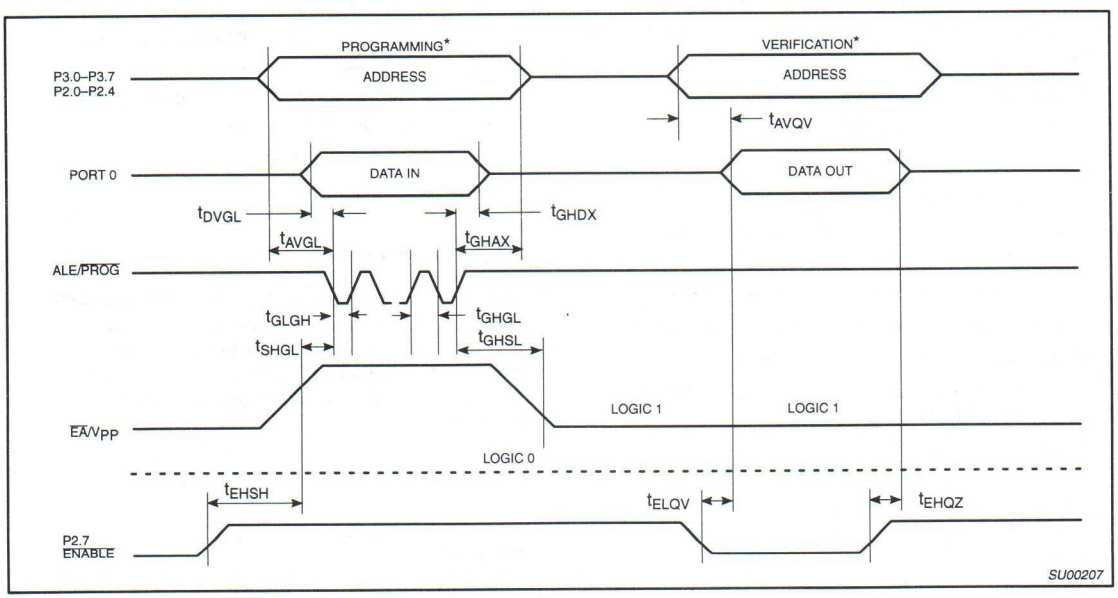



**Figure 40. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (See Figure 41)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 12 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 40 | 60 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




* FOR PROGRAMMING VERIFICATION SEE FIGURE 38.
FOR VERIFICATION CONDITIONS SEE FIGURE 40.

**Figure 41. EPROM Programming and Verification**

# Low voltage 8-bit microcontrollers P80CL580; P83CL580

## 1 FEATURES

- Full static 80C51 Central Processing Unit
- 8-bit CPU, ROM, RAM, I/O in a 56-lead VSO or 64-lead QFP package
- 6 kbytes ROM Program Memory, expandable externally to 64 kbytes (only P83CL580)
- 256 bytes RAM Data Memory, expandable externally to 64 kbytes
- Five 8-bit ports; 40 I/O lines
- Three 16-bit Timer/Event counters
- External memory expandable up to 128 kbytes, external ROM up to 64 kbytes and/or RAM up to 64 kbytes
- On-chip oscillator suitable for RC, LC, quartz crystal or ceramic resonator
- Fifteen source, fifteen vector nested interrupt structure with two priority levels
- Full duplex serial port (UART)
- I²C-bus interface for serial transfer on two lines
- Analog-to-digital converter (ADC) with Power-down mode; 4 input channels and 8-bit ADC
- Pulse Width Modulated (PWM) output (8-bit resolution)
- Watchdog Timer
- Enhanced architecture with:
  - non-page oriented instructions
  - direct addressing
  - four 8-byte RAM register banks
  - stack depth limited only by available internal RAM (maximum 256 bytes)
  - multiply, divide, subtract and compare instructions
- Modes of reduced activity: Power-down and Idle modes
- Wake-up via external interrupts at Port 1
- Frequency range: 0 to 12 MHz. For ADC operation minimum 250 kHz at 2.7 V
- Supply voltage: 2.5 to 6.0 V
- Very low current consumption: typically 4.5 mA at 2.5 V and 8 MHz
- Operating temperature range: –40 to +85 °C.

## 2 GENERAL DESCRIPTION

The P80CL580; P83CL580 (hereafter generally referred to as P8xCL580) is manufactured in an advanced CMOS technology. The instruction set of the P8xCL580 is based on that of the 80C51 and consists of over 100 instructions: 49 one-byte, 46 two-byte, and 16 three-byte. The device operates over a wide range of supply voltages and has low power consumption; there are two software selectable modes for power reduction: Idle and Power-down. For emulation purposes, the P85CL580 (piggy-back version) with 256 bytes of RAM is recommended.

This data sheet details the specific properties of the P80CL580; P83CL580; for details of the 80C51 core see *"Data Handbook IC20"* and for the I²C-bus refer to *"The I²C-bus and how to use it"* ordering nr. 9398 393 40011".

### 2.1 ROMless version: P80CL580

The P80CL580 is the ROMless version of the P83CL580. The mask options on the P80CL580 are fixed as follows:

- All ports have option '1S' (standard port, HIGH after reset), except ports P1.6 and P1.7 which have option '2S' (open-drain, HIGH after reset)
- Oscillator option: Oscillator 3
- Power-on-reset option: off.

## 3 APPLICATIONS

The P8xCL580 is an 8-bit general purpose microcontroller especially suited for cordless telephone and mobile communication applications. The P8xCL580 also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities.

## 4 ORDERING INFORMATION

| TYPE NUMBER | PACKAGE | | |
|---|---|---|---|
| | NAME | DESCRIPTION | VERSION |
| P80CL580HFT | VSO56 | plastic very small outline package; 56 leads | SOT190-1 |
| P83CL580HFT | | | |
| P80CL580HFH | QFP64 | plastic quad flat package; 64 leads (lead length 1.95 mm); body 14 x 20 x 2.8 mm | SOT319-2 |
| P83CL580HFH | | | |

## 5 BLOCK DIAGRAM

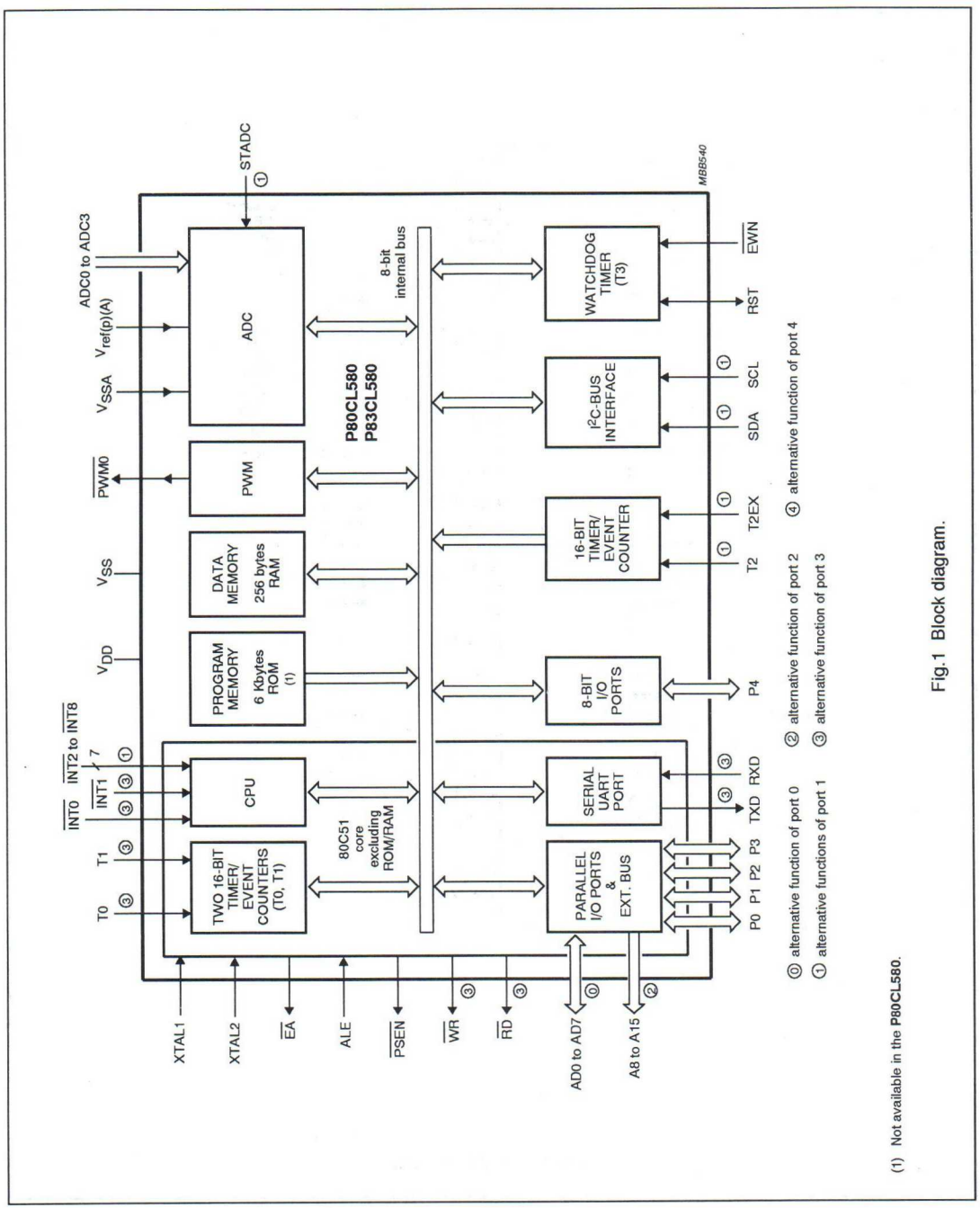



Fig.1 Block diagram.

(1) Not available in the **P80CL580**.

## 6 FUNCTIONAL DIAGRAM

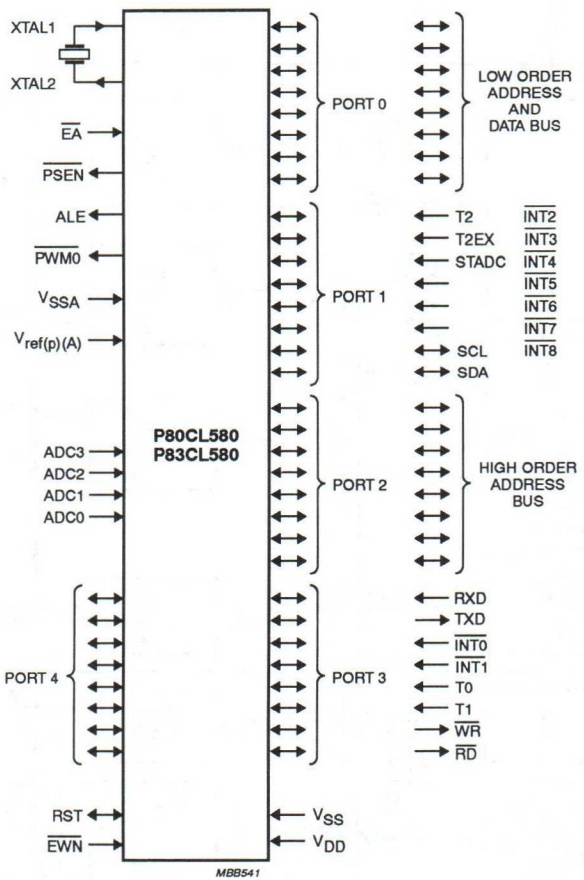



Fig.2 Functional diagram.

## 7 PINNING INFORMATION

### 7.1 Pinning

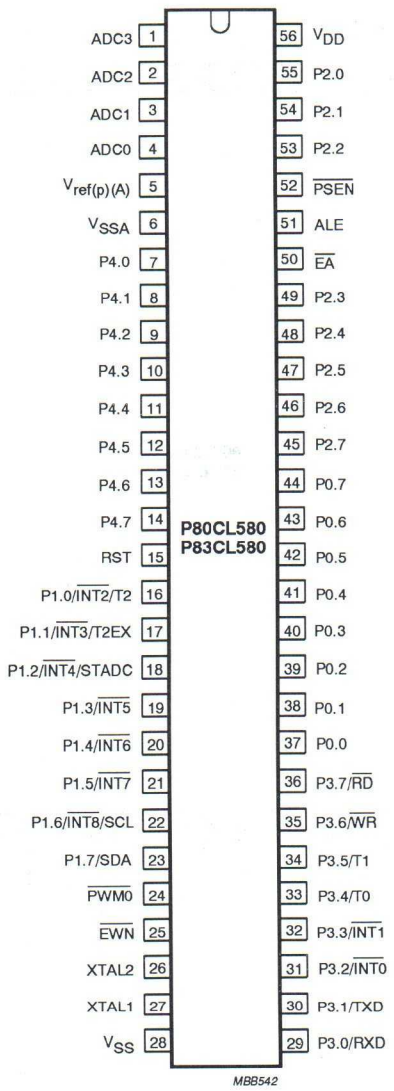



Fig.3 Pin configuration for VSO56 package.

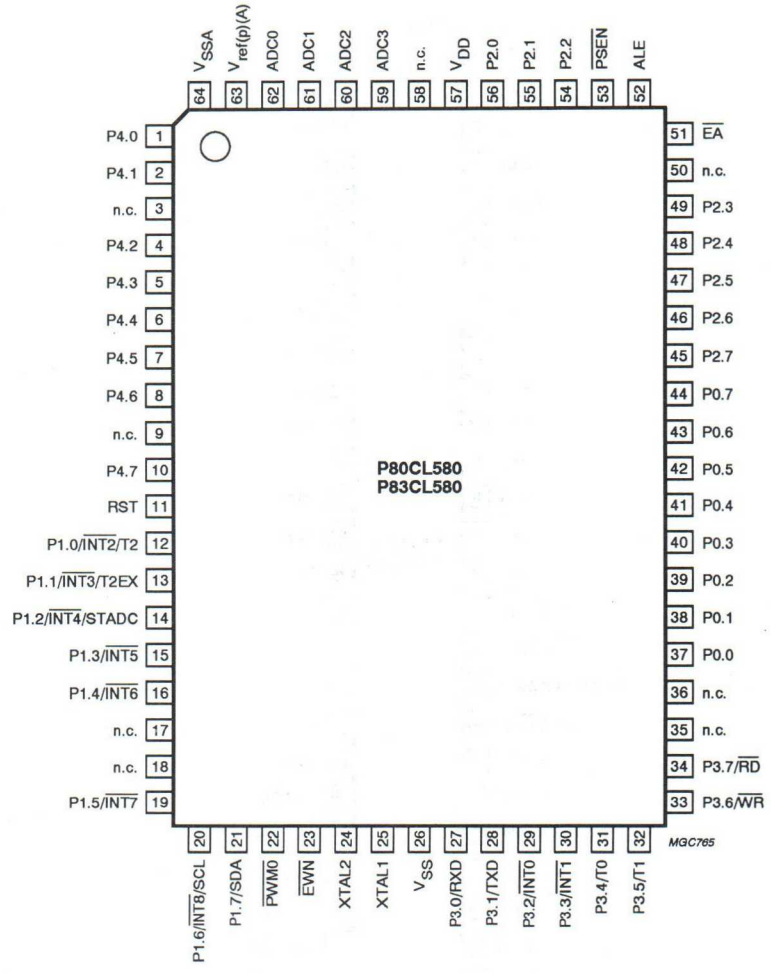



Fig.4 Pin configuration for QFP64 package.

### 7.2 Pin description

**Table 1** Pin description for VSO56 (SOT190-1) and QFP64 (SOT319-2)
For more extensive description of the port pins see Chapter 10 "I/O facilities".

| SYMBOL | PIN | | DESCRIPTION |
|---|---|---|---|
| | VSO56 | QFP64 | |
| ADC3 to ADC0 | 1 to 4 | 59 to 62 | 4 input channels to the ADC. |
| $V_{ref(p)(A)}$ | 5 | 63 | Positive potential of analog-to-digital conversion reference resistor. |
| $V_{SSA}$ | 6 | 64 | Analog part ground. |
| P4.0 to P4.7 | 7 to 14 | 1, 2, 4 to 8, 10 | **Port 4**: 8-bit bidirectional I/O port. |
| RST | 15 | 11 | **Reset**: a HIGH level on this pin for two machine cycles while the oscillator is running resets the device. |
| P1.0/$\overline{INT2}$/T2 | 16 | 12 | **Port 1 (P1.0 to P1.7)**: 8-bit bidirectional I/O port with internal pull-ups; **$\overline{INT2}$ to $\overline{INT8}$**: external interrupt inputs; **T2**: Timer T2 input; **T2EX**: Timer T2 external input; **STADC**: external trigger of the ADC; **SCL**: I²C-bus interface clock; **SDA**: I²C-bus interface data.<br><br>Port 1 pins that have logic 1s written to them are pulled HIGH by the internal pull-ups, and in that state can be used as inputs (note P1.6 and P1.7 are open-drain only). The Port 1 output buffer can sink/source 4 LS TTL loads. As inputs, Port 1 pins that are externally pulled LOW will source current ($I_{IL}$ see Chapter 23) due to the internal pull-ups. |
| P1.1/$\overline{INT3}$/T2EX | 17 | 13 | |
| P1.2/$\overline{INT4}$/STADC | 18 | 14 | |
| P1.3/$\overline{INT5}$ | 19 | 15 | |
| P1.4/$\overline{INT6}$ | 20 | 16 | |
| P1.5/$\overline{INT7}$ | 21 | 19 | |
| P1.6/$\overline{INT8}$/SCL | 22 | 20 | |
| P1.7/SDA | 23 | 21 | |
| $\overline{PWM0}$ | 24 | 22 | **Pulse Width Modulation output 0**. |
| $\overline{EWN}$ | 25 | 23 | **Enable Watchdog Timer**: enable for Watchdog Timer and enable Power-down mode. |
| XTAL2 | 26 | 24 | **Crystal oscillator output**: output of the inverting amplifier of the oscillator. Left open when external clock is used. |
| XTAL1 | 27 | 25 | **Crystal oscillator input**: input to the inverting amplifier of the oscillator, also the input for an externally generated clock source. |
| $V_{SS}$ | 28 | 26 | Ground: circuit ground potential. |
| P3.0/RXD | 29 | 27 | **Port 3 (P3.0 to P3.7)**: 8-bit bidirectional I/O port with internal pull-ups; **RXD**: serial port receiver data input (asynchronous); **TXD**: serial port transmitter data output (asynchronous); **$\overline{INT0}$**: external interrupt 0; **$\overline{INT1}$**: external interrupt 1; **T0**: Timer 0 external input; **T1**: Timer 1 external input; **$\overline{WR}$**: external Data Memory write strobe; **$\overline{RD}$**: external Data Memory read strobe.<br><br>The Port 3 output buffers can sink/source 4 LS TTL inputs. Port 3 pins that have logic 1s written to them are pulled HIGH by the internal pull-ups, and in that state can be used as inputs. As inputs, Port 3 pins that are externally pulled LOW will source current ($I_{IL}$ in the characteristics; see Chapter 23) due to the internal pull-ups. |
| P3.1/TXD | 30 | 28 | |
| P3.2/$\overline{INT0}$ | 31 | 29 | |
| P3.3/$\overline{INT1}$ | 32 | 30 | |
| P3.4/T0 | 33 | 31 | |
| P3.5/T1 | 34 | 32 | |
| P3.6/$\overline{WR}$ | 35 | 33 | |
| P3.7/$\overline{RD}$ | 36 | 34 | |

| SYMBOL | PIN | | DESCRIPTION |
|---|---|---|---|
| | VSO56 | QFP64 | |
| P0.0 to P0.7 | 37 to 44 | 37 to 44 | **Port 0**: 8-bit open-drain bidirectional I/O port. As an open-drain output port it can sink 8 LS TTL loads. Port 0 pins that have logic 1s written to them float, and in that state will function as high impedance inputs. Port 0 is also the multiplexed low order address and data bus during access to external memory. The strong internal pull-ups are used while emitting logic 1s within the low order address. |
| P2.0 to P2.7 | 55 to 53, 49 to 45 | 56 to 54, 49 to 45 | **Port 2**: 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have logic 1s written to them are pulled HIGH by the internal pull-ups, and in that state can be used as inputs. The Port 2 output buffer can sink/source 4 LS TTL loads. Port 2 emits the high-order address byte during accesses to external memory that use 16-bit addresses (MOVX @DPTR). In this application it uses the strong internal pull-ups when emitting logic 1s. During accesses to external memory that use 8-bit addresses (MOVX @Ri). Port 2 emits the contents of the P2 Special Function Register. |
| $\overline{EA}$ | 50 | 51 | **External Access**. When $\overline{EA}$ is held HIGH the CPU executes out of internal Program Memory (unless the program counter exceeds 17FFH). Holding $\overline{EA}$ LOW forces the CPU to execute out of external memory regardless of the value of the Program Counter. |
| ALE | 51 | 52 | **Address Latch Enable**. Output pulse for latching the low byte of the address during access to external memory. ALE is emitted at a constant rate of $\frac{1}{6} \times f_{osc}$, and may be used for external timing or clocking purposes (assuming MOVX instructions are not used). |
| $\overline{PSEN}$ | 52 | 53 | **Program Store Enable**. Output read strobe to external Program Memory. When executing code out of external Program Memory, $\overline{PSEN}$ is activated twice each machine cycle. However, during each access to external Data Memory two $\overline{PSEN}$ activations are skipped. |
| $V_{DD}$ | 56 | 57 | Power supply. |
| n.c. | – | 3, 9, 17, 18, 35, 36, 50 and 58 | Not connected. |

## 8 FUNCTIONAL DESCRIPTION

The functional description of the device is described in:

Chapter 9 "Memory organization"

Chapter 10 "I/O facilities"

Chapter 11 "Timer/event counters"

Chapter 12 "Pulse Width Modulated output"

Chapter 13 "Analog-to-digital converter (ADC)"

Chapter 14 "Reduced power modes"

Chapter 15 "I2C-bus serial I/O"

Chapter 16 "Standard serial interface SIO0: UART"

Chapter 17 "Interrupt system"

Chapter 18 "Oscillator circuitry"

Chapter 19 "Reset".

### 8.1 General

The P8xCL580 is a stand-alone high-performance CMOS microcontroller designed for use in real-time applications such as cordless telephone and mobile communications, instrumentation, industrial control, intelligent computer peripherals and consumer products.

The device provides hardware features, architectural enhancements and new instructions to function as a controller for applications requiring up to 64 kbytes of Program Memory and/or up to 64 kbytes of Data Memory.

The P8xCL580 contains a 6 kbytes Program Memory (ROM; P83CL580); a static 256 bytes Data Memory (RAM); 40 I/O lines; three 16-bit timer/event counters; a fifteen-source two priority-level, nested interrupt structure and on-chip oscillator and timing circuit, 4-channel 8-bit A/D converter, Watchdog Timer and Pulse Width Modulation output.

The device has two software selectable modes of reduced activity for power reduction:

- **Idle mode**; freezes the CPU while allowing the derivative functions (timers, serial I/O, ADC, PWM) and interrupt system to continue functioning.
- **Power-down mode**; saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative.

In addition, two serial interfaces are provided on-chip:

- a standard UART serial interface, and
- a standard $I^2C$-bus serial interface. The $I^2C$-bus serial interface has byte oriented master and slave functions allowing communication with the whole family of $I^2C$-bus compatible devices.

### 8.2 CPU timing

A machine cycle consists of a sequence of 6 states. Each state lasts for two oscillator periods, thus a machine cycle takes 12 oscillator periods or 1 μs if the oscillator frequency ($f_{osc}$) is 12 MHz.

## 9 MEMORY ORGANIZATION

The P8xCL580 has 6 kbytes of Program Memory (ROM; P83CL580 only) plus 256 bytes of Data Memory (RAM) on board. The device has separate address spaces for Program and Data Memory (see Fig.6). Using Port latches P0 and P2, the P8xCL580 can address up to 64 kbytes of external memory. The CPU generates both read ($\overline{RD}$) and write ($\overline{WR}$) signals for external Data Memory accesses, and the read strobe ($\overline{PSEN}$) for external Program Memory.

### 9.1 Program Memory

The P83CL580 contains 6 kbytes of internal ROM. After reset the CPU begins execution at location 0000H. The lower 6 kbytes of Program Memory can be implemented in either on-chip ROM or external Program Memory.

If the $\overline{EA}$ pin is tied to $V_{DD}$, then Program Memory fetches from addresses 0000H to 17FFH are directed to the internal ROM. Fetches from addresses 1800H to FFFFH are directed to external ROM. Program Counter values greater than 17FFH are automatically addressed to external memory regardless of the state of the $\overline{EA}$ pin.

### 9.2 Data Memory

The P8xCL580 contains 256 bytes of internal RAM and 40 Special Function Registers (SFRs). Figure 6 shows the internal Data Memory space divided into the lower 128 bytes, the upper 128 bytes, and the SFRs space. Internal RAM locations 0 to 127 are directly and indirectly addressable. Internal RAM locations 128 to 255 are only indirectly addressable. The Special Function Register locations 128 to 255 bytes are only directly addressable.

### 9.3 Special Function Registers (SFRs)

The upper 128 bytes are the address locations of the SFRs. Figures 7 and 8 show the Special Function Registers space. The SFRs include the port latches, timers, peripheral control, serial I/O registers, etc. These registers can only be accessed by direct addressing. There are 128 directly addressable locations in the SFR address space. Bit addressable SFRs are those that end in 000B.

### 9.4 Addressing

The P8xCL580 has five methods for addressing source operands:

- Register
- Direct
- Indirect
- Immediate
- Base-Register plus Index-Register-Indirect.

The first three methods can be used for addressing destination operands. Most instructions have a 'destination/source' field that specifies the data type, addressing methods and operands involved. For operations other than MOVs, the destination operand is also a source operand.

Access to memory addressing is as follows:

- Registers in one of the four register banks through Direct or Indirect
- Lower 128 bytes of internal RAM through Direct or Indirect; upper 128 bytes of internal RAM through Direct
- Special Function Registers through Direct
- External Data Memory through Indirect
- Program Memory look-up tables through Base-Register plus Index-Register-Indirect.

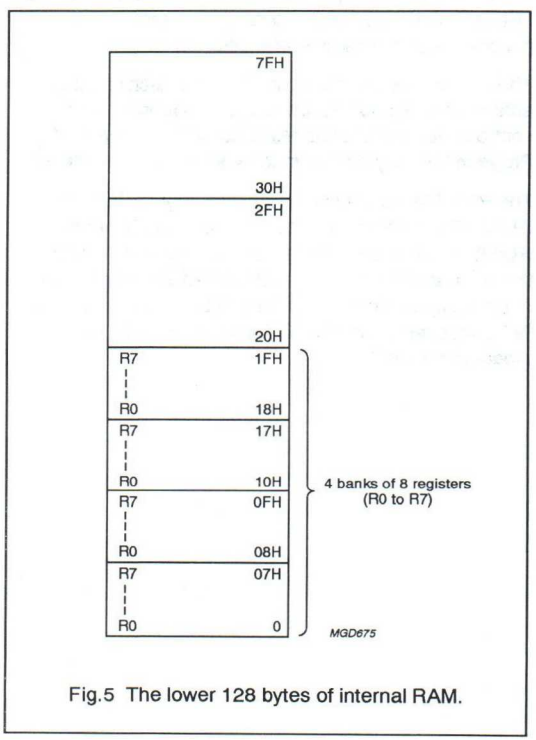
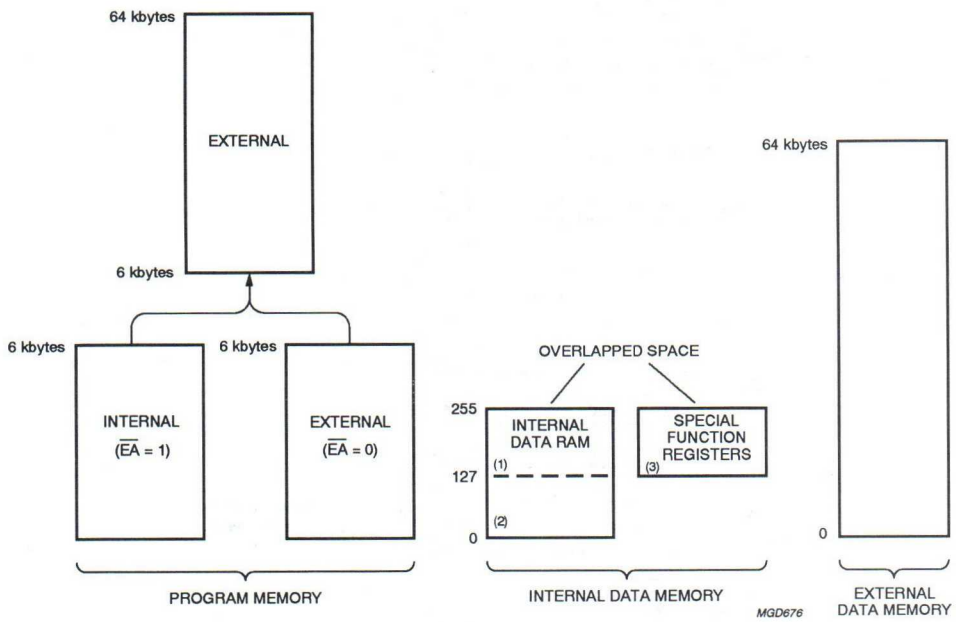



Fig.5 The lower 128 bytes of internal RAM.




(1) Accessible via indirect addressing only.
(2) Accessible via direct and indirect addressing.
(3) Accessible via direct addressing.

Fig.6 Memory map.

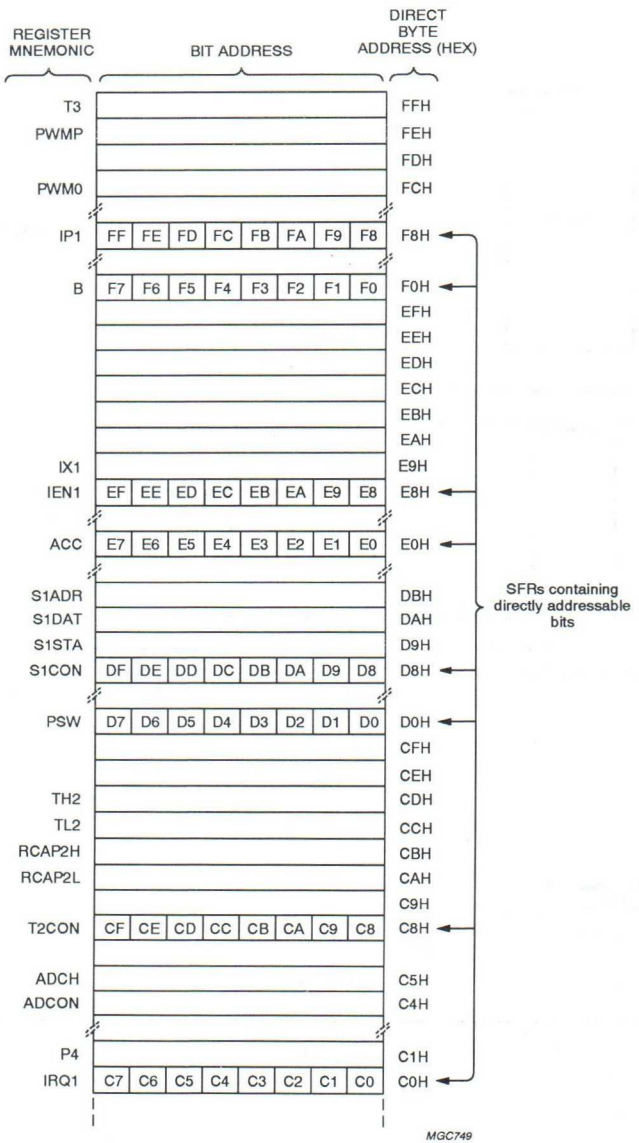



Fig.7 Special Function Register memory map.

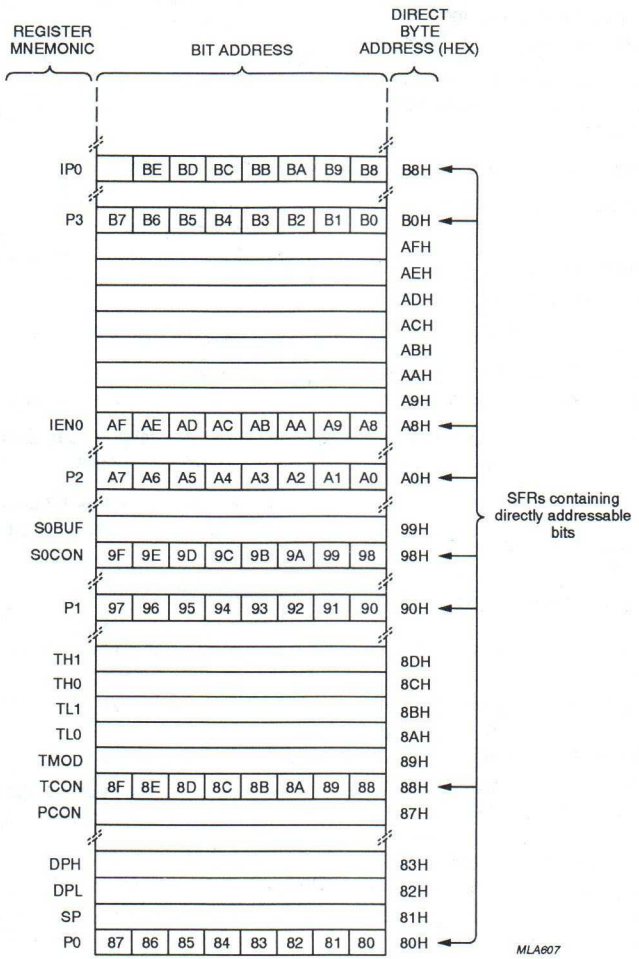



Fig.8 Special Function Register memory map (continued).

## 10 I/O FACILITIES

### 10.1 Ports

The P8xCL580 has 40 I/O lines treated as one 8-bit port plus 32 individually addressable bits or as five parallel 8-bit addressable ports. Port 4 has no alternative function. Ports 0, 1, 2 and 3 have the following alternative functions:

Port 0 Provides the multiplexed low-order address and data bus for expanding the device with standard memories and peripherals.

Port 1 Used for a number of special functions:

- Provides the inputs for the external interrupts: $\overline{INT2}$ to $\overline{INT8}$.
- External activation of Timer 2: T2.
- External trigger of the ADC: STADC.
- The $I^2C$-bus interface: SCL and SDA.

Port 2 Provides the high-order address when expanding the device with external Program or Data Memory.

Port 3 Pins can be configured individually to provide:

- External interrupt request inputs: $\overline{INT1}$ and $\overline{INT0}$.
- Counter input: T1 and T0.
- Control signals to read and write to external memories: $\overline{RD}$ and $\overline{WR}$.
- UART input and output: RXD and TXD.

To enable a port pin alternative function, the port bit latch in its SFR must contain a logic 1.

Each port consists of a latch (SFRs P0 to P4), an output driver and input buffer. Ports 1, 2, 3 and 4 have internal pull-ups (except P1.6 and P1.7). Figure 9(a) shows that the strong transistor 'p1' is turned on for only 2 oscillator periods after a LOW-to-HIGH transition in the port latch. When on, it turns on 'p3' (a weak pull-up) through the inverter. This inverter and 'p3' form a latch which holds the logic 1. In Port 0 the pull-up 'p1' is only on when emitting logic 1s for external memory access. Writing a logic 1 to a Port 0 bit latch leaves both output transistors switched off so that the pin can be used as an high-impedance input.

### 10.2 Port options

Thirtyeight of the 40 port pins (excluding P1.6 and P1.7 with option 2S only) may be individually configured with one of the following options. These options are also shown in Fig.9.

Option 1 **Standard Port**; quasi-bidirectional I/O with pull-up. The strong booster pull-up 'p1' is turned on for two oscillator periods after a LOW-to-HIGH transition in the port latch; Fig.9(a).

Option 2 **Open-drain**; quasi-bidirectional I/O with n-channel open-drain output. Use as an output requires the connection of an external pull-up resistor; see Fig.9(b).

Option 3 **Push-pull**; output with drive capability in both polarities. Under this option, pins can only be used as outputs; see Fig.9(c).

### 10.3 Port 0 options

The definition of port options for Port 0 is slightly different. Two cases are considered. First, access to external memory ($\overline{EA}$ = 0 or access above the built-in memory boundary) and second, I/O accesses.

#### 10.3.1 External memory accesses

Option 1 True logic 0 and logic 1 are written as address to the external memory (strong pull-up to be used).

Option 2 An external pull-up resistor is required for external accesses.

Option 3 Not allowed for external memory accesses as the port can only be used as output.

#### 10.3.2 I/O Accesses

Option 1 When writing a logic 1 to the port latch, the strong pull-up 'p1' will be on for 2 oscillator periods. No weak pull-up exists. Without an external pull-up, this option can be used as a high-impedance input.

Option 2 Open-drain; quasi-directional I/O with n-channel open-drain output. Use as an output requires the connection of an external pull-up resistor. See Fig.9(b).

Option 3 Push-Pull; output with drive capability in both polarities. Under this option pins can only be used as outputs. See Fig.9(c).

### 10.4 SET/RESET options

Individual mask selection of the post-reset state is available with any of the above pins. The required selection is made by appending 'S' or 'R' to Options 1, 2, or 3 above.

Option R RESET, at reset this pin will be initialized LOW.

Option S SET, at reset this pin will be initialized HIGH.

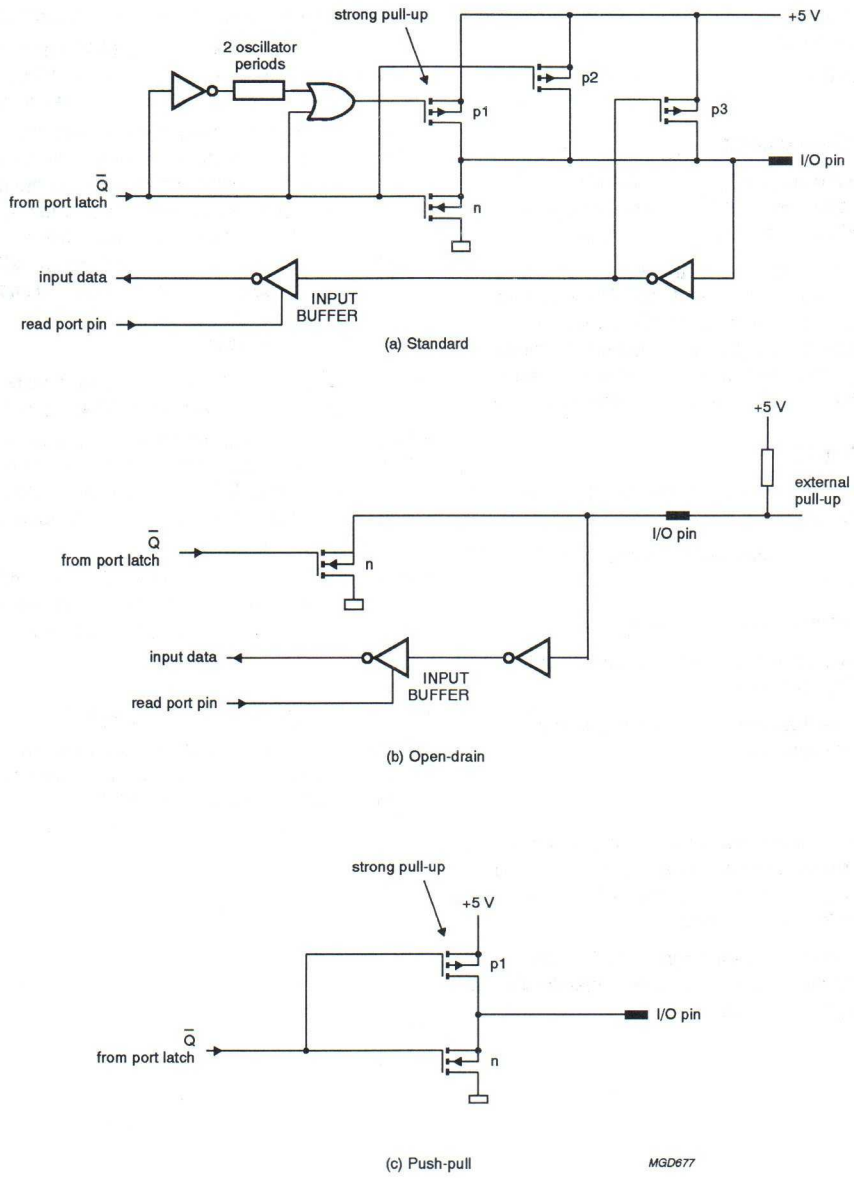



Fig.9 Port configuration options.

## 11 TIMER/EVENT COUNTERS

The P8xCL580 contains three 16-bit timer/event counter registers; Timer 0, Timer 1 and Timer 2 which can perform the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests.

In the 'Timer' operating mode the register is incremented every machine cycle. Since a machine cycle consists of 12 oscillator periods, the count rate is $1/12 \times f_{osc}$.

In the 'Counter' operating mode, the register is incremented in response to a HIGH-to-LOW transition. Since it takes 2 machine cycles (24 oscillator periods) to recognize a HIGH-to-LOW transition, the maximum count rate is $1/24 \times f_{osc}$. To ensure a given level is sampled, it should be held for at least one complete machine cycle.

### 11.1 Timer 0 and Timer 1

Timer 0 and Timer 1 can be programmed independently to operate in four modes:

Mode 0 8-bit timer or 8-bit counter each with divide-by-32 prescaler.

Mode 1 16-bit time-interval or event counter.

Mode 2 8-bit time-interval or event counter with automatic reload upon overflow.

Mode 3 Timer 0 establishes TL0 and TH0 as two separate counters.

### 11.2 Timer T2

Timer T2 is a 16-bit timer/counter that can operate (like Timer 0 and 1) either as a timer or as an event counter. These functions are selected by the state of the C/$\overline{T2}$ bit in the T2CON register; see Tables 2 and 3.

Three operating modes are available Capture, Auto-reload and Baud Rate Generator, which also are selected via the T2CON register; see Table 4.

#### 11.2.1 CAPTURE MODE

Figure 10 shows the Capture mode. Two options in this mode, may be selected by the EXEN2 bit in T2CON:

- If EXEN2 = 0, then Timer 2 is a 16-bit timer or counter which upon overflowing sets the Timer 2 overflow bit TF2, this may then be used to generate an interrupt.
- If EXEN2 = 1, Timer 2 operates as described above but with the additional feature that a HIGH-to-LOW transition at external input T2EX causes the current value in TL2 and TH2 to be captured into registers RCAP2L and RCAP2H respectively. In addition, the transition at T2EX causes the EXF2 bit in T2CON to be set; this may also be used to generate an interrupt.

#### 11.2.2 AUTO-RELOAD MODE

Figure 11 shows the Auto-reload mode. Also two options in this mode are selected by the EXEN2 bit in T2CON:

- If EXEN2 = 0, then when Timer 2 rolls over, it sets the TF2 bit but also causes the Timer 2 registers to be reloaded with the 16-bit value held in registers RCAP2L and RCAP2H. The 16-bit value held in these registers is preset by software.
- If EXEN2 = 1, Timer 2 operates as described above but with the additional feature that a HIGH-to-LOW transition at external input T2EX will also trigger the 16-bit reload and set the EXF2 bit.

#### 11.2.3 BAUD RATE GENERATOR MODE

The Baud Rate Generator mode is selected when RTCLK = 1. It will be described in conjunction with the serial port (UART); see Section 16.3.2.

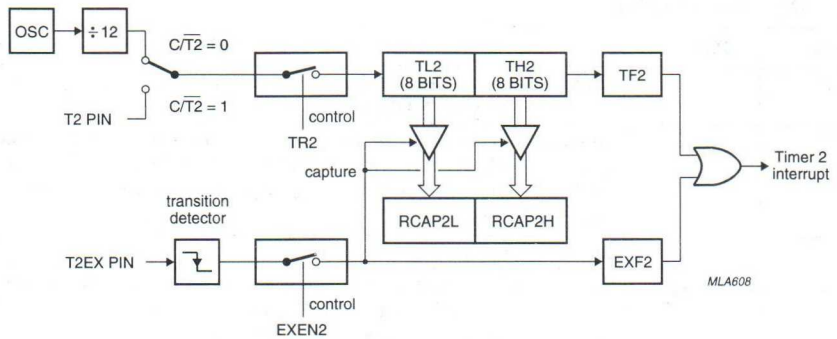



Fig.10 Timer 2 in Capture mode.

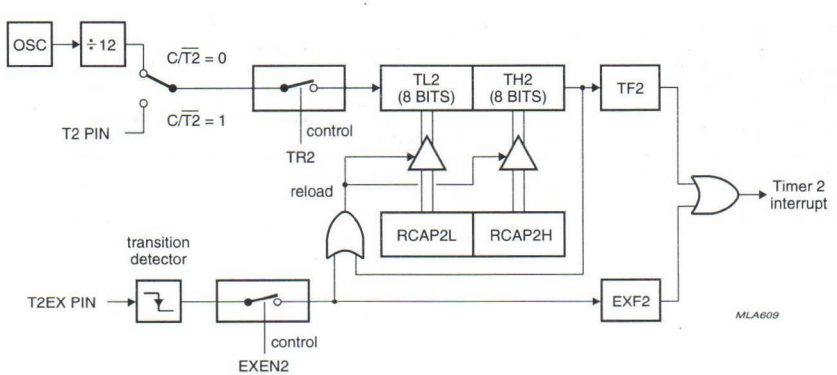



Fig.11 Timer 2 in Auto-Reload mode.

### 11.3 Timer/Counter 2 Control Register (T2CON)

**Table 2** Timer/Counter 2 Control Register (SFR address C8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| TF2 | EXF2 | GF2 | RTCLK | EXEN2 | TR2 | C/$\overline{T2}$ | CP/$\overline{RL2}$ |

**Table 3** Description of T2CON bits.

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | TF2 | **Timer 2 overflow flag**. Set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when RTCLK = 1. |
| 6 | EXF2 | **Timer 2 external flag.** Set when either a capture or reload is caused by a negative transition on T2EX and when EXEN2 = 1. When Timer T2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to Timer 2 interrupt routine. EXF2 must be cleared by software. |
| 5 | GF2 | **General purpose flag** bit. |
| 4 | RTCLK | **Receive/transmit clock flag**. When set, causes the UART serial port to use Timer 2 overflow pulses for its receive and transmit clock in Modes 1 and 3. RTCLK = 0 causes Timer 1 overflows to be used for the receive and transmit clock. |
| 3 | EXEN2 | **Timer 2 external enable flag**. When set, allows a capture or reload to occur as a result of a negative transition on T2EX, if Timer 2 is not being used to clock the serial port. EXEN2 = 0, causes Timer 2 to ignore events at T2EX. |
| 2 | TR2 | **Start/stop control for Timer 2**. TR2 = 1 starts the timer. |
| 1 | C/$\overline{T2}$ | **Timer or counter select for Timer 2**. C/$\overline{T2}$ = 0 selects the internal timer with a clock frequency of $^1/_{12} \times f_{osc}$. C/$\overline{T2}$ = 1 selects the external event counter; negative edge triggered. |
| 0 | CP/$\overline{RL2}$ | **Capture/Reload flag**. When set, captures will occur on negative transitions at T2EX, if EXEN2 = 1. When cleared, auto-reloads will occur either with Timer 2 overflows or negative transitions at T2EX when EXEN2 = 1. When RTCLK = 1, this bit is ignored and the timer is forced to auto-reload on a Timer 2 overflow. |

**Table 4** Timer 2 operating modes; X = don't care.

| RTCLK | CP/$\overline{RL2}$ | TR2 | MODE |
|---|---|---|---|
| 0 | 0 | 1 | 16-bit Auto-reload |
| 0 | 1 | 1 | 16-bit Capture |
| 1 | X | 1 | Baud Rate Generator |
| X | X | 0 | Off |

### 11.4 Watchdog Timer

In addition to Timer T2 and the standard timers, a Watchdog Timer (consisting of an 11-bit prescaler and an 8-bit timer) is also incorporated.

The Watchdog Timer is controlled by the Watchdog Enable pin ($\overline{EWN}$). When $\overline{EWN}$ = 0, the timer is enabled and the Power-down mode is disabled. When $\overline{EWN}$ = 1, the timer is disabled and the Power-down mode is enabled. In the Idle mode the Watchdog Timer and reset circuitry remain active.

The Watchdog Timer is shown in Fig. 12.

The timer frequency is derived from the oscillator frequency using the following formula:

$$f_{timer} = \frac{f_{osc}}{(12 \times 2048)}.$$

When a timer overflow occurs, the microcontroller is reset and a reset output pulse is generated at the RST pin. To prevent a system reset the timer must be reloaded in time by the application software. If the processor suffers a hardware/software malfunction, the software will fail to reload the timer. This failure will produce a reset upon overflow thus preventing the processor running out of control.

The Watchdog Timer can only be reloaded if the condition flag WLE (PCON.4) has been previously set by software. At the moment the counter is loaded the condition flag is automatically cleared.

The time interval between the timer reloading and the occurrence of a reset is dependent upon the reloaded value. For example, this time period may range from 2 ms to 500 ms when using an oscillator frequency $f_{osc}$ = 12 MHz.

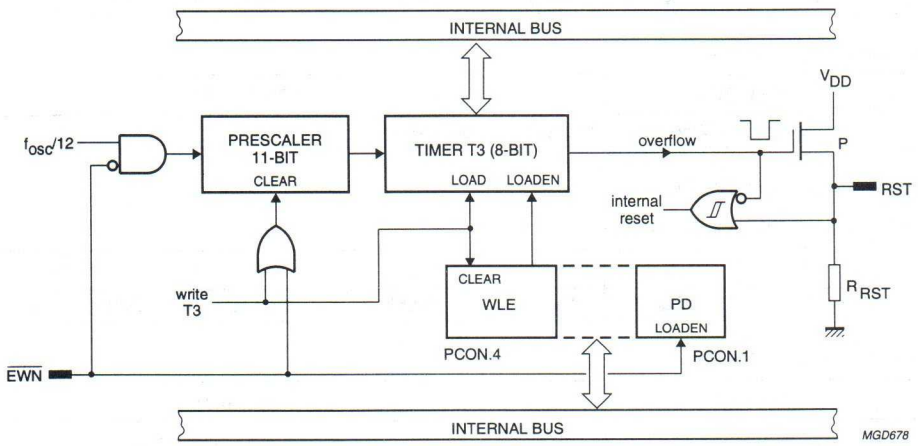



Fig.12 Functional diagram of the T3 Watchdog Timer.

## 12 PULSE WIDTH MODULATED OUTPUT

One Pulse Width Modulated output channel ($\overline{PWM0}$) is provided which outputs pulses of programmable length and interval. The repetition frequency is defined by an 8-bit prescaler (PWMP) that generates the clock for the counter. The 8-bit counter counts modulo 255, i.e. from 0 to 254 inclusive. The value held in the 8-bit counter is compared to the contents of the register PWM0.

Provided the contents of this register are greater than the counter value, the $\overline{PWM0}$ output is set LOW. If the contents of register PWM0 are equal to, or less than the counter value, the $\overline{PWM0}$ output is set HIGH. The pulse-width-ratio is therefore defined by the contents of register PWM0. The pulse-width-ratio will be in the range 0 to $^{255}/_{255}$ and may be programmed in increments of $^{1}/_{255}$.

The repetition frequency ($f_{PWM}$) at the $\overline{PWM0}$ output is given by:

$$f_{PWM} = \frac{f_{osc}}{\{2 \times (1 + PWMP) \times 255\}}$$

For $f_{osc}$ = 12 MHz the above formula gives a repetition frequency range of 92 Hz to 23.5 kHz.

By loading the PWM0 register with either 00H or FFH, the $\overline{PWM0}$ output can be retained at a constant HIGH or LOW level respectively. When loading FFH into the PWM0 register, the 8-bit counter will never actually reach this value.

The $\overline{PWM0}$ output pin is driven by push-pull drivers and is not shared with any other function.

### 12.1 Prescaler Frequency Control Register (PWMP)

**Table 5** Prescaler Frequency Control Register (address FEH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| PWMP.7 | PWMP.6 | PWMP.5 | PWMP.4 | PWMP.3 | PWMP.2 | PWMP.1 | PWMP.0 |

**Table 6** Description of PWMP bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 to 1 | PWMP.7 to PWMP.0 | Prescaler division factor = (PWMP) + 1. |

### 12.2 Pulse Width Register (PWM0)

**Table 7** Pulse Width Register (address FCH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| PWM0.7 | PWM0.6 | PWM0.5 | PWM0.4 | PWM0.3 | PWM0.2 | PWM0.1 | PWM0.0 |

**Table 8** Description of PWM0 bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 to 1 | PWM0.7 to PWM0.0 | LOW/HIGH ratio of $\overline{PWM0}$ signal = $\frac{(PWM0)}{\{255 - (PWM0)\}}$ |

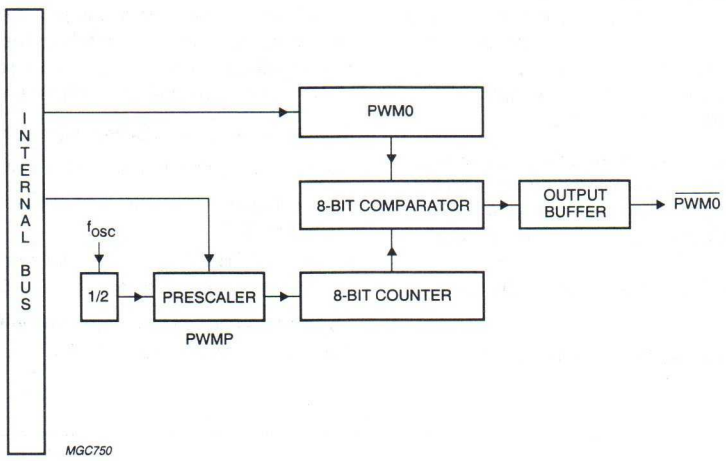



Fig.13 Functional diagram of Pulse Width Modulated output ($\overline{PWM0}$).

## 13 ANALOG-TO-DIGITAL CONVERTER (ADC)

The analog input circuitry consists of a 4-bit analog multiplexer and an ADC with 8-bit resolution. The analog reference voltage ($V_{ref(p)(A)}$) and analog ground ($V_{SSA}$) are connected via separate input pins. The conversion is selectable from 24 machine cycles (24 μs at $f_{osc}$ = 12 MHz) to 48 machine cycles. The functional diagram of the ADC is shown in Fig. 14.

The ADC is controlled using the ADC Control Register (ADCON). Input channels are selected by the analog multiplexer via the ADCON register bits AADR0 and AADR1. The completion of the 8-bit ADC conversion is flagged by ADCI in the ADCON register and the result is stored in the Special Function Register ADCH (address C5H).

An ADC conversion in progress is unaffected by an external software ADC start.

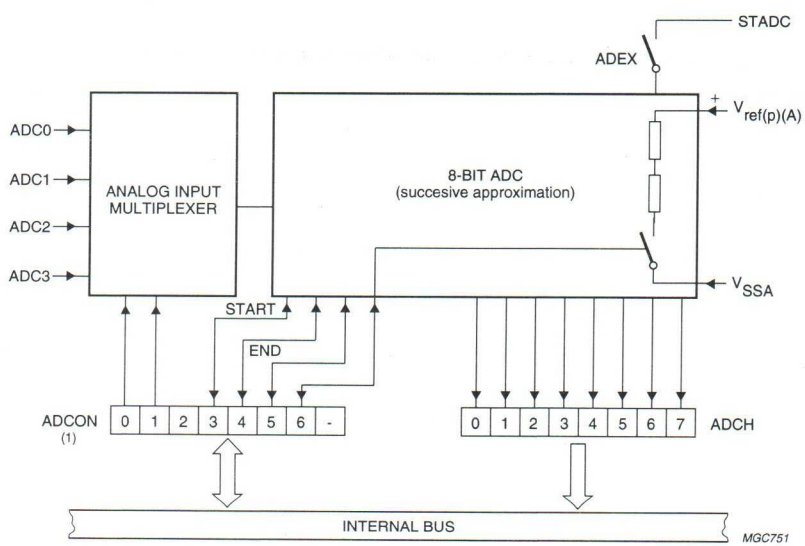
The result of a completed conversion remains unaffected provided ADCI = 1. While ADCS = 1 or ADCI = 1, a new ADC start will be blocked and consequently lost.

An ADC conversion already in progress is aborted when the Power-down mode is entered. The result of a completed conversion (ADCI = 1) remains unaffected when entering the Idle or Power-down mode.

The analog-to-digital conversion can be started in 3 ways:

- Start in operating mode, continue in operating mode.
- Start in operating mode, by setting the ADCS bit, then go to Idle mode.
- Set the ADEX bit, go to the Idle mode and start conversion externally via the STADC pin.

For the three cases mentioned above the internal flag ADCI is set upon completion of the conversion.




(1) For the descriptions of ADCON bits see Table 10.

Fig.14 Functional diagram of analog input.

### 13.1 ADC Control Register (ADCON)

**Table 9** ADC Control Register (address C4H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | ADPD | ADEX | ADCI | ADCS | CKDIV | AADR1 | AADR0 |

**Table 10** Description of ADCON bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | – | Reserved. |
| 6 | ADPD | **Power-down.** This bit switches off the resistor reference to save power even when the CPU is operating. |
| 5 | ADEX | **Enable external start of conversion**. This bit determines whether a conversion can be started using the external pin STADC. When ADEX = 0, a conversion cannot be started externally using STADC. When ADEX = 1, a conversion can be started externally using STADC. |
| 4 | ADCI | **ADC interrupt flag**. This flag is set when an ADC conversion result is ready to be read. An interrupt is invoked if this is enabled. This flag must be cleared by software (it cannot be set by software); see Table 11. |
| 3 | ADCS | **ADC start and status flag**. When this bit is set an ADC conversion is started. ADCS may be set by software or by the external signal STADC. The ADC logic ensures that this signal is HIGH while the ADC is busy. On completion of the conversion ADCS is reset and after that the interrupt flag ADCI is set. ADCS cannot be reset by software; see Table 11. |
| 2 | CKDIV | This bit selects the conversion time, in terms of instruction cycles. This allows the CPU to be run at the maximum frequency (12 MHz) yet keeping the ADC timing at low frequency. When CKDIV = 0, the conversion time is equivalent to 24 instruction cycles. When CKDIV = 1, the conversion time is equivalent to 48 instruction cycles. The conversion time includes a sampling time of 6 cycles. |
| 1 | AADR1 | **Analog input select**. These bits are used to select one of the four analog inputs; see Table 12. They only can be changed when ADCI and ADCS are both LOW. |
| 0 | AADR0 | |

**Table 11** Analog-to-digital operation

| ADCI | ADCS | OPERATION |
|---|---|---|
| 0 | 0 | ADC not busy; a conversion can be started. |
| 0 | 1 | ADC busy; start of a new conversion is blocked. |
| 1 | 0 | Conversion completed; start of a new conversion is blocked. |
| 1 | 1 | Intermediate status for a maximum of one machine cycle before conversion is completed (ADCI = 1, ADCS = 0). |

**Table 12** Selection of analog input channel

| AADR1 | AADR0 | SELECTED CHANNEL |
|---|---|---|
| 0 | 0 | AD0 |
| 0 | 1 | AD1 |
| 1 | 0 | AD2 |
| 1 | 1 | AD3 |

## 14 REDUCED POWER MODES

There are two software selectable modes of reduced activity for further power reduction: Idle and Power-down.

### 14.1 Idle mode

Idle mode operation permits the interrupt, serial ports, timer blocks, PWM and ADC to continue to function while the clock to the CPU is halted.

The following functions remain active during the Idle mode:

- Timer 0, Timer 1, Timer 2 and Timer 3
- UART, $I^2C$-bus interface
- External interrupt
- $\overline{PWM0}$ (reset; output = HIGH)
- ADC.

These functions may generate an interrupt or reset; thus ending the Idle mode.

The instruction that sets bit IDL (PCON.0) is the last instruction executed in the normal operating mode before the Idle mode is activated. Once in Idle mode, the CPU status is preserved along with the Stack Pointer, Program Counter, Program Status Word and Accumulator. The RAM and all other registers maintain their data during Idle mode. The status of the external pins during Idle mode is shown in Table 13.

There are two ways to terminate the Idle mode:

1. Activation of any enabled interrupt will cause IDL (PCON.0) to be cleared by hardware thus terminating the Idle mode. The interrupt is serviced, and following the RETI instruction, the next instruction to be executed will be the one following the instruction that put the device in the Idle mode. The flag bits GF0 (PCON.2) and GF1 (PCON.3) may be used to determine whether the interrupt was received during normal execution or during the Idle mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When the Idle mode is terminated by an interrupt, the service routine can examine the status of the flag bits.
2. The second way of terminating the Idle mode is with an external hardware reset, or an internal reset caused by an overflow of Timer T2. Since the oscillator is still running, the hardware reset is required to be active for two machine cycles (24 oscillator periods) to complete the reset operation. Reset redefines all SFRs but does not affect the on-chip RAM.

### 14.2 Power-down mode

The Power-down operation freezes the oscillator. The Power-down mode can only be activated by setting the PD bit in the PCON register.

The instruction that sets PD (PCON.1) is the last executed prior to going into the Power-down mode. Once in the Power-down mode, the oscillator is stopped. The contents of the on-chip RAM and the SFRs are preserved. The port pins output the value held by their respective SFRs. ALE and $\overline{PSEN}$ are held LOW.

In the Power-down mode, $V_{DD}$ may be reduced to minimize circuit power consumption. The supply voltage must not be reduced until the Power-down mode is entered, and must be restored before the hardware reset is applied which will free the oscillator. Reset should not be released until the oscillator has restarted and stabilized.

### 14.3 Wake-up mode

Setting the PD flag in the PCON register forces the controller into the Power-down mode. Setting this flag enables the controller to be woken-up from the Power-down mode with either the external interrupts $\overline{INT2}$ to $\overline{INT8}$, or a reset operation. The wake-up operation has two basic approaches as explained in Section 14.3.1; 14.3.2 and illustrated in Fig.15.

#### 14.3.1 WAKE-UP USING $\overline{INT2}$ TO $\overline{INT8}$

If any of the interrupts $\overline{INT2}$ to $\overline{INT8}$ are enabled, the device can be woken-up from the Power-down mode with the external interrupts. To ensure that the oscillator is stable before the controller restarts, the internal clock will remain inactive for 1536 oscillator periods. This is controlled by an on-chip delay counter.

#### 14.3.2 WAKE-UP USING RST

To wake-up the P8xCL580, the RST pin must be kept HIGH for a minimum of 24 periods. The on-chip delay counter is inactive. The user must ensure that the oscillator is stable before any operation is attempted.

### 14.4 Status of external pins

The status of the external pins during Idle and Power-down mode is shown in Table 13. If the Power-down mode is activated whilst accessing external Program Memory, the port data that is held in the Special Function Register P2 is restored to Port 2.

If the data is a logic 1, the port pin is held HIGH during the Power-down mode by the strong pull-up transistor 'p1'; see Fig.9(a).

**Table 13** Status of external pins during Idle and Power-down modes

| MODE | MEMORY | ALE | $\overline{\text{PSEN}}$ | $\overline{\text{PWM0}}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 | PORT 4 |
|---|---|---|---|---|---|---|---|---|---|
| Idle | internal | 1 | 1 | active | port data | port data | port data | port data | port data |
| | external | 1 | 1 | active | floating | port data | address | port data | port data |
| Power-down | internal | 0 | 0 | HIGH | port data | port data | port data | port data | port data |
| | external | 0 | 0 | HIGH | floating | port data | port data | port data | port data |

### 14.5 Power Control Register (PCON)

Idle and Power-down modes are activated by software using this SFR. PCON is not bit addressable, the reset value of PCON is 0XX00000B.

**Table 14** Power Control Register (address 87H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SMOD | – | – | WLE | GF1 | GF0 | PD | IDL |

**Table 15** Description of PCON bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | SMOD | **Double Baud rate bit**. When set to a logic 1 the baud rate is doubled when the serial port SIO0 is being used in modes 1, 2 or 3. |
| 6 and 5 | – | Reserved. |
| 4 | WLE | **Watchdog Load Enable**. This flag must be set by software prior to loading the Watchdog Timer (T3). It is cleared when T3 is loaded. |
| 3 and 2 | GF1 and GF0 | **General purpose flag bits**. |
| 1 | PD | **Power-down bit**. Setting this bit activates the Power-down mode. This bit can only be set if input $\overline{\text{EWN}}$ is HIGH. If a logic 1 is written to both PD and IDL at the same time, PD takes precedence. |
| 0 | IDL | **Idle mode bit**. Setting this bit activates the Idle mode. |

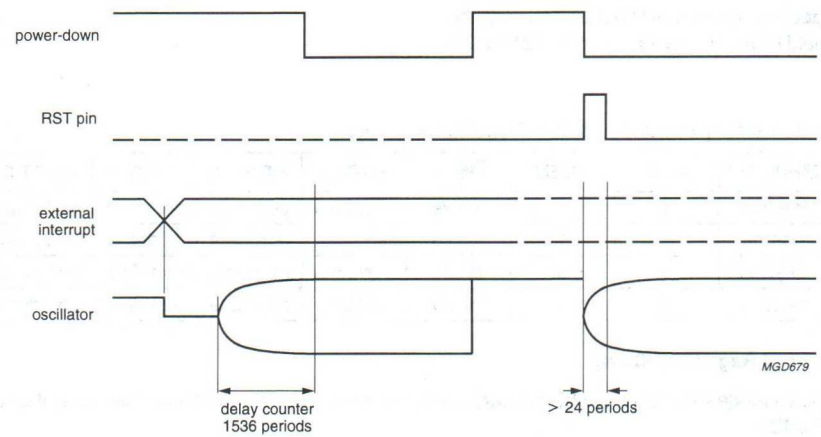



Fig.15 Wake-up operation.

## 15 I²C-BUS SERIAL I/O

The serial port supports the twin line I²C-bus, which consists of a data line (SDA) and a clock line (SCL). These lines also function as the I/O port lines P1.7 and P1.6 respectively.

The system is unique because data transport, clock generation, address recognition and bus control arbitration are all controlled by hardware.

The I²C-bus serial I/O has complete autonomy in byte handling and operates in 4 modes:

- Master transmitter
- Master receiver
- Slave transmitter
- Slave receiver.

These functions are controlled by the Serial Control Register S1CON. S1STA is the Status Register whose contents may also be used as a vector to various service routines. S1DAT is the Data Shift Register and S1ADR is the Slave Address Register. Slave address recognition is performed by on-chip hardware.

Figure 16 is the block diagram of the I²C-bus serial I/O.

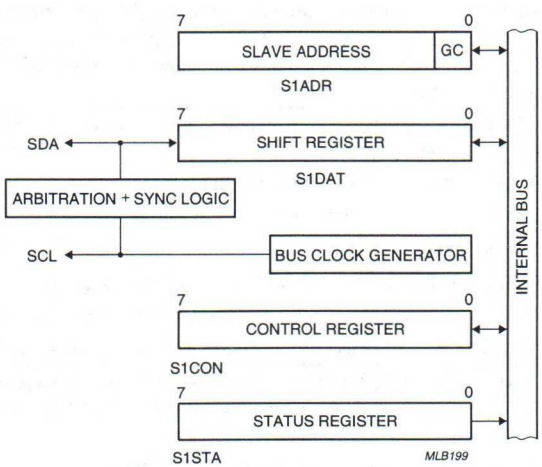



Fig.16 Block diagram of I²C-bus serial I/O.

### 15.1 Serial Control Register (S1CON)

**Table 16** Serial Control Register (SFR address D8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |

**Table 17** Description of S1CON bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | CR2 | This bit along with bits CR1 (S1CON.1) and CR0 (S1CON.0) determines the serial clock frequency when SIO is in the Master mode. See Table 18. |
| 6 | ENS1 | **ENABLE serial I/O**. When ENS1 = 0, the serial I/O is disabled. SDA and SCL outputs are in the high impedance state; P1.6 and P1.7 function as open-drain ports. When ENS1 = 1, the serial I/O is enabled. Output port latches P1.6 and P1.7 must be set to logic 1. |
| 5 | STA | **START flag**. When this bit is set in Slave mode, the SIO hardware checks the status of the $I^2C$-bus and generates a START condition if the bus is free or after the bus becomes free. If STA is set while the SIO is in Master mode, SIO will generate a repeated START condition. |
| 4 | STO | **STOP flag**. With this bit set while in Master mode a STOP condition is generated. When a STOP condition is detected on the $I^2C$-bus, the SIO hardware clears the STO flag. STO may also be set in Slave mode in order to recover from an error condition. In this case no STOP condition is transmitted to the $I^2C$-bus. However, the SIO hardware behaves as if a STOP condition has been received and releases the SDA and SCL. The SIO then switches to the not addressed slave receiver mode. The STOP flag is cleared by the hardware. |
| 3 | SI | **SIO interrupt flag**. This flag is set, and an interrupt is generated, after any of the following events occur:<br>• A start condition is generated in Master mode<br>• Own slave address has been received during AA = 1<br>• The general call address has been received while GC (S1ADR.0) = 1 and AA = 1<br>• A data byte has been received or transmitted in Master mode (even if arbitration is lost)<br>• A data byte has been received or transmitted as selected slave<br>• A Stop or Start condition is received as selected slave receiver or transmitter. |
| 2 | AA | **Assert Acknowledge**. When this bit is set, an acknowledge (low level to SDA) is returned during the acknowledge clock pulse on the SCL line when:<br>• Own slave address is received<br>• General call address is received; GC (S1ADR.0) = 1<br>• A data byte is received while the device is programmed to be a Master Receiver<br>• A data byte is received while the device is a selected Slave Receiver.<br>When this bit is reset, no acknowledge is returned. Consequently, no interrupt is requested when the own slave address or general call address is received. |
| 1 | CR1 | These two bits along with the CR2 (S1CON.7) bit determine the serial clock frequency when SIO is in the Master mode. See Table 18. |
| 0 | CR0 | |

**Table 18** Selection of the serial clock frequency SCL in a Master mode of operation

| CR2 | CR1 | CR0 | $f_{osc}$ DIVISOR | BIT RATE (kHz) AT $f_{osc}$ | | |
|---|---|---|---|---|---|---|
| | | | | 3.58 MHz | 6 MHz | 12 MHz |
| 0 | 0 | 0 | 256 | 14.0 | 23.4 | 46.9 |
| 0 | 0 | 1 | 224 | 16.0 | 26.8 | 53.6 |
| 0 | 1 | 0 | 192 | 18.6 | 31.3 | 62.5 |
| 0 | 1 | 1 | 160 | 22.4 | 37.5 | 75.0 |
| 1 | 0 | 0 | 960 | 3.73 | 6.25 | 12.5 |
| 1 | 0 | 1 | 120 | 29.8 | 50.0 | 100.0 |
| 1 | 1 | 0 | 60 | 59.7 | 100.0 | – |
| 1 | 1 | 1 | not allowed | – | – | – |

### 15.2 Serial Status Register (S1STA)

S1STA is a read-only register.The contents of this register may be used as a vector to a service routine. This optimizes the response time of the software and consequently that of the $I^2C$-bus. The status codes for all possible modes of the $I^2C$-bus interface are given in Tables 21 to 25.

**Table 19** Serial Status Register (address D9H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SC4 | SC3 | SC2 | SC1 | SC0 | 0 | 0 | 0 |

**Table 20** Description of S1STA bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 3 to 7 | SC4 to SC0 | 5-bit status code. |
| 0 to 2 | – | These three bits are always zero. |

**Table 21** MST/TRX mode

| S1STA VALUE | DESCRIPTION |
|---|---|
| 08H | A START condition has been transmitted. |
| 10H | A repeated START condition has been transmitted. |
| 18H | SLA and W have been transmitted, ACK has been received. |
| 20H | SLA and W have been transmitted, $\overline{\text{ACK}}$ received. |
| 28H | DATA of S1DAT has been transmitted, ACK received. |
| 30H | DATA of S1DAT has been transmitted, $\overline{\text{ACK}}$ received. |
| 38H | Arbitration lost in SLA, R/W or DATA. |

**Table 22** MST/REC mode

| S1STA VALUE | DESCRIPTION |
|---|---|
| 08H | A START condition has been transmitted. |
| 10H | A repeated START condition has been transmitted. |
| 38H | Arbitration lost while returning $\overline{ACK}$. |
| 40H | SLA and R have been transmitted, ACK received. |
| 48H | SLA and R have been transmitted, $\overline{ACK}$ received. |
| 50H | DATA has been received, ACK returned. |
| 58H | DATA has been received, $\overline{ACK}$ returned. |

**Table 23** SLV/REC mode

| S1STA VALUE | DESCRIPTION |
|---|---|
| 60H | Own SLA and W have been received, ACK returned. |
| 68H | Arbitration lost in SLA, R/W as MST. Own SLA and W have been received, ACK returned. |
| 70H | General CALL has been received, ACK returned. |
| 78H | Arbitration lost in SLA, R/W as MST. General CALL has been received. |
| 80H | Previously addressed with own SLA. DATA byte received, ACK returned. |
| 88H | Previously addressed with own SLA. DATA byte received, $\overline{ACK}$ returned. |
| 90H | Previously addressed with general CALL. DATA byte has been received, ACK has been returned. |
| 98H | Previously addressed with general CALL. DATA byte has been received, $\overline{ACK}$ has been returned. |
| A0H | A STOP condition or repeated START condition has been received while still addressed as SLV/REC or SLV/TRX. |

**Table 24** SLV/TRX mode

| S1STA VALUE | DESCRIPTION |
|---|---|
| A8H | Own SLA and R have been received, ACK returned. |
| B0H | Arbitration lost in SLA, R/W as MST. Own SLA and R have been received, ACK returned. |
| B8H | DATA byte has been transmitted, ACK received. |
| C0H | DATA byte has been transmitted, $\overline{ACK}$ received. |
| C8H | Last DATA byte has been transmitted (AA = 0), ACK received. |

**Table 25** Miscellaneous.

| S1STA VALUE | DESCRIPTION |
|---|---|
| 00H | Bus error during MST mode or selected SLV mode, due to an erroneous START or STOP condition. |
| F8H | No relevant state information available, SI = 0. |

**Table 26** Symbols used in Tables 21 to 25.

| SYMBOL | DESCRIPTION |
|---|---|
| SLA | 7-bit slave address |
| R | Read bit |
| W | Write bit |
| ACK | Acknowledgement (acknowledge bit is logic 0) |
| $\overline{\text{ACK}}$ | No acknowledgement (acknowledge bit is logic 1) |
| DATA | 8-bit data byte to or from $I^2C$-bus |
| MST | Master |
| SLV | Slave |
| TRX | Transmitter |
| REC | Receiver |

### 15.3 Data Shift Register (S1DAT)

S1DAT contains the serial data to be transmitted or data which has just been received. The MSB (bit 7) is transmitted or received first; i.e. data shifted from right to left.

**Table 27** Data Shift Register (SFR address DAH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| S1DAT.7 | S1DAT.6 | S1DAT.5 | S1DAT.4 | S1DAT.3 | S1DAT.2 | S1DAT.1 | S1DAT.0 |

### 15.4 Address Register (S1ADR)

This 8-bit register may be loaded with the 7-bit slave address to which the controller will respond when programmed as a slave receiver/transmitter.

**Table 28** Address Register (SFR address DBH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SLA6 | SLA5 | SLA4 | SLA3 | SLA2 | SLA1 | SLA0 | GC |

**Table 29** Description of S1ADR bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 to 1 | SLA6 to SLA0 | Own slave address. |
| 0 | GC | This bit is used to determine whether the general call address is recognized. When GC = 0, the general call address is not recognized; when GC = 1, the general call address is recognized. |

## 16 STANDARD SERIAL INTERFACE SIO0: UART

This serial port is full duplex which means that it can transmit and receive simultaneously. It is also receive-buffered and can commence reception of a second byte before a previously received byte has been read from the register. (However, if the first byte has not been read by the time the reception of the second byte is complete, one of the bytes will be lost). The serial port receive and transmit registers are both accessed via the Special Function Register S0BUF. Writing to S0BUF loads the transmit register and reading S0BUF accesses a physically separate receive register.

The serial port can operate in 4 modes:

Mode 0 Serial data enters and exits through RXD. TXD outputs the shift clock. Eight bits are transmitted/received (LSB first). The baud rate is fixed at $1/12 \times f_{osc}$. See Figs 18 and 19.

Mode 1 10 bits are transmitted (through TXD) or received (through RXD): a start bit (logic 0), 8 data bits (LSB first), and a stop bit (logic 1). On receive, the stop bit goes into RB8 in Special Function Register S0CON. The baud rate is variable. See Figs 20 and 21.

Mode 2 11 bits are transmitted (through TXD) or received (through RXD): start bit (logic 0), 8 data bits (LSB first), a programmable 9$^{th}$ data bit, and a stop bit (logic 1). On transmit, the 9$^{th}$ data bit (TB8 in S0CON) can be assigned the value of a logic 0 or logic 1. Or, for example, the parity bit (P, in the PSW) could be moved into TB8. On receive, the 9$^{th}$ data bit goes into RB8 in S0CON, while the stop bit is ignored. The baud rate is programmable to either $1/32$ or $1/64 \times f_{osc}$. See Figs 22 and 23.

Mode 3 11 bits are transmitted (through TXD) or received (through RXD): a start bit (logic 0), 8 data bits (LSB first), a programmable 9$^{th}$ data bit and a stop bit (logic 1). In fact, Mode 3 is the same as Mode 2 in all respects except baud rate. The baud rate in Mode 3 is variable. See Figs 24 and 25.

In all four modes, transmission is initiated by any instruction that uses S0BUF as a destination register. Reception is initiated in Mode 0 by the condition RI = 0 and REN = 1. Reception is initiated in the other modes by the incoming start bit if REN = 1.

### 16.1 Multiprocessor communications

Modes 2 and 3 have a special provision for multiprocessor communications. In these modes, 9 data bits are received. The 9$^{th}$ bit goes into RB8. The following bit is the stop bit. The port can be programmed such that when the stop bit is received, the serial port interrupt will be activated, but only if RB8 = 1. This feature is enabled by setting bit SM2 in S0CON. One use of this feature, in multiprocessor systems, is as follows.

When the master processor wants to transmit a block of data to one of several slaves, it first sends out an address byte which identifies the target slave. An address byte differs from a data byte in that the 9$^{th}$ bit is HIGH in an address byte and LOW in a data byte. With SM2 = 1, no slave will be interrupted by a data byte. An address byte, however, will interrupt all slaves, so that each slave can examine the received byte and see if it is being addressed. The addressed slave will clear its SM2 bit and prepare to receive the data bytes that will be sent. The slaves that were not being addressed leave their SM2 bits set and go on about their business, ignoring the coming data bytes.

SM2 has no effect in Mode 0, and in Mode 1 can be used to check the validity of the stop bit. In a Mode 1 reception, if SM2 = 1, the receive interrupt will not be activated unless a valid stop bit is received.

### 16.2 Serial Port Control and Status Register (S0CON)

The Serial Port Control and Status Register is the Special Function Register S0CON. The register contains not only the mode selection bits, but also the 9th data bit for transmit and receive (TB8 and RB8), and the serial port interrupt bits (TI and RI).

**Table 30** Serial Port Control Register (address 98H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SMO | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |

**Table 31** Description of S0CON bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | SM0 | These bits are used to select the serial port mode; see Table 32. |
| 6 | SM1 | |
| 5 | SM2 | Enables the multiprocessor communication feature in Modes 2 and 3. In these modes, if SM2 = 1, then RI will not be activated if the received 9th data bit (RB8) is a logic 0. In Mode 1, if SM2 = 1, then RI will not be activated unless a valid stop bit was received. In Mode 0, SM2 should be a logic 0. |
| 4 | REN | Enables serial reception and is set by software to enable reception, and cleared by software to disable reception. |
| 3 | TB8 | Is the 9th data bit that will be transmitted in Modes 2 and 3. Set or cleared by software as desired. |
| 2 | RB8 | In Modes 2 and 3, is the 9th data bit received. In Mode 1, if SM2 = 0 then RB8 is the stop bit that was received. In Mode 0, RB8 is not used. |
| 1 | TI | **The transmit interrupt flag**. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit time in the other modes, in any serial transmission. Must be cleared by software. |
| 0 | RI | **The receive interrupt flag**. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial transmission (except see SM2). Must be cleared by software. |

**Table 32** Selection of the serial port modes

| SMO | SM1 | MODE | DESCRIPTION | BAUD RATE |
|---|---|---|---|---|
| 0 | 0 | Mode 0 | Shift register | $1/12 \times f_{osc}$ |
| 0 | 1 | Mode 1 | 8-bit UART | variable |
| 1 | 0 | Mode 2 | 9-bit UART | $1/32$ or $1/64 \times f_{osc}$ |
| 1 | 1 | Mode 3 | 9-bit UART | variable |

### 16.3 Baud rates

The baud rate in Mode 0 is fixed and may be calculated as:

$$\text{Baud Rate} = \frac{f_{osc}}{12}$$

The baud rate in Mode 2 depends on the value of the SMOD bit in Special Function Register PCON and may be calculated as:

$$\text{Baud Rate} = \frac{2^{SMOD}}{64} \times f_{osc}$$

- If SMOD = 0 (value on reset), the baud rate is $1/64 \times f_{osc}$
- If SMOD = 1, the baud rate is $1/32 \times f_{osc}$

The baud rates in Modes 1 and 3 are determined by the Timer 1 or Timer 2 overflow rate.

#### 16.3.1 Using Timer 1 to generate baud rates

When Timer 1 is used as the Baud Rate Generator, the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate and the value of the SMOD bit as follows:

$$\text{Baud Rate} = \frac{2^{SMOD}}{32} \times \text{Timer 1 Overflow Rate.}$$

The Timer 1 interrupt should be disabled in this application. The Timer itself can be configured for either 'timer' or 'counter' operation in any of its 3 running modes. In most typical applications, it is configured for 'timer' operation, in the Auto-reload mode (high nibble of TMOD = 0010B). In this case the baud rate is given by the formula:

$$\text{Baud Rate} = \frac{2^{SMOD}}{32} \times \frac{f_{osc}}{\{12 \times (256 - TH1)\}}$$

By configuring Timer 1 to run as a 16-bit timer (high nibble of TMOD = 0001B), and using the Timer 1 interrupt to do a 16-bit software reload, very low baud rates can be achieved. Table 33 lists commonly used baud rates and how they can be obtained from Timer 1.

**Table 33** Commonly used baud rates generated by Timer 1

| BAUD RATE (kb/s) | $f_{osc}$ (MHz) | SMOD | C/T | TIMER 1 MODE | RELOAD VALUE |
|---|---|---|---|---|---|
| 1000.0[1] | 12.000 | X[2] | X | X | X |
| 375.0[3] | 12.000 | 1 | X | X | X |
| 62.5[4] | 12.000 | 1 | 0 | Mode 2 | FFH |
| 19.2 | 11.059 | 1 | 0 | Mode 2 | FDH |
| 9.6 | 11.059 | 0 | 0 | Mode 2 | FDH |
| 4.8 | 11.059 | 0 | 0 | Mode 2 | FAH |
| 2.4 | 11.059 | 0 | 0 | Mode 2 | F4H |
| 1.2 | 11.059 | 0 | 0 | Mode 2 | E8H |
| 137.5 | 11.986 | 0 | 0 | Mode 2 | 1DH |
| 110.0 | 6.000 | 0 | 0 | Mode 2 | 72H |
| 110.0 | 12.000 | 0 | 0 | Mode 1 | FEEBH |

**Notes**

1. Maximum in Mode 0.
2. X = don't care
3. Maximum in Mode 2.
4. Maximum in Modes 1 and 3.

### 16.3.2 Using Timer 2 to generate baud rates

Timer 2 is selected as a Baud Rate Generator by setting the RTCLK bit in T2CON. The Baud Rate Generator mode is similar to the Auto-reload mode, in that a roll-over in TH2 causes Timer 2 registers to be reloaded with the 16-bit value held in the registers RCAP2H and RCAP2L, which are preset by software. Baud rates in Modes 1 and 3 are determined by Timer 2's overflow rate as specified below.

$$\text{Baud Rate} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

The Timer 2 can be configured for either 'timer' or 'counter' operation. In the most typical applications, it is configured for 'timer' operation ($C/\overline{T2} = 0$). 'Timer' operation is slightly different for Timer 2 when it is being used as a Baud Rate Generator. Normally, as a timer it would increment every machine cycle at a frequency of $1/12 \times f_{osc}$. However, as a Baud Rate Generator it increments every state time at a frequency of $1/2 \times f_{osc}$. In this case the baud rate in Modes 1 and 3 is determined as:

$$\text{Baud Rate} = \frac{f_{osc}}{32 \times \{65536 - (\text{RCAP2H; RCAP2L})\}}$$

Where (RCAP2H; RCAP2L) is the content of registers RCAP2H and RCAP2L taken as a 16-bit unsigned integer.

The Baud Rate Generator mode for Timer 2 is shown in Fig.17. This figure is only valid if RTCLK = 1. At roll-over TH2 does not set the TF2 bit in T2CON and therefore, will not generate an interrupt. Consequently, the Timer 2 interrupt does not need to be disabled when in the Baud Rate Generator mode. If EXEN2 is set, a HIGH-to-LOW transition on T2EX will set the EXF2 bit, also in T2CON, but will not cause a reload from (RCAP2H; RCAP2L) to (TH2, TL2). Therefore, in this mode T2EX may be used as an additional external interrupt.

When Timer 2 is operating as a timer (TR2 = 1), in the Baud Rate Generator mode, registers TH2 and TL2 should not be accessed (read or write). Under these conditions the timer is being incremented every state time and therefore the results of a read or write may not be accurate. The registers RCAP2H and RCAP2L however, may be read but not written to. A write might overlap a reload and cause write and/or reload errors. If a write operation is required, Timer 2 or RCAP2H/RCAP2L should first be turned off by clearing the TR2 bit.

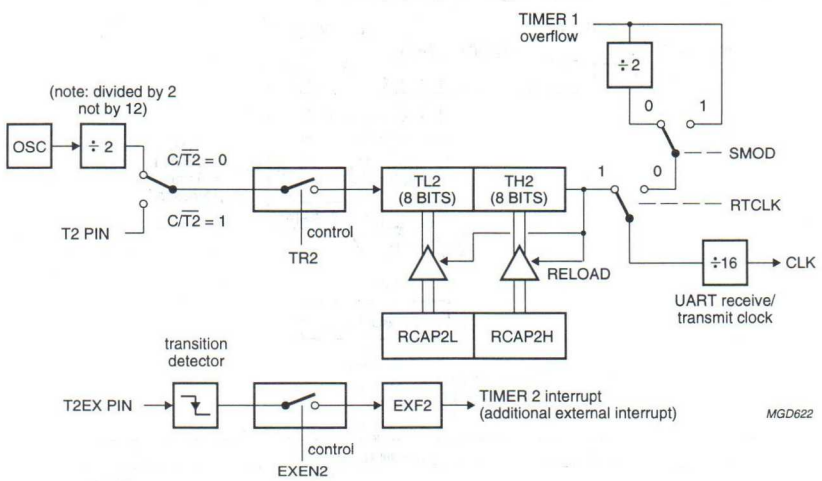



Fig.17 Timer 2 in Baud Rate Generator mode.

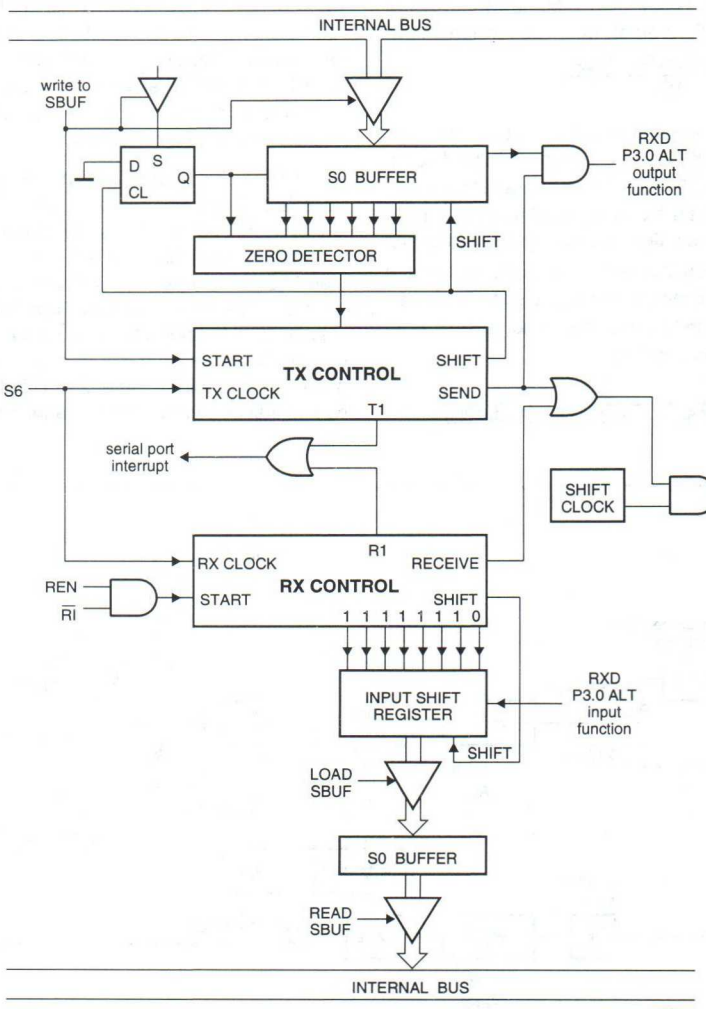



Fig.18 Serial port Mode 0.

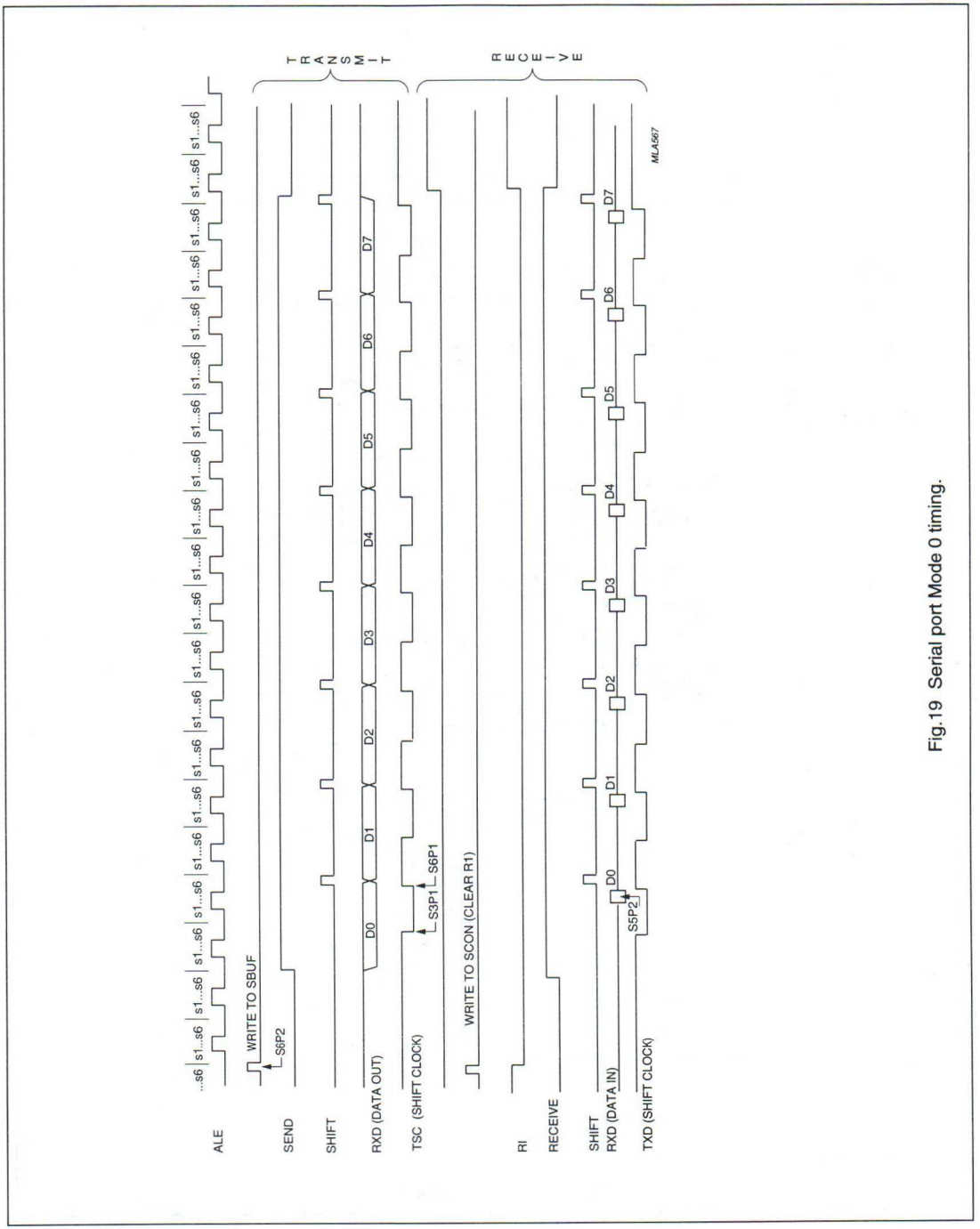



Fig.19 Serial port Mode 0 timing.

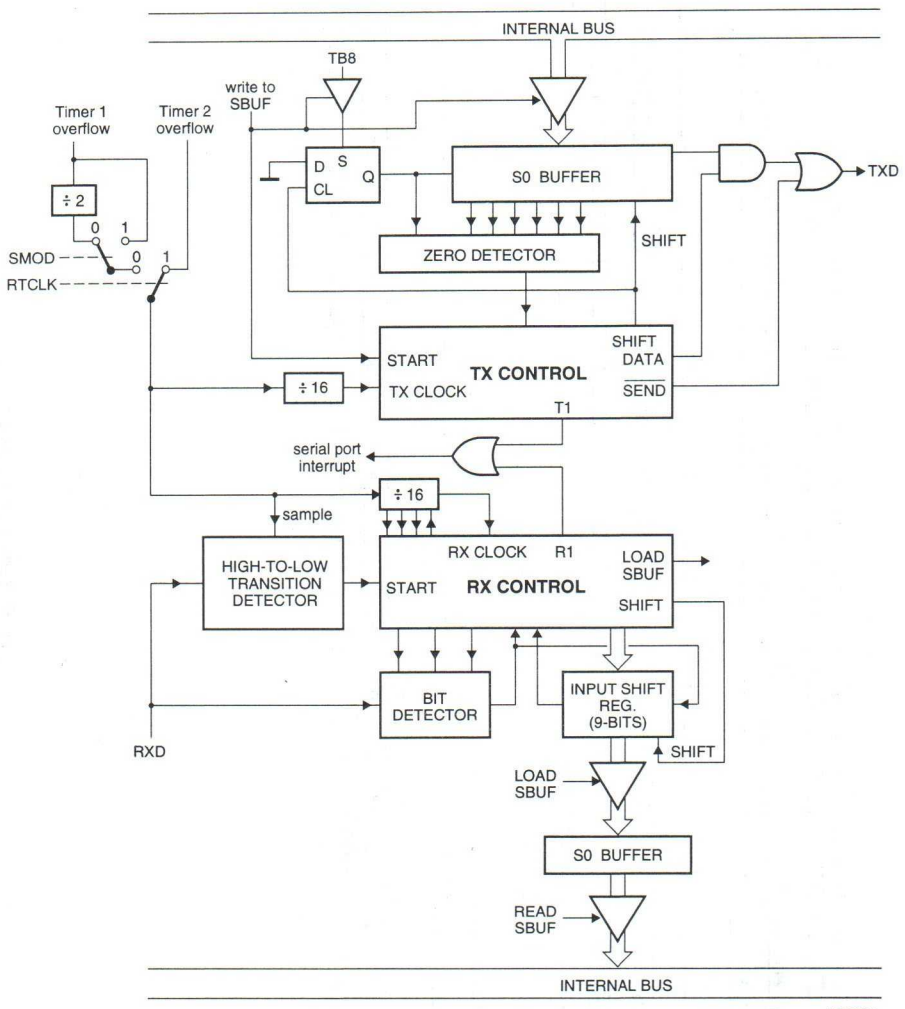



Fig.20 Serial port Mode 1.

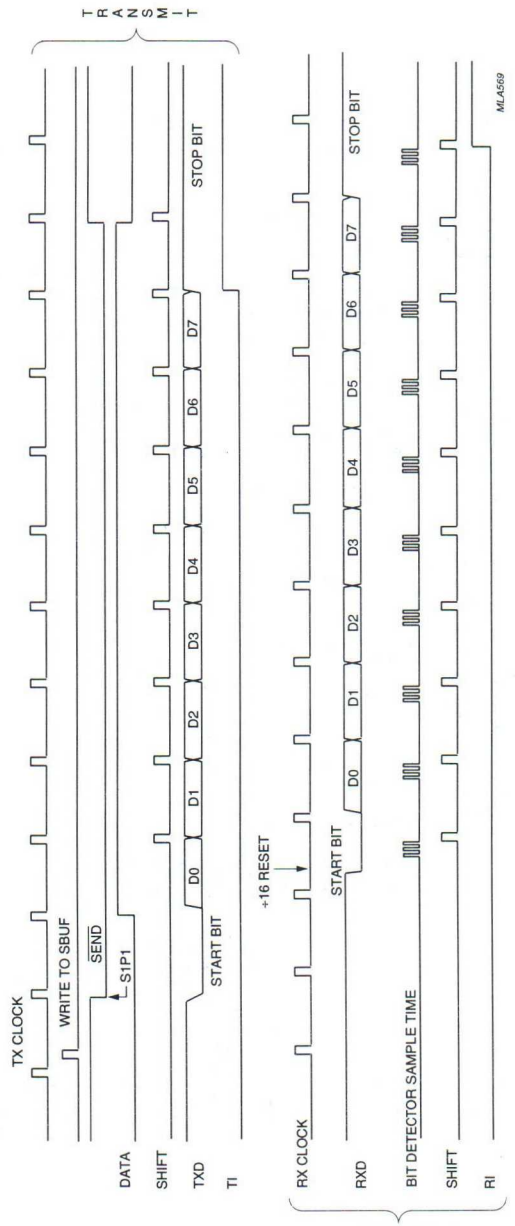



Fig.21 Serial port Mode 1 timing.

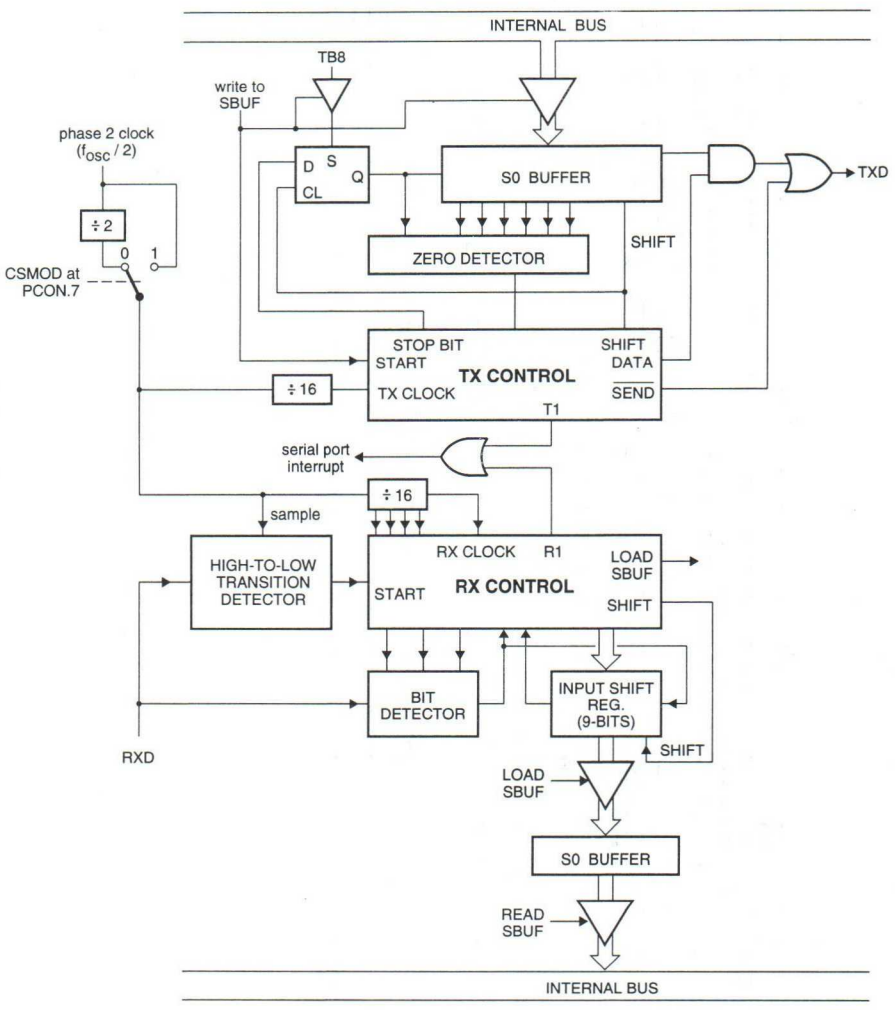



Fig.22 Serial port Mode 2.

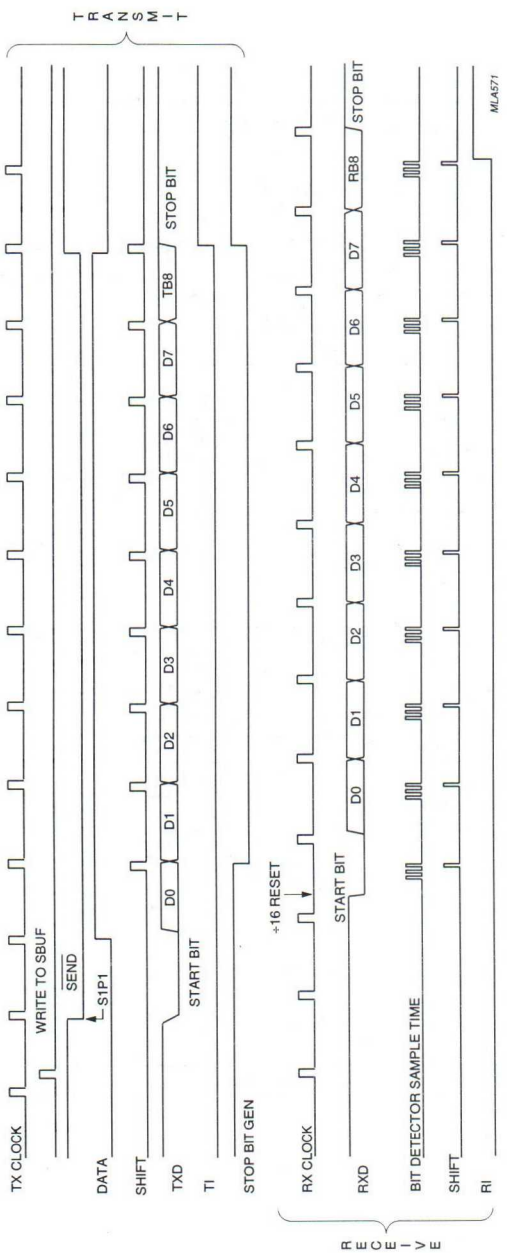



Fig.23 Serial port Mode 2 timing.

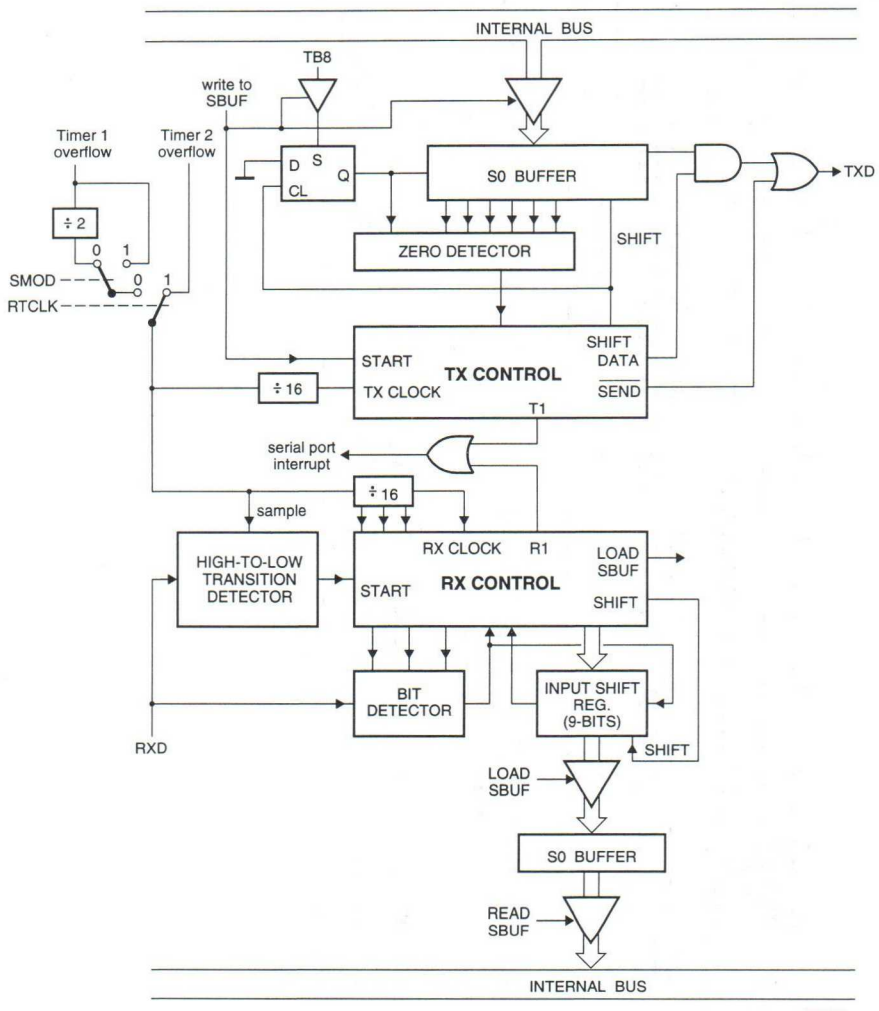



Fig.24 Serial port Mode 3.

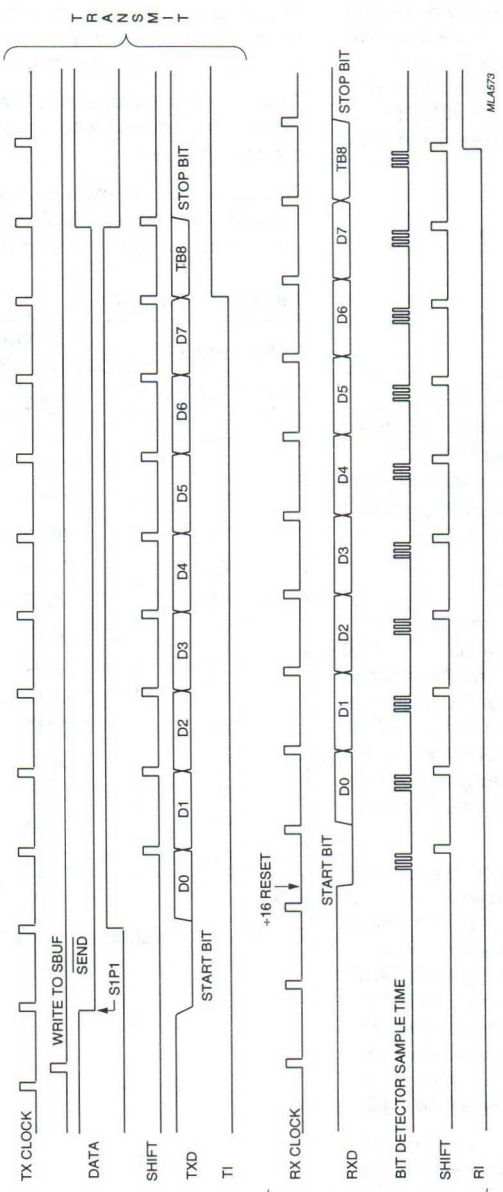



Fig.25 Serial port Mode 3 timing.

## 17 INTERRUPT SYSTEM

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronously to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. The P8xCL580 acknowledges interrupt requests from fifteen sources as follows:

- $\overline{\text{INT0}}$ to $\overline{\text{INT8}}$
- Timer 0, Timer 1 and Timer 2
- $I^2C$-bus serial I/O
- UART
- ADC.

Each interrupt vectors to a separate location in Program Memory for its service routine. Each source can be individually enabled or disabled by corresponding bits in the Interrupt Enable Registers (1EN0 and 1EN1). The priority level is selected via the Interrupt Priority Registers (IP0 and IP1). All enabled sources can be globally disabled or enabled. Figure 26 shows the interrupt system.

### 17.1 External interrupts $\overline{\text{INT2}}$ to $\overline{\text{INT8}}$

Port 1 lines serve an alternative purpose as seven additional interrupts $\overline{\text{INT2}}$ to $\overline{\text{INT8}}$. When enabled, each of these lines may wake-up the device from the Power-down mode. Using the Interrupt Polarity Register (IX1), each pin may be initialized to be either active HIGH or active LOW. IRQ1 is the Interrupt Request Flag Register. If the interrupt is enabled, each flag will be set on an interrupt request but must be cleared by software, i.e. via the interrupt software or when the interrupt is disabled.

A low-priority interrupt can be interrupted by a high-priority interrupt but not by another low-priority interrupt. A high-priority can not be interrupted by any other interrupt. If two interrupt requests of different priority levels are received simultaneously, the request having the highest priority level will be serviced. If interrupt requests of the same priority level are received simultaneously an internal polling sequence determines which request is serviced. Thus within each priority level there is a second priority structure determined by the polling sequence (see Fig.26).

Port 1 interrupts are level sensitive. A Port 1 interrupt will be recognized when a level (HIGH or LOW depending on the Interrupt Polarity Register) on P1.n is held active for at least one machine cycle. The interrupt request is not serviced until the next machine cycle. Figure 27 shows the external interrupt system.

### 17.2 Interrupt priority

Each interrupt source can be set to either a high priority or to a low priority. If interrupts of the same priority are requested simultaneously, the processor will branch to the interrupt polled first, according to Table 34.

A low priority interrupt routine can only be interrupted by a high priority interrupt. A high priority interrupt routine can not be interrupted.

Table 34 shows the interrupt vectors in order of priority. The vector indicates the ROM location where the appropriate interrupt service routine starts.

**Table 34** Interrupt vectors

| SYMBOL[1] | VECTOR ADDRESS (HEX) | SOURCE |
|---|---|---|
| X0 (highest) | 0003 | External 0 |
| S1 | 002B | $I^2C$ port |
| X5 | 0053 | External 5 |
| T0 | 000B | Timer 0 |
| T2 | 0033 | Timer 2 |
| X6 | 005B | External 6 |
| X1 | 0013 | External 1 |
| X2 | 003B | External 2 |
| X7 | 0063 | External 7 |
| T1 | 001B | Timer 1 |
| X3 | 0043 | External 3 |
| X8 | 006B | External 8 |
| SO | 0023 | UART |
| X4 | 004B | External 4 |
| ADC (lowest) | 0073 | ADC |

**Note**

1. X0 has the highest priority; ADC the lowest.

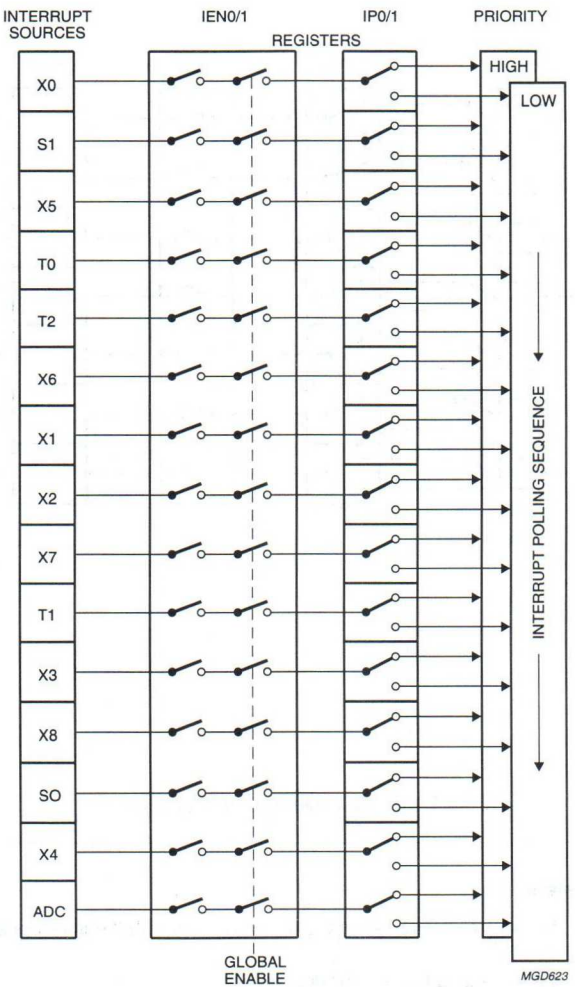



Fig.26 Interrupt system.

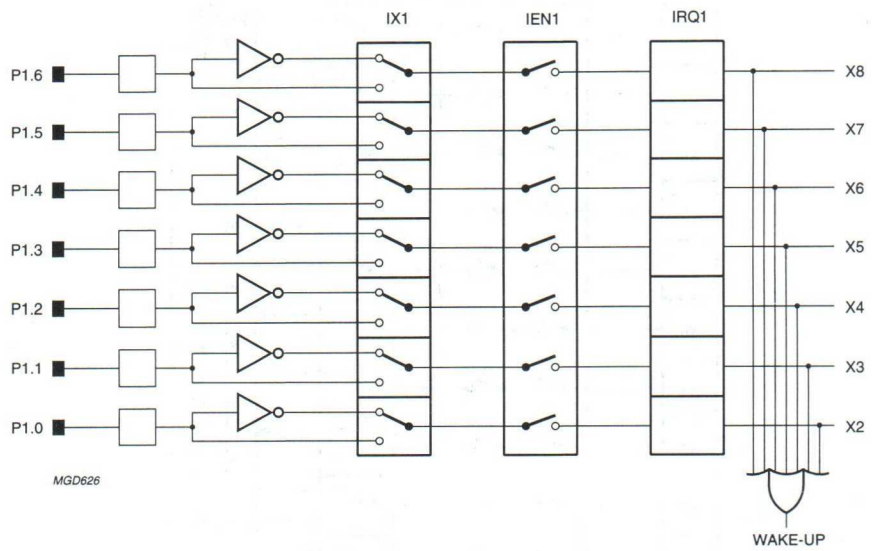



Fig.27 External interrupt configuration.

### 17.3 Interrupt related registers

The registers IEN0, IEN1, IP0, IP1, IX1 and IRQ1 are used in conjunction with the interrupt system.

**Table 35** Special Function Registers related to the interrupt system

| ADDRESS | REGISTER | DESCRIPTION |
|---|---|---|
| A8H | IEN0 | Interrupt Enable Register |
| E8H | IEN1 | Interrupt Enable Register ($\overline{INT2}$ to $\overline{INT8}$) |
| B8H | IP0 | Interrupt Priority Register |
| F8H | IP1 | Interrupt Priority Register ($\overline{INT2}$ to $\overline{INT8}$, ADC) |
| E9H | IX1 | Interrupt Polarity Register |
| C0H | IRQ1 | Interrupt Request Flag Register |

### 17.3.1 Interrupt Enable Register (IEN0)

Bit values: 0 = interrupt disabled; 1 = interrupt enabled.

**Table 36** Interrupt Enable Register (SFR address A8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| EA | ET2 | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 |

**Table 37** Description of IEN0 bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | EA | General enable/disable control. If EA = 0, no interrupt is enabled. If EA = 1, any individually enabled interrupt will be accepted. |
| 6 | ET2 | Enable T2 interrupt. |
| 5 | ES1 | Enable $I^2C$ interrupt. |
| 4 | ES0 | Enable UART SIO interrupt. |
| 3 | ET1 | Enable Timer 1 interrupt (T1). |
| 2 | EX1 | Enable external interrupt 1. |
| 1 | ET0 | Enable Timer 0 interrupt (T0). |
| 0 | EX0 | Enable external interrupt 0. |

### 17.3.2 Interrupt Enable Register (IEN1)

Bit values: 0 = interrupt disabled; 1 = interrupt enabled.

**Table 38** Interrupt Enable Register (SFR address E8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| EAD | EX8 | EX7 | EX6 | EX5 | EX4 | EX3 | EX2 |

**Table 39** Description of IEN1 bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | EAD | Enable ADC interrupt. |
| 6 | EX8 | Enable external interrupt 8. |
| 5 | EX7 | Enable external interrupt 7. |
| 4 | EX7 | Enable external interrupt 6. |
| 3 | EX5 | Enable external interrupt 5. |
| 2 | EX4 | Enable external interrupt 4. |
| 1 | EX3 | Enable external interrupt 3. |
| 0 | EX2 | Enable external interrupt 2. |

### 17.3.3 Interrupt Priority Register (IP0)

Bit values: 0 = low priority; 1 = high priority.

**Table 40** Interrupt Priority Register (SFR address B8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | PT2 | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 |

**Table 41** Description of IP0 bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | – | Reserved. |
| 6 | PT2 | Timer 2 interrupt priority level. |
| 5 | PS1 | $I^2C$ interrupt priority level. |
| 4 | PS0 | UART SIO interrupt priority level. |
| 3 | PT1 | Timer 1 interrupt priority level. |
| 2 | PX1 | External interrupt 1 priority level. |
| 1 | PT0 | Timer 0 interrupt priority level. |
| 0 | PX0 | External interrupt 0 priority level. |

### 17.3.4 Interrupt Priority Register (IP1)

Bit values: 0 = low priority; 1 = high priority.

**Table 42** Interrupt Priority Register (SFR address F8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| PADC | PX8 | PX7 | PX6 | PX5 | PX4 | PX3 | PX2 |

**Table 43** Description of IP1 bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | PADC | ADC interrupt priority level. |
| 6 | PX8 | External interrupt 8 priority level. |
| 5 | PX7 | External interrupt 7 priority level. |
| 4 | PX6 | External interrupt 6 priority level. |
| 3 | PX5 | External interrupt 5 priority level. |
| 2 | PX4 | External interrupt 4 priority level. |
| 1 | PX3 | External interrupt 3 priority level. |
| 0 | PX2 | External interrupt 2 priority level. |

### 17.3.5 Interrupt Polarity Register (IX1)

Writing either a logic 1 or logic 0 to any Interrupt Polarity Register bit sets the polarity level of the corresponding external interrupt to an active HIGH or active LOW respectively.

**Table 44** Interrupt Polarity Register (SFR address E9H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | IL8 | IL7 | IL6 | IL5 | IL4 | IL3 | IL2 |

**Table 45** Description of IX1 bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | – | reserved |
| 6 | IL8 | external interrupt 8 polarity level |
| 5 | IL7 | external interrupt 7 polarity level |
| 4 | IL6 | external interrupt 6 polarity level |
| 3 | IL5 | external interrupt 5 polarity level |
| 2 | IL4 | external interrupt 4 polarity level |
| 1 | IL3 | external interrupt 3 polarity level |
| 0 | IL2 | external interrupt 2 polarity level |

### 17.3.6 Interrupt Request Flag Register (IRQ1)

**Table 46** Interrupt Request Flag Register (SFR address C0H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | IQ8 | IQ7 | IQ6 | IQ5 | IQ4 | IQ3 | IQ2 |

**Table 47** Description of IRQ1 bits

| BIT | SYMBOL | DESCRIPTION |
|---|---|---|
| 7 | – | reserved |
| 6 | IQ8 | external interrupt 8 request flag |
| 5 | IQ7 | external interrupt 7 request flag |
| 4 | IQ6 | external interrupt 6 request flag |
| 3 | IQ5 | external interrupt 5 request flag |
| 2 | IQ4 | external interrupt 4 request flag |
| 1 | IQ3 | external interrupt 3 request flag |
| 0 | IQ2 | external interrupt 2 request flag |

## 18 OSCILLATOR CIRCUITRY

The on-chip oscillator circuitry of the P8xCL580 is a single-stage inverting amplifier biased by an internal feedback resistor. The oscillator circuit is shown in Fig.29. For operation as a standard quartz oscillator, no external components are needed, except for the 32 kHz option. When using external capacitors, ceramic resonators, coils and RC networks to drive the oscillator, five different configurations are supported (see Table 48 and Fig.28).

In the Power-down mode the oscillator is stopped and XTAL1 is pulled HIGH. The oscillator invertor is switched off to ensure no current will flow regardless of the voltage at XTAL1, for configurations (a), (b), (c), (d), (e) and (g) of Fig.28.

To drive the device with an external clock source, apply the external clock signal to XTAL1, and leave XTAL2 to float, as shown in Fig.28(f). There are no requirements on the duty cycle of the external clock, since the input to the internal clocking circuitry is buffered by a flip-flop.

Various oscillator options are provided for optimum on-chip oscillator performance; these are specified in Table 48 and shown in Fig.28. The required option should be stated when ordering.

**Table 48** Oscillator options

| OPTION | APPLICATION |
|---|---|
| Oscillator 1 | For 32 kHz clock applications with external trimmer for frequency adjustment. A 4.7 MΩ bias resistor is needed for use in parallel with the crystal; see Fig.28(c). |
| Oscillator 2 | Low-power, low-frequency operations using LC components; see Fig.28(e). |
| Oscillator 3 | Medium frequency range applications. |
| Oscillator 4 | High frequency range applications. |
| RC oscillator | RC oscillator configuration; see Figs 28(g) and 30. |

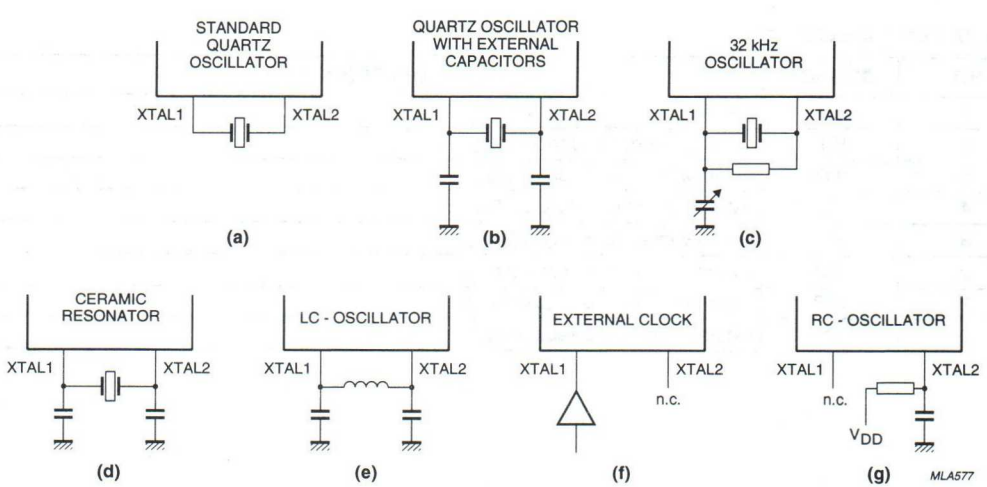



Fig.28 Oscillator configurations.

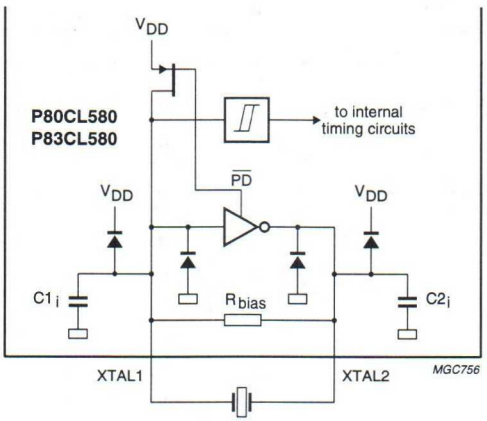
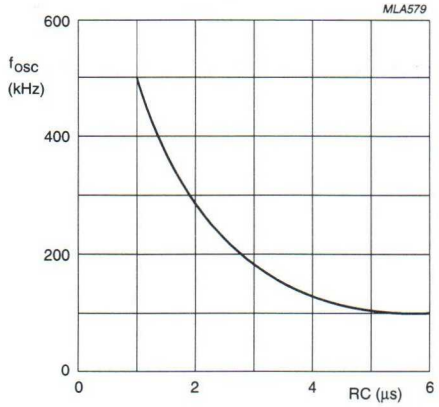



Fig.29 Standard oscillator.




RC oscillator frequency is externally adjustable; 100 kHz ≤ $f_{osc}$ ≤ 500 kHz.

Fig.30 RC oscillator; frequency as a function of RC.

**Table 49** Oscillator type selection guide

| RESONATOR | FREQUENCY (MHz) | OPTION (see Table 48) | C1 EXT. (pF) | | C2 EXT. (pF) | | RESONATOR MAX. SERIES RESISTANCE |
|---|---|---|---|---|---|---|---|
| | | | MIN. | MAX. | MIN. | MAX. | |
| Quartz | 0.032 | Oscillator 1 | 0 | 0 | 5 | 15 | 15 kΩ[1] |
| | 1.0 | Oscillator 2 | 0 | 30 | 0 | 30 | 600 Ω |
| | 3.58 | | 0 | 15 | 0 | 15 | 100 Ω |
| | 4.0 | | 0 | 20 | 0 | 20 | 75 Ω |
| | 6.0 | Oscillator 3 | 0 | 10 | 0 | 10 | 60 Ω |
| | 10.0 | Oscillator 4 | 0 | 15 | 0 | 15 | 60 Ω |
| | 12.0 | | 0 | 10 | 0 | 10 | 40Ω |
| | 16.0 | | 0 | 15 | 0 | 15 | 20 Ω |
| PXE | 0.455 | Oscillator 2 | 40 | 50 | 40 | 50 | 10 Ω |
| | 1.0 | | 15 | 50 | 15 | 50 | 100 Ω |
| | 3.58 | | 0 | 40 | 0 | 40 | 10 Ω |
| | 4.0 | | 0 | 40 | 0 | 40 | 10 Ω |
| | 6.0 | | 0 | 20 | 0 | 20 | 5 Ω |
| | 10.0 | Oscillator 3 | 0 | 15 | 0 | 15 | 6 Ω |
| | 12.0 | Oscillator 4 | 10 | 40 | 10 | 40 | 6 Ω |
| LC | | Oscillator 2 | 20 | 90 | 20 | 90 | 10 µH = 1 Ω<br>100 µH = 5 Ω<br>1 mH = 75 Ω |

**Note**

1. 32 kHz quartz crystals with a series resistance >15 kΩ will reduce the guaranteed supply voltage range to 2.5 to 3.5 V.

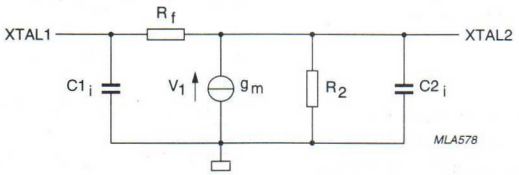



Fig.31 Equivalent circuit diagram.

**Table 50** Oscillator equivalent circuit parameters.
The equivalent circuit data of the internal oscillator compares with that of matched crystals.

| SYMBOL | PARAMETER | OPTION | CONDITION | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|---|---|
| $g_m$ | transconductance | Oscillator 1; 32 kHz | $T_{amb}$ = +25 °C; $V_{DD}$ = 4.5 V | – | 15 | – | µS |
| | | Oscillator 2 | | 200 | 600 | 1000 | µS |
| | | Oscillator 3 | | 400 | 1500 | 4000 | µS |
| | | Oscillator 4 | | 1000 | 4000 | 10000 | µS |
| $C1_i$ | input capacitance | Oscillator 1; 32 kHz | | – | 3.0 | – | pF |
| | | Oscillator 2 | | – | 8.0 | – | pF |
| | | Oscillator 3 | | – | 8.0 | – | pF |
| | | Oscillator 4 | | – | 8.0 | – | pF |
| $C2_i$ | output capacitance | Oscillator 1; 32 kHz | | – | 23 | – | pF |
| | | Oscillator 2 | | – | 8.0 | – | pF |
| | | Oscillator 3 | | – | 8.0 | – | pF |
| | | Oscillator 4 | | – | 8.0 | – | pF |
| R2 | output resistance | Oscillator 1; 32 kHz | | – | 3800 | – | kΩ |
| | | Oscillator 2 | | – | 65 | – | kΩ |
| | | Oscillator 3 | | – | 18 | – | kΩ |
| | | Oscillator 4 | | – | 5.0 | – | kΩ |

## 19 RESET

To initialize the P8xCL580 a reset is performed by either of three methods:

- Applying an external signal to the RST pin
- Via Power-on-reset circuitry
- Watchdog Timer.

A reset leaves the internal registers as shown in Chapter 20. The reset state of the port pins is mask-programmable and can be defined by the user.

### 19.1 External reset using the RST pin

The reset input for the P8xCL580 is RST. A Schmitt trigger is used at the input for noise rejection. The output of the Schmitt trigger is sampled by the reset circuitry every machine cycle. A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods) while the oscillator is running. The CPU responds by executing an internal reset. Port pins adopt their reset state immediately after the RST goes HIGH. During reset, ALE and $\overline{PSEN}$ are held HIGH.

The external reset is asynchronous to the internal clock. The RST pin is sampled during state 5, phase 2 of every machine cycle. After a HIGH is detected at the RST pin, an internal reset is repeated until RST goes LOW. The reset circuitry is also affected by the Watchdog timer; see Section 11.4. The internal RAM is not affected by reset. When $V_{DD}$ is turned on, the RAM contents are indeterminate.

### 19.2 Power-on-reset

The device contains on-chip circuitry which switches the port pins to the customer defined logic level as soon as $V_{DD}$ exceeds 1.3 V; if the mask option 'ON' has been chosen. As soon as the minimum supply voltage is reached, the oscillator will start up. However, to ensure that the oscillator is stable before the controller starts, the clock signals are gated away from the CPU for a further 1536 oscillator periods. During that time the CPU is held in a reset state. A hysteresis of approximately 50 mV at a typical power-on switching level of 1.3 V will ensure correct operation (see Fig.34).

The on-chip Power-on reset circuitry can also be switched off via the mask option 'OFF'. This option reduces the Power-down current to typically 800 nA and can be chosen if external reset circuitry is used. For applications not requiring the internal reset, option 'OFF' should be chosen.

An automatic reset can be obtained by connecting the RST pin to $V_{DD}$ via a 10 µF capacitor. At power-on, the voltage on the RST pin is equal to $V_{DD}$ minus the capacitor voltage, and decreases from $V_{DD}$ as the capacitor charges through the internal resistor ($R_{RST}$) to ground. The larger the capacitor, the more slowly $V_{RST}$ decreases. $V_{RST}$ must remain above the lower threshold of the Schmitt trigger long enough to effect a complete reset. The time required is the oscillator start-up time, plus 2 machine cycles. The Power-on-reset circuitry is shown in Fig.33.

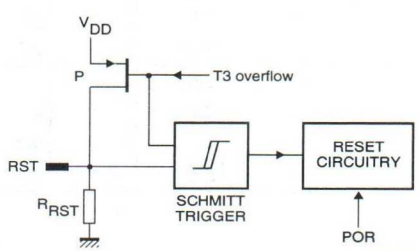



Fig.32 Reset configuration.

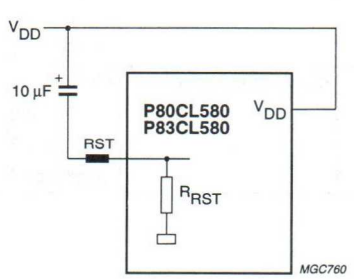



Fig.33 Recommended Power-on-reset circuitry

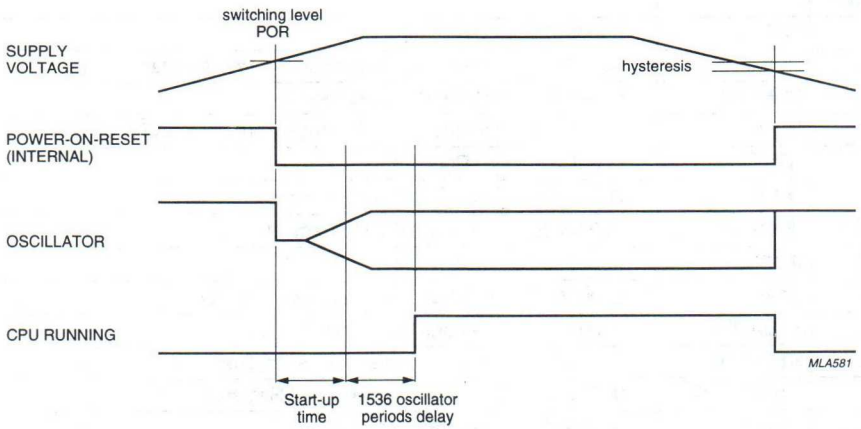



Fig.34 Power-on-reset switching level.

## 20 SPECIAL FUNCTION REGISTERS_OVERVIEW

The P8xCL580 has 40 SFRs available to the user.

| ADDRESS (HEX) | NAME | RESET VALUE (B) | FUNCTION |
|---|---|---|---|
| FF | T3[1] | 00000000 | Watchdog Timer |
| FE | PWMP[1] | 00000000 | Prescaler Frequency Control Register |
| FC | PWM0[1] | 00000000 | Pulse Width Register 0 |
| F8 | IP1[1] | 00000000 | Interrupt Priority Register ($\overline{INT2}$ to $\overline{INT8}$, ADC) |
| F0 | B[2] | 00000000 | B Register |
| E9 | IX1[1] | 00000000 | Interrupt Polarity Register |
| E8 | IEN1[1][2] | 00000000 | Interrupt Enable Register 1 |
| E0 | ACC[2] | 00000000 | Accumulator |
| DB | S1ADR[1] | 00000000 | I²C-bus Slave Address Register |
| DA | S1DAT[1] | 00000000 | I²C-bus Data Shift Register |
| D9 | S1STA[1] | 11111000 | I²C-bus Serial Status Register |
| D8 | S1CON[1][2] | 00000000 | I²C-bus Serial Control Register |
| D0 | PSW[2] | 00000000 | Program Status Word |
| CD | TH2[1] | 00000000 | Timer 2 High byte |
| CC | TL2[1] | 00000000 | Timer 2 Low byte |
| CB | RCAP2H[1] | 00000000 | Timer 2 Reload/Capture Register High byte |
| CA | RCAP2L[1] | 00000000 | Timer 2 Reload/Capture Register Low byte |
| C8 | T2CON[1][2] | 00000000 | Timer/Counter 2 Control Register |
| C5 | ADCH[1] | 11111111 | ADC Result Register |
| C4 | ADCON[1] | X0000000 | ADC Control Register |
| C1 | P4[1] | XXXXXXXX[3] | Digital I/O Port Register 4 |
| C0 | IRQ1[1][2] | 00000000 | Interrupt Request Flag Register |

| ADDRESS (HEX) | NAME | RESET VALUE (B) | FUNCTION |
|---|---|---|---|
| B8 | IP0[2] | X0000000 | Interrupt Priority Register 0 |
| B0 | P3[2] | XXXXXXXX[3] | Digital I/O Port Register 3 |
| A8 | IEN0[2] | 00000000 | Interrupt Enable Register |
| A0 | P2[2] | XXXXXXXX[3] | Digital I/O Port Register 2 |
| 99 | S0BUF | XXXXXXXX | Serial Data Buffer Register 0 |
| 98 | S0CON[2] | 00000000 | Serial Port Control Register 0 |
| 90 | P1[2] | XXXXXXXX[3] | Digital I/O Port Register 1 |
| 8D | TH1 | 00000000 | Timer 1 High byte |
| 8C | TH0 | 00000000 | Timer 0 High byte |
| 8B | TL1 | 00000000 | Timer 1 Low byte |
| 8A | TL0 | 00000000 | Timer 0 Low byte |
| 89 | TMOD | 00000000 | Timer 0 and 1 Mode Control Register |
| 88 | TCON[2] | 00000000 | Timer 0 and 1 Control/External Interrupt Control Register |
| 87 | PCON | 00XX0000 | Power Control Register |
| 83 | DPH | 00000000 | Data Pointer High byte |
| 82 | DPL | 00000000 | Data Pointer Low byte |
| 81 | SP | 00000111 | Stack Pointer |
| 80 | P0[2] | XXXXXXXX[3] | Digital I/O Port Register 0 |

**Notes**

1. P8xCL580 specific SFRs.
2. Bit addressable register.
3. Port reset state determined by the customer.

## 21 INSTRUCTION SET

The P8xCL580 uses a powerful instruction set which permits the expansion of on-chip CPU peripherals and optimizes byte efficiency and execution speed. Assigned opcodes add new high-power operation and permit new addressing modes. The instruction set consists of 49 single-byte, 46 two-byte and 16 three-byte instructions. When using a 12 MHz oscillator, 64 instructions execute in 1 μs and 45 instructions execute in 2 μs. Multiply and divide instructions execute in 4 μs.

For the description of the **Data Addressing modes** and **Hexadecimal opcode cross-reference** see Table 55.

**Table 51** Instruction set description: Arithmetic operations

| MNEMONIC | | DESCRIPTION | BYTES | CYCLES | OPCODE (HEX) |
|---|---|---|---|---|---|
| **Arithmetic operations** | | | | | |
| ADD | A,Rr | Add register to A | 1 | 1 | 2* |
| ADD | A,direct | Add direct byte to A | 2 | 1 | 25 |
| ADD | A,@Ri | Add indirect RAM to A | 1 | 1 | 26, 27 |
| ADD | A,#data | Add immediate data to A | 2 | 1 | 24 |
| ADDC | A,Rr | Add register to A with carry flag | 1 | 1 | 3* |
| ADDC | A,direct | Add direct byte to A with carry flag | 2 | 1 | 35 |
| ADDC | A,@Ri | Add indirect RAM to A with carry flag | 1 | 1 | 36, 37 |
| ADDC | A,#data | Add immediate data to A with carry flag | 2 | 1 | 34 |
| SUBB | A,Rr | Subtract register from A with borrow | 1 | 1 | 9* |
| SUBB | A,direct | Subtract direct byte from A with borrow | 2 | 1 | 95 |
| SUBB | A,@Ri | Subtract indirect RAM from A with borrow | 1 | 1 | 96, 97 |
| SUBB | A,#data | Subtract immediate data from A with borrow | 2 | 1 | 94 |
| INC | A | Increment A | 1 | 1 | 04 |
| INC | Rr | Increment register | 1 | 1 | 0* |
| INC | direct | Increment direct byte | 2 | 1 | 05 |
| INC | @Ri | Increment indirect RAM | 1 | 1 | 06, 07 |
| DEC | A | Decrement A | 1 | 1 | 14 |
| DEC | Rr | Decrement register | 1 | 1 | 1* |
| DEC | direct | Decrement direct byte | 2 | 1 | 15 |
| DEC | @Ri | Decrement indirect RAM | 1 | 1 | 16, 17 |
| INC | DPTR | Increment data pointer | 1 | 2 | A3 |
| MUL | AB | Multiply A and B | 1 | 4 | A4 |
| DIV | AB | Divide A by B | 1 | 4 | 84 |
| DA | A | Decimal adjust A | 1 | 1 | D4 |

**Table 52** Instruction set description: Logic operations

| MNEMONIC | | DESCRIPTION | BYTES | CYCLES | OPCODE (HEX) |
|---|---|---|---|---|---|
| **Logic operations** | | | | | |
| ANL | A,Rr | AND register to A | 1 | 1 | 5* |
| ANL | A,direct | AND direct byte to A | 2 | 1 | 55 |
| ANL | A,@Ri | AND indirect RAM to A | 1 | 1 | 56, 57 |
| ANL | A,#data | AND immediate data to A | 2 | 1 | 54 |
| ANL | direct,A | AND A to direct byte | 2 | 1 | 52 |
| ANL | direct,#data | AND immediate data to direct byte | 3 | 2 | 53 |
| ORL | A,Rr | OR register to A | 1 | 1 | 4* |
| ORL | A,direct | OR direct byte to A | 2 | 1 | 45 |
| ORL | A,@Ri | OR indirect RAM to A | 1 | 1 | 46, 47 |
| ORL | A,#data | OR immediate data to A | 2 | 1 | 44 |
| ORL | direct,A | OR A to direct byte | 2 | 1 | 42 |
| ORL | direct,#data | OR immediate data to direct byte | 3 | 2 | 43 |
| XRL | A,Rr | Exclusive-OR register to A | 1 | 1 | 6* |
| XRL | A,direct | Exclusive-OR direct byte to A | 2 | 1 | 65 |
| XRL | A,@Ri | Exclusive-OR indirect RAM to A | 1 | 1 | 66, 67 |
| XRL | A,#data | Exclusive-OR immediate data to A | 2 | 1 | 64 |
| XRL | direct,A | Exclusive-OR A to direct byte | 2 | 1 | 62 |
| XRL | direct,#data | Exclusive-OR immediate data to direct byte | 3 | 2 | 63 |
| CLR | A | Clear A | 1 | 1 | E4 |
| CPL | A | Complement A | 1 | 1 | F4 |
| RL | A | Rotate A left | 1 | 1 | 23 |
| RLC | A | Rotate A left through the carry flag | 1 | 1 | 33 |
| RR | A | Rotate A right | 1 | 1 | 03 |
| RRC | A | Rotate A right through the carry flag | 1 | 1 | 13 |
| SWAP | A | Swap nibbles within A | 1 | 1 | C4 |

**Table 53** Instruction set description: Data transfer

| MNEMONIC | | DESCRIPTION | BYTES | CYCLES | OPCODE (HEX) |
|---|---|---|---|---|---|
| **Data transfer** | | | | | |
| MOV | A,Rr | Move register to A | 1 | 1 | E* |
| MOV | A,direct (note 1) | Move direct byte to A | 2 | 1 | E5 |
| MOV | A,@Ri | Move indirect RAM to A | 1 | 1 | E6, E7 |
| MOV | A,#data | Move immediate data to A | 2 | 1 | 74 |
| MOV | Rr,A | Move A to register | 1 | 1 | F* |
| MOV | Rr,direct | Move direct byte to register | 2 | 2 | A* |
| MOV | Rr,#data | Move immediate data to register | 2 | 1 | 7* |
| MOV | direct,A | Move A to direct byte | 2 | 1 | F5 |
| MOV | direct,Rr | Move register to direct byte | 2 | 2 | 8* |
| MOV | direct,direct | Move direct byte to direct | 3 | 2 | 85 |
| MOV | direct,@Ri | Move indirect RAM to direct byte | 2 | 2 | 86, 87 |
| MOV | direct,#data | Move immediate data to direct byte | 3 | 2 | 75 |
| MOV | @Ri,A | Move A to indirect RAM | 1 | 1 | F6, F7 |
| MOV | @Ri,direct | Move direct byte to indirect RAM | 2 | 2 | A6, A7 |
| MOV | @Ri,#data | Move immediate data to indirect RAM | 2 | 1 | 76, 77 |
| MOV | DPTR,#data 16 | Load data pointer with a 16-bit constant | 3 | 2 | 90 |
| MOVC | A,@A+DPTR | Move code byte relative to DPTR to A | 1 | 2 | 93 |
| MOVC | A,@A+PC | Move code byte relative to PC to A | 1 | 2 | 83 |
| MOVX | A,@Ri | Move external RAM (8-bit address) to A | 1 | 2 | E2, E3 |
| MOVX | A,@DPTR | Move external RAM (16-bit address) to A | 1 | 2 | E0 |
| MOVX | @Ri,A | Move A to external RAM (8-bit address) | 1 | 2 | F2, F3 |
| MOVX | @DPTR,A | Move A to external RAM (16-bit address) | 1 | 2 | F0 |
| PUSH | direct | Push direct byte onto stack | 2 | 2 | C0 |
| POP | direct | Pop direct byte from stack | 2 | 2 | D0 |
| XCH | A,Rr | Exchange register with A | 1 | 1 | C* |
| XCH | A,direct | Exchange direct byte with A | 2 | 1 | C5 |
| XCH | A,@Ri | Exchange indirect RAM with A | 1 | 1 | C6, C7 |
| XCHD | A,@Ri | Exchange LOW-order digit indirect RAM with A | 1 | 1 | D6, D7 |

**Note**

1. MOV A,ACC is not permitted.

**Table 54** Instruction set description: Boolean variable manipulation, Program and machine control

| MNEMONIC | | DESCRIPTION | BYTES | CYCLES | OPCODE (HEX) |
|---|---|---|---|---|---|
| **Boolean variable manipulation** | | | | | |
| CLR | C | Clear carry flag | 1 | 1 | C3 |
| CLR | bit | Clear direct bit | 2 | 1 | C2 |
| SETB | C | Set carry flag | 1 | 1 | D3 |
| SETB | bit | Set direct bit | 2 | 1 | D2 |
| CPL | C | Complement carry flag | 1 | 1 | B3 |
| CPL | bit | Complement direct bit | 2 | 1 | B2 |
| ANL | C,bit | AND direct bit to carry flag | 2 | 2 | 82 |
| ANL | C,/bit | AND complement of direct bit to carry flag | 2 | 2 | B0 |
| ORL | C,bit | OR direct bit to carry flag | 2 | 2 | 72 |
| ORL | C,/bit | OR complement of direct bit to carry flag | 2 | 2 | A0 |
| MOV | C,bit | Move direct bit to carry flag | 2 | 1 | A2 |
| MOV | bit,C | Move carry flag to direct bit | 2 | 2 | 92 |
| **Program and machine control** | | | | | |
| ACALL | addr11 | Absolute subroutine call | 2 | 2 | •1 |
| LCALL | addr16 | Long subroutine call | 3 | 2 | 12 |
| RET | | Return from subroutine | 1 | 2 | 22 |
| RETI | | Return from interrupt | 1 | 2 | 32 |
| AJMP | addr11 | Absolute jump | 2 | 2 | ♦1 |
| LJMP | addr16 | Long jump | 3 | 2 | 02 |
| SJMP | rel | Short jump (relative address) | 2 | 2 | 80 |
| JMP | @A+DPTR | Jump indirect relative to the DPTR | 1 | 2 | 73 |
| JZ | rel | Jump if A is zero | 2 | 2 | 60 |
| JNZ | rel | Jump if A is not zero | 2 | 2 | 70 |
| JC | rel | Jump if carry flag is set | 2 | 2 | 40 |
| JNC | rel | Jump if carry flag is not set | 2 | 2 | 50 |
| JB | bit,rel | Jump if direct bit is set | 3 | 2 | 20 |
| JNB | bit,rel | Jump if direct bit is not set | 3 | 2 | 30 |
| JBC | bit,rel | Jump if direct bit is set and clear bit | 3 | 2 | 10 |
| CJNE | A,direct,rel | Compare direct to A and jump if not equal | 3 | 2 | B5 |
| CJNE | A,#data,rel | Compare immediate to A and jump if not equal | 3 | 2 | B4 |
| CJNE | Rr,#data,rel | Compare immediate to register and jump if not equal | 3 | 2 | B* |
| CJNE | @Ri,#data,rel | Compare immediate to indirect and jump if not equal | 3 | 2 | B6, B7 |
| DJNZ | Rr,rel | Decrement register and jump if not zero | 2 | 2 | D* |
| DJNZ | direct,rel | Decrement direct and jump if not zero | 3 | 2 | D5 |
| NOP | | No operation | 1 | 1 | 00 |

**Table 55** Description of the mnemonics in the Instruction set

| MNEMONIC | DESCRIPTION |
|---|---|
| **Data addressing modes** | |
| Rr | Working register R0-R7. |
| direct | 128 internal RAM locations and any special function register (SFR). |
| @Ri | Indirect internal RAM location addressed by register R0 or R1 of the actual register bank. |
| #data | 8-bit constant included in instruction. |
| #data 16 | 16-bit constant included as bytes 2 and 3 of instruction. |
| bit | Direct addressed bit in internal RAM or SFR. |
| addr16 | 16-bit destination address. Used by LCALL and LJMP.<br>The branch will be anywhere within the 64 kbytes Program Memory address space. |
| addr11 | 111-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 kbytes page of Program Memory as the first byte of the following instruction. |
| rel | Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps.<br>Range is –128 to +127 bytes relative to first byte of the following instruction. |
| **Hexadecimal opcode cross-reference** | |
| * | 8, 9, A, B, C, D, E, F. |
| • | 1, 3, 5, 7, 9, B, D, F. |
| ♦ | 0, 2, 4, 6, 8, A, C, E. |

**Table 56** Instruction map

| First hexadecimal character of opcode ↓ | ← Second hexadecimal character of opcode → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | NOP | AJMP addr11 | LJMP addr16 | RR A | INC A | INC direct | INC @Ri 0 | INC @Ri 1 | INC Rr 0 | INC Rr 1 | INC Rr 2 | INC Rr 3 | INC Rr 4 | INC Rr 5 | INC Rr 6 | INC Rr 7 |
| 1 | JBC bit,rel | ACALL addr11 | LCALL addr16 | RRC A | DEC A | DEC direct | DEC @Ri 0 | DEC @Ri 1 | DEC Rr 0 | DEC Rr 1 | DEC Rr 2 | DEC Rr 3 | DEC Rr 4 | DEC Rr 5 | DEC Rr 6 | DEC Rr 7 |
| 2 | JB bit,rel | AJMP addr11 | RET | RL A | ADD A,#data | ADD A,direct | ADD A,@Ri 0 | ADD A,@Ri 1 | ADD A,Rr 0 | ADD A,Rr 1 | ADD A,Rr 2 | ADD A,Rr 3 | ADD A,Rr 4 | ADD A,Rr 5 | ADD A,Rr 6 | ADD A,Rr 7 |
| 3 | JNB bit,rel | ACALL addr11 | RETI | RLC A | ADDC A,#data | ADDC A,direct | ADDC A,@Ri 0 | ADDC A,@Ri 1 | ADDC A,Rr 0 | ADDC A,Rr 1 | ADDC A,Rr 2 | ADDC A,Rr 3 | ADDC A,Rr 4 | ADDC A,Rr 5 | ADDC A,Rr 6 | ADDC A,Rr 7 |
| 4 | JC rel | AJMP addr11 | ORL direct,A | ORL direct,#data | ORL A,#data | ORL A,direct | ORL A,@Ri 0 | ORL A,@Ri 1 | ORL A,Rr 0 | ORL A,Rr 1 | ORL A,Rr 2 | ORL A,Rr 3 | ORL A,Rr 4 | ORL A,Rr 5 | ORL A,Rr 6 | ORL A,Rr 7 |
| 5 | JNC rel | ACALL addr11 | ANL direct,A | ANL direct,#data | ANL A,#data | ANL A,direct | ANL A,@Ri 0 | ANL A,@Ri 1 | ANL A,Rr 0 | ANL A,Rr 1 | ANL A,Rr 2 | ANL A,Rr 3 | ANL A,Rr 4 | ANL A,Rr 5 | ANL A,Rr 6 | ANL A,Rr 7 |
| 6 | JZ rel | AJMP addr11 | XRL direct,A | XRL direct,#data | XRL A,#data | XRL A,direct | XRL A,@Ri 0 | XRL A,@Ri 1 | XRL A,Rr 0 | XRL A,Rr 1 | XRL A,Rr 2 | XRL A,Rr 3 | XRL A,Rr 4 | XRL A,Rr 5 | XRL A,Rr 6 | XRL A,Rr 7 |
| 7 | JNZ rel | ACALL addr11 | ORL C,bit | JMP @A+DPTR | MOV A,#data | MOV direct,#data | MOV @Ri,#data 0 | MOV @Ri,#data 1 | MOV Rr,#data 0 | MOV Rr,#data 1 | MOV Rr,#data 2 | MOV Rr,#data 3 | MOV Rr,#data 4 | MOV Rr,#data 5 | MOV Rr,#data 6 | MOV Rr,#data 7 |
| 8 | SJMP rel | AJMP addr11 | ANL C,bit | MOVC A,@A+PC | DIV AB | MOV direct,direct | MOV direct,@Ri 0 | MOV direct,@Ri 1 | MOV direct,Rr 0 | MOV direct,Rr 1 | MOV direct,Rr 2 | MOV direct,Rr 3 | MOV direct,Rr 4 | MOV direct,Rr 5 | MOV direct,Rr 6 | MOV direct,Rr 7 |
| 9 | MOV DTPR,#data16 | ACALL addr11 | MOV bit,C | MOVC A,@A+DPTR | SUBB A,#data | SUBB A,direct | SUBB A,@Ri 0 | SUBB A,@Ri 1 | SUB A,Rr 0 | SUB A,Rr 1 | SUB A,Rr 2 | SUB A,Rr 3 | SUB A,Rr 4 | SUB A,Rr 5 | SUB A,Rr 6 | SUB A,Rr 7 |
| A | ORL C,/bit | AJMP addr11 | MOV bit,C | INC DPTR | MUL AB | | MOV @Ri,direct 0 | MOV @Ri,direct 1 | MOV Rr,direct 0 | MOV Rr,direct 1 | MOV Rr,direct 2 | MOV Rr,direct 3 | MOV Rr,direct 4 | MOV Rr,direct 5 | MOV Rr,direct 6 | MOV Rr,direct 7 |
| B | ANL C,/bit | ACALL addr11 | CPL bit | CPL C | CJNE A,#data,rel | CJNE A,direct,rel | CJNE @Ri,#data,rel 0 | CJNE @Ri,#data,rel 1 | CJNE Rr,#data,rel 0 | CJNE Rr,#data,rel 1 | CJNE Rr,#data,rel 2 | CJNE Rr,#data,rel 3 | CJNE Rr,#data,rel 4 | CJNE Rr,#data,rel 5 | CJNE Rr,#data,rel 6 | CJNE Rr,#data,rel 7 |
| C | PUSH direct | AJMP addr11 | CLR bit | CLR C | SWAP A | XCH A,direct | XCH A,@Ri 0 | XCH A,@Ri 1 | XCH A,Rr 0 | XCH A,Rr 1 | XCH A,Rr 2 | XCH A,Rr 3 | XCH A,Rr 4 | XCH A,Rr 5 | XCH A,Rr 6 | XCH A,Rr 7 |
| D | POP direct | ACALL addr11 | SETB bit | SETB C | DA A | DJNZ direct,rel | XCHD A,@Ri 0 | XCHD A,@Ri 1 | DJNZ Rr,rel 0 | DJNZ Rr,rel 1 | DJNZ Rr,rel 2 | DJNZ Rr,rel 3 | DJNZ Rr,rel 4 | DJNZ Rr,rel 5 | DJNZ Rr,rel 6 | DJNZ Rr,rel 7 |
| E | MOVX A,@DTPR | AJMP addr11 | MOVX A,@Ri 0 | MOVX A,@Ri 1 | CLR A | MOV A,direct [1] | MOV A,@Ri 0 | MOV A,@Ri 1 | MOV A,Rr 0 | MOV A,Rr 1 | MOV A,Rr 2 | MOV A,Rr 3 | MOV A,Rr 4 | MOV A,Rr 5 | MOV A,Rr 6 | MOV A,Rr 7 |
| F | MOVX @DTPR,A | ACALL addr11 | MOVX @Ri,A 0 | MOVX @Ri,A 1 | CPL A | MOV direct,A | MOV @Ri,A 0 | MOV @Ri,A 1 | MOV Rr,A 0 | MOV Rr,A 1 | MOV Rr,A 2 | MOV Rr,A 3 | MOV Rr,A 4 | MOV Rr,A 5 | MOV Rr,A 6 | MOV Rr,A 7 |

**Note**

1. MOV A, ACC is not a valid instruction.

## 22 LIMITING VALUES

In accordance with the Absolute Maximum Rating System (IEC 134).

| SYMBOL | PARAMETER | MIN. | MAX. | UNIT |
|---|---|---|---|---|
| $V_{DD}$ | supply voltage | −0.5 | +6.5 | V |
| $V_I$ | input voltage on any pin with respect to ground ($V_{SS}$) | −0.5 | $V_{DD}$ + 0.5 | V |
| $I_I$, $I_O$ | DC current on any input or output | – | 5.0 | mA |
| $P_{tot}$ | total power dissipation | – | 300 | mW |
| $T_{stg}$ | storage temperature | −65 | +150 | °C |
| $T_{amb}$ | operating ambient temperature | −40 | +85 | °C |
| $T_j$ | operating junction temperature | – | +125 | °C |

## 23 DC CHARACTERISTICS

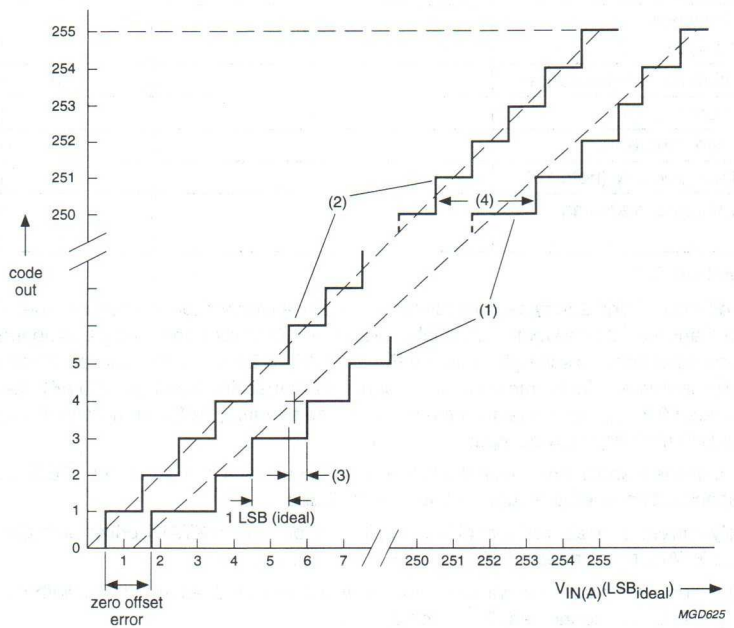
$V_{DD}$ = 1.8 to 6 V; $V_{SS}$ = 0 V; $T_{amb}$ = −25 to +55 °C; see notes 1 and 2; all voltages are with respect to $V_{SS}$ unless otherwise specified.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|---|
| **Supply** | | | | | | |
| $V_{DD}$ | supply voltage<br>operating<br>RAM retention voltage in Power-down mode | | <br>2.5<br>1.0 | <br>–<br>– | <br>6.0<br>6.0 | <br>V<br>V |
| $I_{DD}$ | supply current operating | $V_{DD}$ = 5 V; $f_{CLK}$ = 12 MHz; note 3 | – | – | 27.0 | mA |
| | | $V_{DD}$ = 3 V; $f_{CLK}$ = 3.58 MHz; note 3 | – | – | 5.0 | µA |
| $I_{DD(ID)}$ | supply current Idle mode | $V_{DD}$ = 5 V; $f_{CLK}$ = 12 MHz; note 4 | – | – | 10.0 | mA |
| | | $V_{DD}$ = 3 V; $f_{CLK}$ = 3.58 MHz; note 4 | – | – | 3.0 | mA |
| $I_{DD(PD)}$ | Power-down current | $V_{DD}$ = 1.8 V; $T_{amb}$ = 25°C; note 5 | – | – | 10 | µA |
| **Inputs** (note 6) | | | | | | |
| $V_{IL}$ | LOW level input voltage | | $V_{SS}$ | – | 0.3$V_{DD}$ | V |
| $V_{IH}$ | HIGH level input voltage | | 0.7$V_{DD}$ | – | $V_{DD}$ | V |
| $I_{LI}$ | input leakage current (Port 0; $\overline{EA}$) | $V_{SS}$ < $V_I$ < $V_{DD}$ | – | – | ±10 | µA |
| **Outputs** | | | | | | |
| $I_{OL}$ | LOW level output current (except SDA; SCL) | $V_{DD}$ = 5 V; $V_{OL}$ = 0.4 V | 1.6 | – | – | mA |
| | | $V_{DD}$ = 2.5 V; $V_{OL}$ = 0.4 V | 0.7 | – | – | mA |
| | LOW level output current SDA; SCL | $V_{DD}$ = 5 V; $V_{OL}$ = 0.4 V | 3.0 | – | – | mA |
| | LOW level output current $\overline{PWM0}$ | $V_{DD}$ = 5 V; $V_{OL}$ = 0.4 V | 3.0 | – | – | mA |
| | | $V_{DD}$ = 2.5 V; $V_{OL}$ = 0.4 V | 1.6 | – | – | mA |
| $I_{OH}$ | HIGH level output current $\overline{PWM0}$ | $V_{DD}$ = 5 V; $V_{OH}$ = $V_{DD}$ − 0.4 V | −3.2 | – | – | mA |
| | | $V_{DD}$ = 2.5 V; $V_{OH}$ = $V_{DD}$ − 0.4 V | −1.6 | – | – | mA |
| $I_{OH}$ | HIGH level output current (push-pull options only) | $V_{DD}$ = 5 V; $V_{OH}$ = $V_{DD}$ − 0.4 V | −1.6 | – | – | mA |
| | | $V_{DD}$ = 2.5 V; $V_{OH}$ = $V_{DD}$ − 0.4 V | −0.7 | – | – | mA |

| SYMBOL | PARAMETER | CONDITIONS | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|---|
| $I_{IL}$ | input current logic 0 | $V_{DD}$ = 5 V; $V_{IN}$ = 0.4 V | – | – | –100 | µA |
| | | $V_{DD}$ = 2.5 V; $V_{IN}$ = 0.4 V | – | – | –50 | µA |
| $I_{ITL}$ | input current logic 0; HIGH-to-LOW transition | $V_{DD}$ = 5 V; $V_{IN}$ = 0.5$V_{DD}$ | – | – | –1.0 | mA |
| | | $V_{DD}$ = 2.5 V; $V_{IN}$ = 0.5$V_{DD}$ | – | – | –500 | µA |
| $R_{RST}$ | RST pull-down resistor | | 10 | – | 200 | kΩ |
| **Analog inputs** (note 7) | | | | | | |
| $V_{IN(A)}$ | analog input voltage | | $V_{SSA}$ | – | $V_{DD}$ | mA |
| $V_{ref(p)(A)}$ | reference voltage | | 2.7 | – | $V_{DD}$ | mA |
| $R_{ref}$ | resistance between $V_{ref(p)(A)}$ and $V_{SSA}$ | | 25 | – | 100 | kΩ |
| $C_{AIN}$ | analog on-chip input capacitance | | – | 3 | – | pF |
| $A_e$ | absolute error (note 8) | | – | – | ±1 | LSB |
| $OS_e$ | zero-offset error (note 9) | | – | – | ±1 | LSB |
| $DL_e$ | differential non-linearity (note 10) | | – | – | ±1 | LSB |
| $M_{ctc}$ | channel-to-channel matching (note 11) | | – | – | ±½ | LSB |

**Notes to the DC characteristics**

1. Capacitive loading on Ports 0 and 2 may cause spurious noise pulses to be superimposed on the LOW level output voltage of ALE, Port 1 and Port 3 pins when these make a HIGH-to-LOW transition during bus operations. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make HIGH-to-LOW transitions during bus operations. In the most adverse conditions (capacitive loading > 100 pF), the noise pulse on the ALE line may exceed 0.8 V. In such events it may be required to qualify ALE with a Schmitt trigger, or use an address latch with a Schmitt trigger strobe input.
2. Capacitive loading on Ports 0 and 2 may cause the HIGH level output voltage on ALE and $\overline{PSEN}$ to momentarily fall below the 0.9$V_{DD}$ specification when the address bits are stabilizing.
3. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$; $V_{IH} = V_{DD}$; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = $V_{DD}$.
4. The Idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$; $V_{IH} = V_{DD}$; XTAL2 not connected; $\overline{EA}$ = Port 0 = $V_{DD}$.
5. The power-down current is measured with all output pins disconnected; XTAL1 not connected; $\overline{EA}$ = Port 0 = $V_{DD}$; RST = $V_{SS}$.
6. The input threshold voltage of P1.6/SCL and P1.7/SDA meet the $I^2C$-bus specification. Therefore, an input voltage below 0.3$V_{DD}$ will be recognized as a logic 0 and an input voltage above 0.7$V_{DD}$ will be recognized as a logic 1.
7. $V_{DD}$ = 2.7 to 6 V; $V_{SS}$ = 0 V; $V_{SSA}$ = 0 V; $V_{ref(p)(A)} = V_{DD}$; $T_{amb}$ = –40 to +85 °C, unless otherwise specified. $f_{xtal(min)}$ = 250 kHz.
8. Absolute error: the maximum difference between actual and ideal code transitions. Absolute error accounts for all deviations of an actual converter from an ideal converter.
9. Zero-offset error: the difference between the actual and ideal input voltage corresponding to the first actual code transition.
10. Differential non-linearity: the difference between the actual and ideal code widths.
11. Channel-to-channel matching: the difference between corresponding code transitions of actual characteristics taken from different channels under the same temperature, voltage and frequency conditions. Not tested, but verified on sampling basis.




(1) Example of an actual transfer curve.
(2) The ideal transfer curve.
(3) Differential non-linearity ($DL_e$).
(4) Absolute error.

Fig.35 Analog-to-digital conversion characteristics.

## 24 AC CHARACTERISTICS

$V_{DD}$ = 5 V; $V_{SS}$ = 0 V; $T_{amb}$ = −40 to +85 °C; $C_L$ = 50 pF for Port 0, ALE and $\overline{PSEN}$; $C_L$ = 40 pF for all other outputs unless specified; $t_{CLK} = 1/f_{CLK}$.

| SYMBOL | PARAMETER | $f_{osc}$ = 12 MHz | | $f_{osc}$ = VARIABLE | | UNIT |
|---|---|---|---|---|---|---|
| | | MIN. | MAX. | MIN. | MAX. | |
| **Program Memory** (Fig.36) | | | | | | |
| $t_{LHLL}$ | ALE pulse width | 127 | – | $2t_{CLK} - 40$ | – | ns |
| $t_{AVLL}$ | address valid to ALE LOW | 43 | – | $t_{CLK} - 40$ | – | ns |
| $t_{LLAX}$ | address hold after ALE LOW | 48 | – | $t_{CLK} - 35$ | – | ns |
| $t_{LLIV}$ | ALE LOW to valid instruction in | – | 233 | – | $4t_{CLK} - 100$ | ns |
| $t_{LLPL}$ | ALE LOW to $\overline{PSEN}$ LOW | 58 | – | $t_{CLK} - 25$ | – | ns |
| $t_{PLPH}$ | $\overline{PSEN}$ pulse width | 215 | – | $3t_{CLK} - 35$ | – | ns |
| $t_{PLIV}$ | $\overline{PSEN}$ LOW to valid instruction in | – | 125 | – | $3t_{CLK} - 125$ | ns |
| $t_{PXIX}$ | input instruction hold after $\overline{PSEN}$ | 0 | – | 0 | – | ns |
| $t_{PXIZ}$ | input instruction float after $\overline{PSEN}$ | – | 63 | – | $t_{CLK} - 20$ | ns |
| $t_{PXAV}$ | $\overline{PSEN}$ to address valid | 75 | – | $t_{CLK} - 8$ | – | ns |
| $t_{AVIV}$ | address to valid instruction in | – | 302 | – | $5t_{CLK} - 115$ | ns |
| $t_{PLAZ}$ | $\overline{PSEN}$ LOW to address float | 12 | – | 0 | – | ns |
| **External Data Memory** (Figs 37 and 38) | | | | | | |
| $t_{RLRH}$ | $\overline{RD}$ pulse width | 400 | – | $6t_{CLK} - 100$ | – | ns |
| $t_{WLWH}$ | $\overline{WR}$ pulse width | 400 | – | $6t_{CLK} - 100$ | – | ns |
| $t_{LLAX}$ | address hold after ALE LOW | 48 | – | $t_{CLK} - 35$ | – | ns |
| $t_{RLDV}$ | $\overline{RD}$ LOW to valid data in | – | 150 | – | $5t_{CLK} - 165$ | ns |
| $t_{RHDZ}$ | data float after $\overline{RD}$ | – | 97 | – | $2t_{CLK} - 70$ | ns |
| $t_{LLDV}$ | ALE LOW to valid data in | – | 517 | | $8t_{CLK} - 150$ | ns |
| $t_{AVDV}$ | address to valid data in | – | 585 | – | $9t_{CLK} - 165$ | ns |
| $t_{LLWL}$ | ALE LOW to $\overline{RD}$ or $\overline{WR}$ LOW | 200 | 300 | $3t_{CLK} - 50$ | $3t_{CLK} + 50$ | ns |
| $t_{AVWL}$ | address valid to $\overline{RD}$ or $\overline{WR}$ LOW | 203 | – | 4 | – | ns |
| $t_{WHLH}$ | $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH | 43 | 123 | $t_{CLK} - 40$ | $t_{CLK} + 40$ | ns |
| $t_{QVWX}$ | data valid to $\overline{WR}$ transition | 23 | – | $t_{CLK} - 60$ | – | ns |
| $t_{QVWH}$ | data valid time $\overline{WR}$ HIGH | 433 | – | $7t_{CLK} - 150$ | – | ns |
| $t_{WHQX}$ | data hold after $\overline{WR}$ | 33 | – | $t_{CLK} - 50$ | – | ns |
| $t_{RLAZ}$ | $\overline{RD}$ LOW to address float | – | 12 | – | 12 | ns |

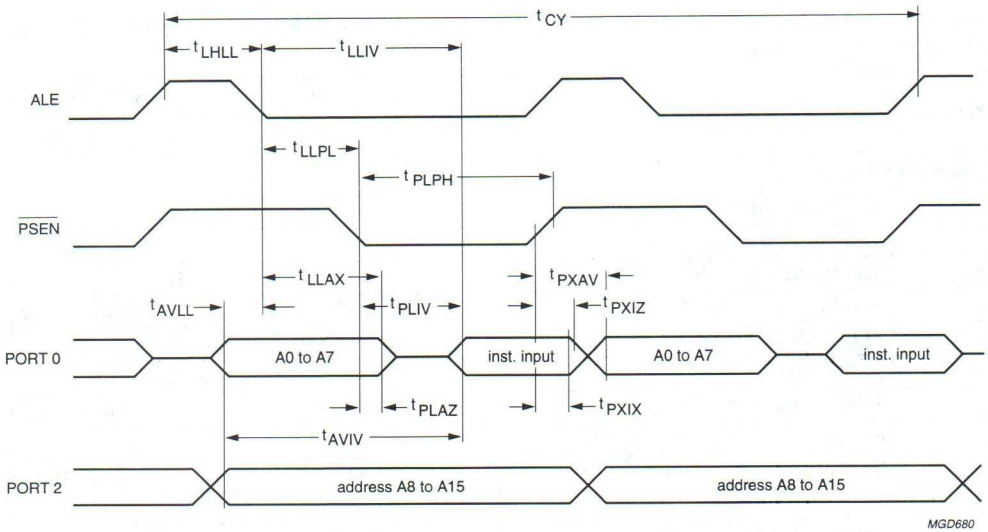



Fig.36 Read from Program Memory.

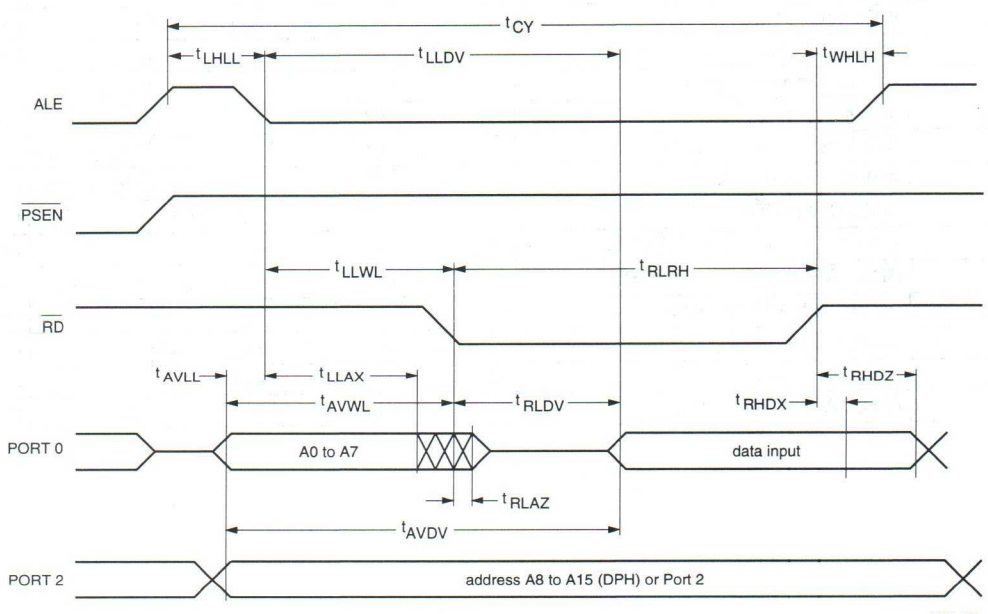



Fig.37 Read from Data Memory.

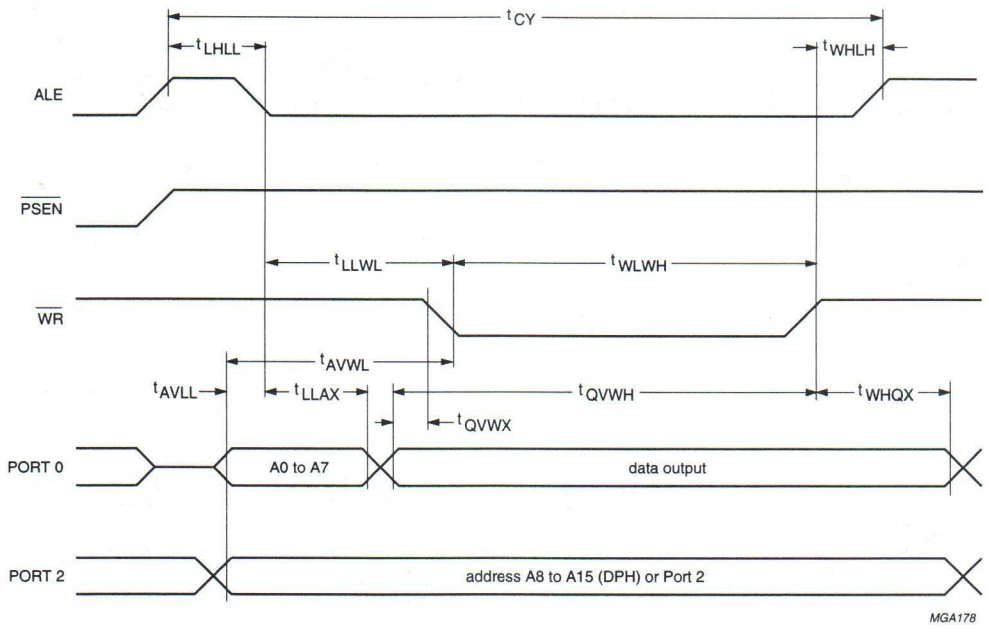



Fig.38 Write to Data Memory.

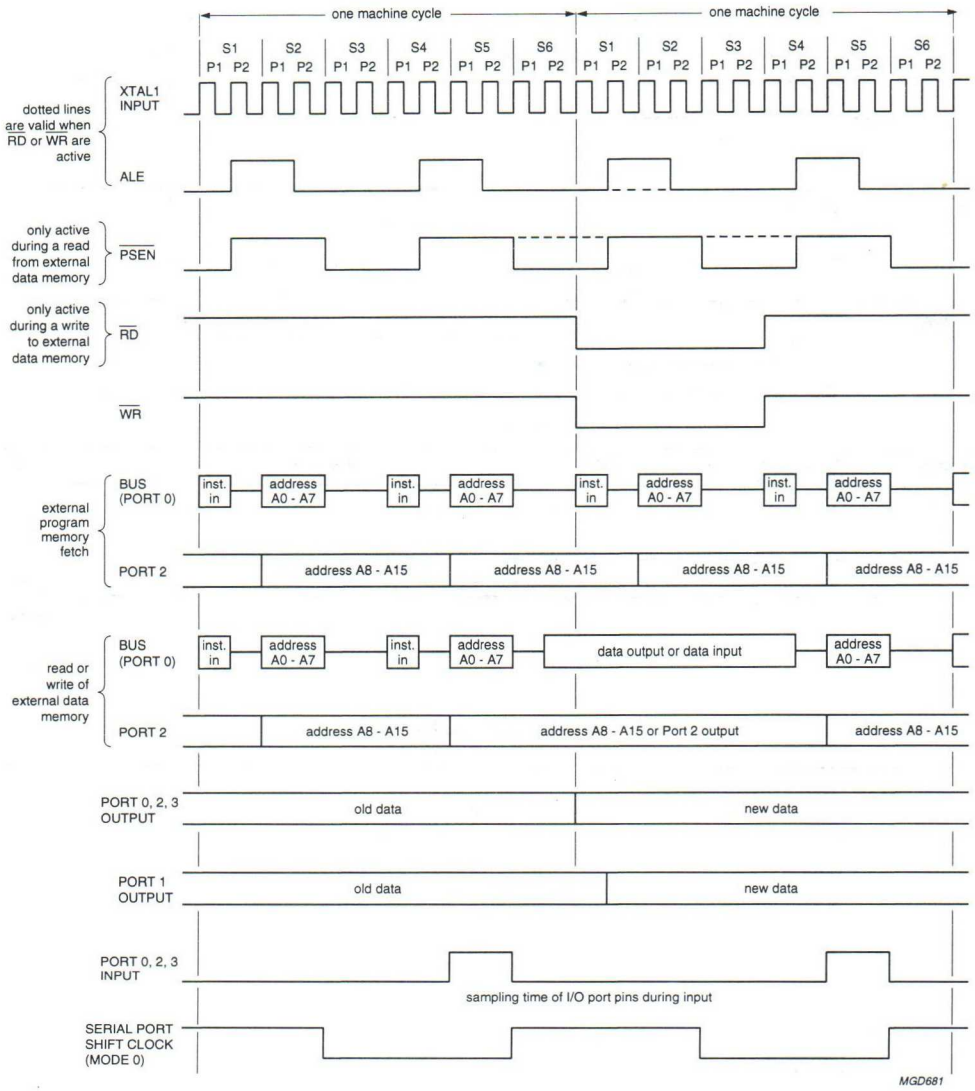



Fig.39 Instruction cycle timing.

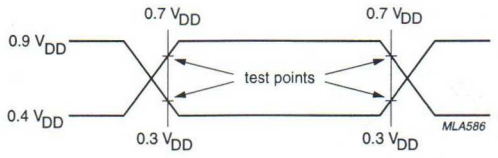



Fig.40 AC testing input waveform.

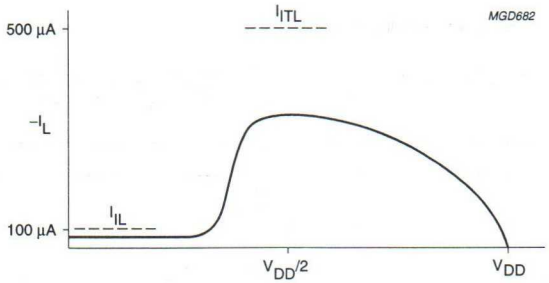
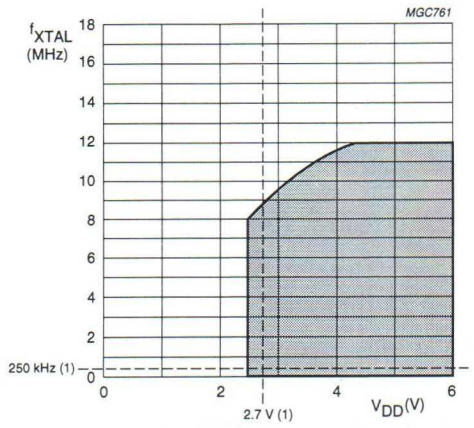



Fig.41 Input current.




(1) The area above the dotted lines give the ADC operating area.

Fig.42 Frequency operating range.

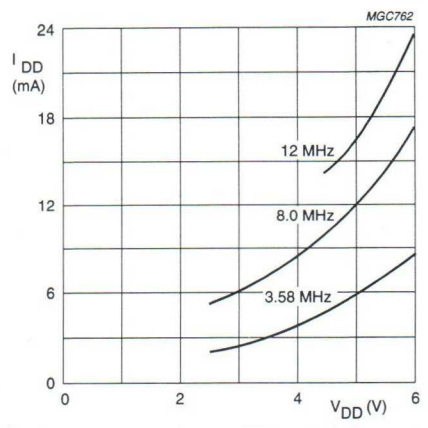



Fig.43 Typical operating current vs frequency and $V_{DD}$, $T_{amb}$ = 25 °C, oscillator option OSC3.

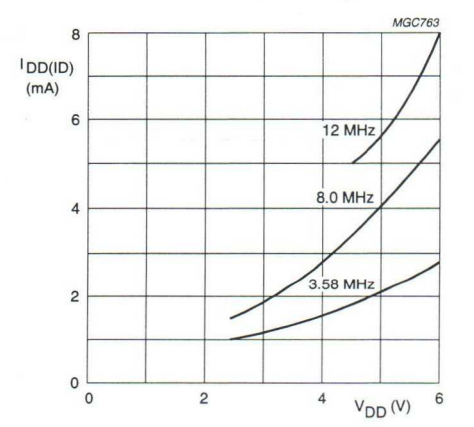



Fig.44 Typical Idle current vs frequency and $V_{DD}$, $T_{amb}$ = 25 °C, oscillator option OSC3.

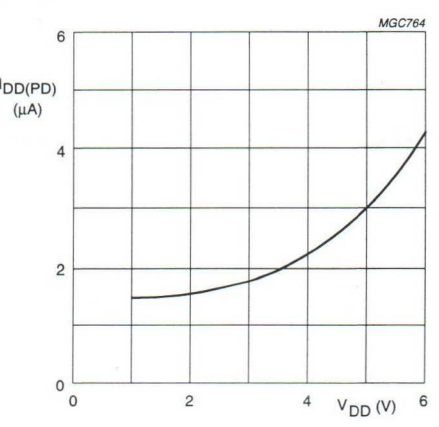



Fig.45 Typical Power-down current vs $V_{DD}$, $T_{amb}$ = 25 °C.

# 8-bit microcontroller with on-chip CAN P8xC592

## 1 FEATURES

- 80C51 central processing unit (CPU)
- 16 kbytes on-chip ROM, externally expandible to 64 kbytes
- 2 x 256 bytes on-chip RAM, externally expandible to 64 kbytes
- Two standard 16-bit timers/counters
- One additional 16-bit timer/counter coupled to four capture and three compare registers
- 10-bit ADC with 8 multiplexed analog inputs
- Two 8-bit resolution Pulse Width Modulated outputs
- 15 interrupt sources with 2 priority levels (2 to 6 external interrupt sources possible)
- Five 8-bit I/O ports, plus one 8-bit input port shared with analog inputs
- CAN-controller (CAN = Controller Area Network) with DMA data transfer facility to internal RAM
- 1 Mbit/s CAN-controller with bus failure management facility
- $\frac{1}{2}AV_{DD}$ reference voltage
- Full-duplex UART compatible with the standard 80C51
- On-chip Watchdog Timer (WDT)
- 1.2 to 16 MHz clock frequency.

## 2 GENERAL DESCRIPTION

The P8xC592 is a single-chip 8-bit high-performance microcontroller with on-chip CAN-controller, derived from the 80C51 microcontroller family. It uses the powerful 80C51 instruction set. Figure 1 shows a block diagram of the P8xC592.

The P8xC592 is manufactured in an advanced CMOS process, and is designed for use in automotive and general industrial applications. In addition to the 80C51 standard features, the device provides a number of dedicated hardware functions for these applications.

Two versions of the P8xC592 will be offered:

- P80C592 (without ROM)
- P83C592 (with ROM).

Hereafter these versions will be referred to as P8xC592.

The temperature range includes (max. $f_{CLK}$ = 16 MHz):

- –40 to +85 °C version, for general applications
- –40 to +125 °C version for automotive applications.

The P8xC592 combines the functions of the P8xC552 (microcontroller) and the PCA82C200 (Philips CAN-controller) with the following enhanced features:

- 16 kbytes Program Memory
- 2 x 256 bytes Data Memory
- DMA between CAN Transmit/Receive Buffer and internal RAM.

The main differences between P8xC592 and P8xC552 are:

- 16 kbytes programmable ROM (P8xC552 has 8 kbytes)
- Additional 256 bytes RAM
- A CAN-controller instead of the $I^2C$-serial interface.

## 3 ORDERING INFORMATION

| TYPE NUMBER | PACKAGE | | | TEMPERATURE RANGE (°C) | FREQ. (MHz) |
|---|---|---|---|---|---|
| | NAME | DESCRIPTION | VERSION | | |
| **Without ROM** | | | | | |
| P80C592FFA | PLCC68 | plastic leaded chip carrier; 68 leads | SOT188-2 | –40 to +85 | 1.2 to 16 |
| P80C592FHA | | | | –40 to +125 | |
| **With ROM** | | | | | |
| P83C592FFA | PLCC68 | plastic leaded chip carrier; 68 leads | SOT188-2 | –40 to +85 | 1.2 to 16 |
| P83C592FHA | | | | –40 to +125 | |

## 4 BLOCK DIAGRAM

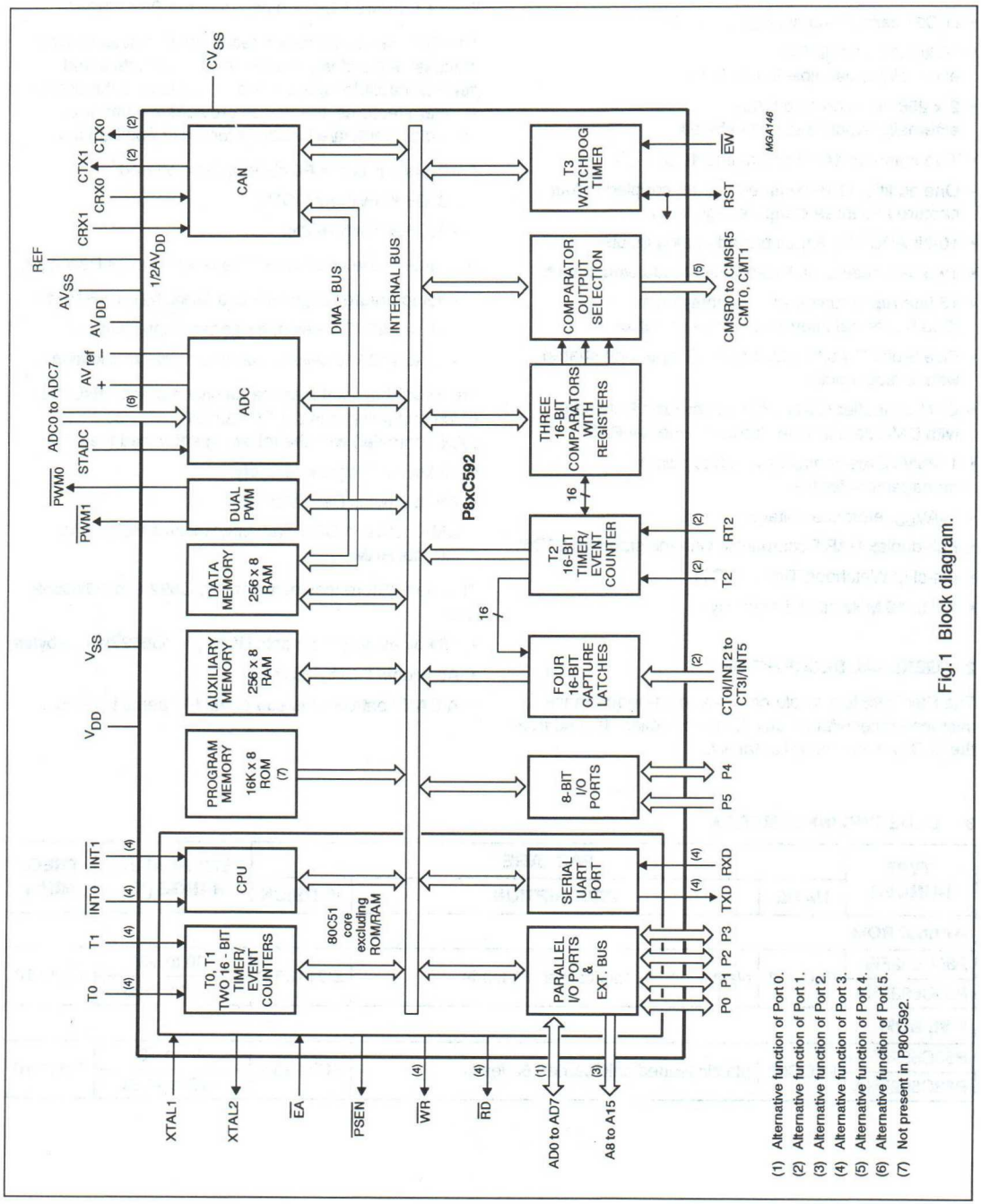



Fig.1 Block diagram.

(1) Alternative function of Port 0.
(2) Alternative function of Port 1.
(3) Alternative function of Port 2.
(4) Alternative function of Port 3.
(5) Alternative function of Port 4.
(6) Alternative function of Port 5.
(7) Not present in P80C592.

## 5 PINNING

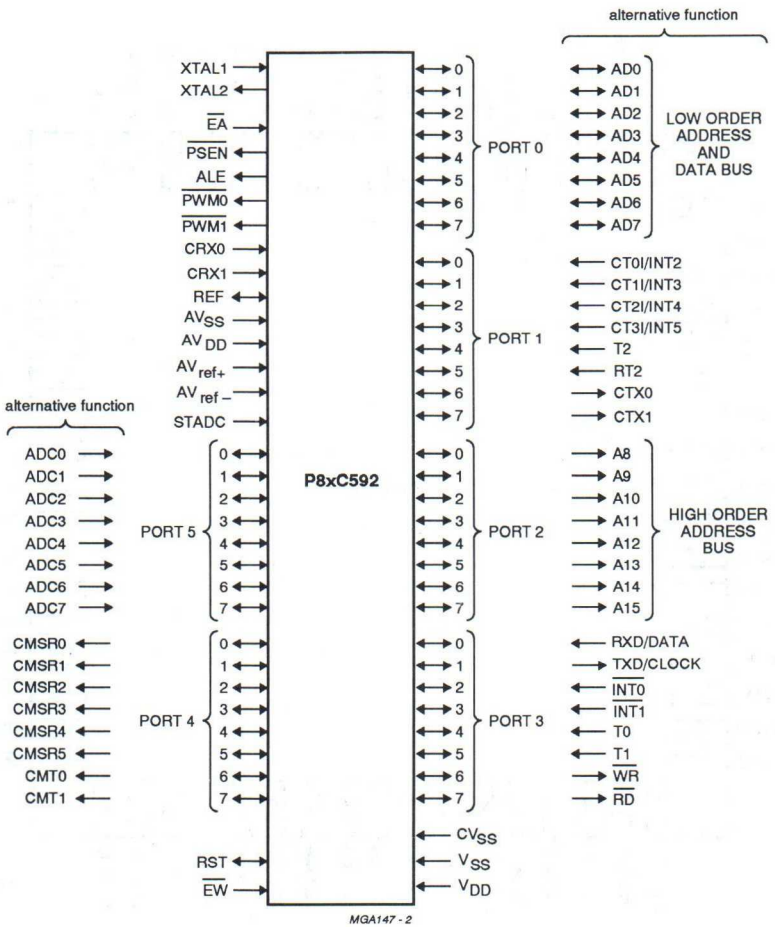



Fig.2 Pin functions.

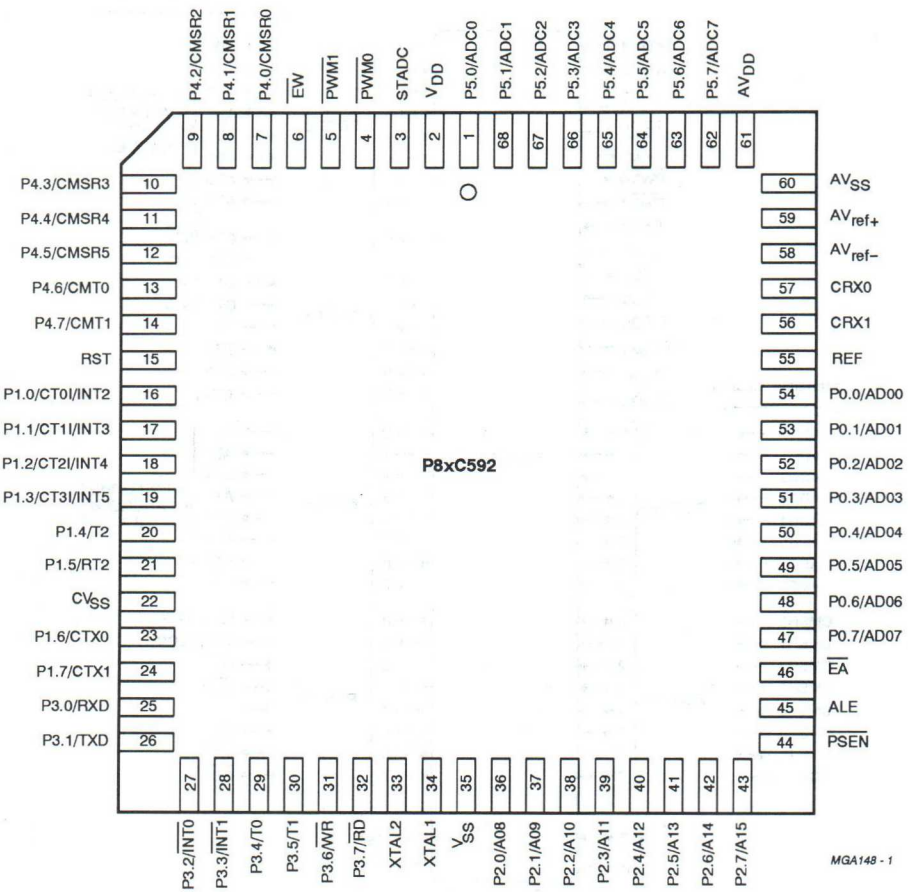



Fig.3 Pin configuration PLCC68/SOT188-2 version (P8xC592FFA; FHA;).

**Table 1** Pin description for **single function** pins (SOT188-2; see note 1)

| SYMBOL | PIN | DESCRIPTION |
|---|---|---|
| $V_{DD}$ | 2 | **Power supply,** digital part (+5 V). For normal operation and power reduced modes. |
| STADC | 3 | **Start ADC operation.** Input starting analog-to-digital conversion (note 2). This pin must not float. |
| $\overline{PWM0}$ | 4 | **Pulse width modulation output 0.** |
| $\overline{PMW1}$ | 5 | **Pulse width modulation output 1.** |
| $\overline{EW}$ | 6 | **Enable Watchdog Timer (WDT):** enable for T3 Watchdog Timer and disable Power-down mode. This pin must not float. |
| RST | 15 | **Reset:** input to reset the P8xC592 (note 3). |
| $CV_{SS}$ | 22 | **CAN ground potential** for the CAN transmitter outputs. |
| XTAL2 | 33 | **Crystal pin 2:** output of the inverting amplifier that forms the oscillator. When an external clock oscillator is used this pin is left open-circuit. |
| XTAL1 | 34 | **Crystal pin 1:** input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external clock oscillator signal, when an external oscillator is used. |
| $V_{SS}$ | 35 | **Ground**, digital part. |
| $\overline{PSEN}$ | 44 | **Program Store Enable:** Read strobe to external Program Memory (active LOW). Drive: 8 × LSTTL inputs. |
| ALE | 45 | **Address Latch Enable:** latches the Low-byte of the address during accesses to external memory (note 4). Drive: 8 × LSTTL inputs; handles CMOS inputs without an external pull-up. |
| $\overline{EA}$ | 46 | **External Access input.** See note 5. |
| REF | 55 | **$\frac{1}{2}AV_{DD}$ reference voltage** output respectively input (note 6). |
| CRX1 | 56 | **Inputs from the CAN-bus line** to the differential input comparator of the on-chip CAN-controller (note 7). |
| CRX0 | 57 | |
| $AV_{REF-}$ | 58 | **Low-end of ADC** (analog-to-digital) conversion reference resistor. |
| $AV_{REF+}$ | 59 | **High-end of ADC** (analog-to-digital) conversion reference resistor (note 8). |
| $AV_{SS}$ | 60 | **Ground**, analog part. For ADC, CAN receiver and reference voltage. |
| $AV_{DD}$ | 61 | **Power supply**, analog part (+5 V). For ADC, CAN receiver and reference voltage. |

**Notes**

1. To avoid a 'latch up' effect at power-on: $V_{SS} - 0.5\ V <$ 'voltage on any pin at any time' $< V_{DD} + 0.5\ V$.
2. Triggered by a rising edge. ADC operation can also be started by software.
3. RST also provides a reset pulse as output when timer T3 overflows or after a CAN wake-up from Power-down.
4. ALE is activated every six oscillator periods. During an external data memory access one ALE pulse is skipped.
5. See Section 7.1, Table 3 for $\overline{EA}$ operation. For P83Cxxx microcontrollers specified with the option 'ROM-code protection', the $\overline{EA}$ pin is latched during reset and is 'don't care' after reset, regardless of whether the ROM-code protection is selected or not.

6. Pin 55, REF:
   a) Selection of input resp. output dependent of CAN Control Register bit 5 (CR.5; see Section 13.5.3 Table 32).
   b) If the internal reference is used, then REF should be connected to $AV_{SS}$ via a capacitor with a value of ≥10 nF.
   c) After an external reset (RST = HIGH) the internal $\frac{1}{2}AV_{DD}$ source is activated and, REF is a reference output.
   d) If the CAN-controller is in the reset state, e.g. after an external reset, then the $\frac{1}{2}AV_{DD}$ source is switched off during Power-down mode.
7. CAN-bus line:
   a) CRX0 level > CRX1 level is interpreted as a logic 1 (recessive).
   b) CRX0 level < CRX1 level is interpreted as a logic 0 (dominant).
8. The level of $AV_{REF+}$ must be higher than that of $AV_{REF-}$.

**Table 2** Pin description for pins with **alternative functions** (SOT188-2 and NO330; see note 1)

| SYMBOL | | PIN | DESCRIPTION |
|---|---|---|---|
| **DEFAULT** | **ALTERNATIVE** | | |
| **Port 4** | | | |
| P4.0 to P4.7 | | 7 to 14 | **8-bit quasi-bidirectional I/O port.** |
| | CMSR0 | 7 | **Compare and Set/Reset outputs** for Timer T2. |
| | CMSR1 | 8 | |
| | CMSR2 | 9 | |
| | CMSR3 | 10 | |
| | CMSR4 | 11 | |
| | CMSR5 | 12 | |
| | CMT0 | 13 | **Compare and toggle outputs** for Timer T2. |
| | CMT1 | 14 | |
| **Port 1** | | | |
| P1.0 to P1.7 | | 16 to 21, 23, 24 | **8-bit quasi-bidirectional I/O port.** |
| | CT0I/INT2 | 16 | **Capture timer inputs** for Timer T2, or **External interrupt inputs.** |
| | CT1I/INT3 | 17 | |
| | CT2I/INT4 | 18 | |
| | CT3I/INT5 | 19 | |
| | T2 | 20 | **T2 event input** (rising edge triggered). |
| | RT2 | 21 | **T2 timer reset input** (rising edge triggered). |
| | CTX0 | 23 | **CAN transmitter output 0** (note 2). |
| | CTX1 | 24 | **CAN transmitter output 1** (note 2). |

| SYMBOL | | PIN | DESCRIPTION |
|---|---|---|---|
| **DEFAULT** | **ALTERNATIVE** | | |
| **Port 3** | | | |
| P3.0 to P3.7 | | 25 to 32 | **8-bit quasi-bidirectional I/O port.** |
| | RXD | 25 | **Serial Input Port.** |
| | TXD | 26 | **Serial Output Port.** |
| | $\overline{INT0}$ | 27 | **External interrupt inputs.** |
| | $\overline{INT1}$ | 28 | |
| | T0 | 29 | **Timer 0 external input.** |
| | T1 | 30 | **Timer 1 external input.** |
| | $\overline{WR}$ | 31 | **External Data Memory Write strobe.** |
| | $\overline{RD}$ | 32 | **External Data Memory Read strobe.** |
| **Port 2 (Sink/source: 1 × TTL = 4 × LSTTL inputs)** | | | |
| P2.0 to P2.7 | | 36 to 43 | **8-bit quasi-bidirectional I/O port.** |
| | A08 to A15 | | **High-order address byte for external memory.** |
| **Port 0 (Sink/source: 8 × LSTTL inputs)** | | | |
| P0.7 to P0.0 | | 47 to 54 | **8-bit open drain bidirectional I/O port.** |
| | AD7 to AD0 | | **Multiplexed Low-order address and Data bus for external memory.** |
| **Port 5** | | | |
| P5.7 to P5.0 | | 62 to 68, 1 | **8-bit input port.** |
| | ADC7 to ADC0 | | **8 input channels to ADC.** |

**Notes**

1. To avoid a 'latch up' effect at power-on: $V_{SS} - 0.5\ V <$ 'voltage on any pin at any time' $< V_{DD} + 0.5\ V$.
2. If the CAN-controller is in the reset state (e.g. after a power-up reset; CAN Control Register bit CR.0; see Section 13.5.3 Table 32), the CAN transmitter outputs are floating and the pins P1.6 and P1.7 can be used as open-drain port pins. After a power-up reset the port data is HIGH, leaving the pins P1.6 and P1.7 floating.

## 6 FUNCTIONAL DESCRIPTION

The P8xC592 functions will be described as shown in the following overview:

- Memory organization
- I/O Port structure
- Pulse Width Modulated outputs
- Analog-to-digital Converter
- Timers/Counters
- Serial I/O Ports
- Interrupt system
- Power reduction modes
- Oscillator circuitry
- Reset circuitry
- Instruction Set.

## 7 MEMORY ORGANIZATION

The Central Processing Unit (CPU) manipulates operands in three memory spaces (see Fig.4) as follows:

- 16 kbytes internal resp. 64 kbytes external Program Memory
- 512 bytes internal Data Memory MAIN- and AUXILIARY RAM
- up to 64 kbytes external Data Memory (with 256 bytes residing in the internal AUXILIARY RAM).

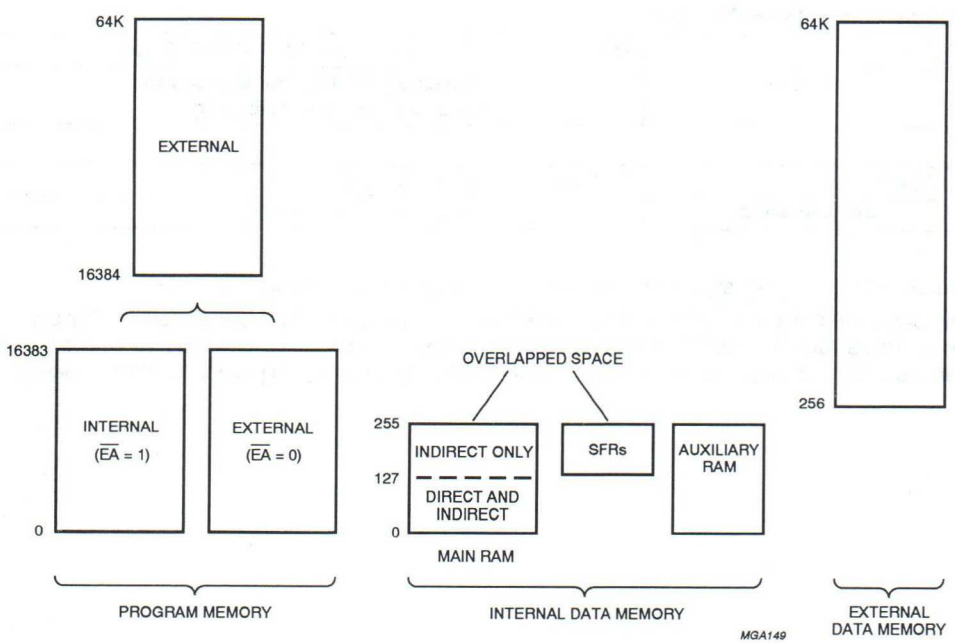



Fig.4 Memory map.

### 7.1 Program Memory

The Program Memory of the P8xC592 consists of 16 kbytes ROM on-chip, externally expandible up to 64 kbytes.

**Table 3** Instruction fetch controlled by $\overline{EA}$

| PIN $\overline{EA}$ (note 1) | | INSTRUCTIONS FETCHED FROM: | ADDRESS LOCATION |
|---|---|---|---|
| DURING RESET LATCHED TO: | AFTER RESET | | |
| H | – | internal Program Memory (note 2) | 0000H → 3FFFH |
| H | – | external Program Memory | 4000H → FFFFH |
| L | – | | 0000H → FFFFH |
| – | 'don't care' | – | – |

**Notes**

1. This implementation prevents reading of the internal program code by switching from external Program Memory during a MOVC instruction.
2. By setting a security bit the internal Program Memory content is protected, which means it cannot be read out. If the security bit has been set to LOW there are no restrictions for the MOVC instruction.

### 7.2 Internal Data Memory

The internal Data Memory is physically built-up and accessible as shown in Table 4 (see Fig.5).

**Table 4** Internal Data Memory size and address mode

| INTERNAL DATA MEMORY | SIZE | LOCATION | ADDRESS MODE | | POINTERS |
|---|---|---|---|---|---|
| | | | DIRECT | INDIRECT | |
| MAIN RAM (note 1) | 256 bytes | 0 to 127 | X | X | address pointers are R0 and R1 of the selected register bank |
| | | 128 to 255 | – | X | |
| AUXILIARY RAM (note 2) | 256 bytes | 0 to 255 | – | X | address pointers are R0 and R1 of the selected register bank and the DPTR |
| SFRs (note 3) | 128 bytes | 128 to 255 | X | – | – |

**Notes**

1. MAIN RAM can be addressed directly and indirectly as in the 80C51.
2. AUXILIARY RAM (0 to 255):
   a) Is indirectly addressable in the same way as the external Data Memory with MOVX instructions.
   b) Access will not affect the ports P0, P2, P3.6 and P3.7 during internal program execution.
3. SFRs = Special Function Registers.

### 7.2.1 MAIN RAM

Four 8-bit register banks occupy the lower RAM area,

- BANK 0: location 0 to 7
- BANK 1: location 8 to 15
- BANK 2: location 16 to 23
- BANK 4: location 24 to 31.

Only one of these banks may be enabled at the same time.

The next 16 bytes, locations 32 through 45, contains 128 directly addressable bit locations.

The stack can be located anywhere in the internal MAIN RAM address space. The stack depth is only limited by the internal RAM space available. All registers except the program counter and the four 8-bit register banks reside in the SFR address space.

## 7.3 External Data Memory

An access to external Data Memory locations higher than 255 will be performed with the MOVX @DPTR instructions in the same way as in the 80C51 structure, i.e. with P0 and P2 as data/address bus and P3.6 and P3.7 as Write and Read strobe signals.

Note that these external Data Memory locations cannot be accessed with R0 or R1 as address pointer.

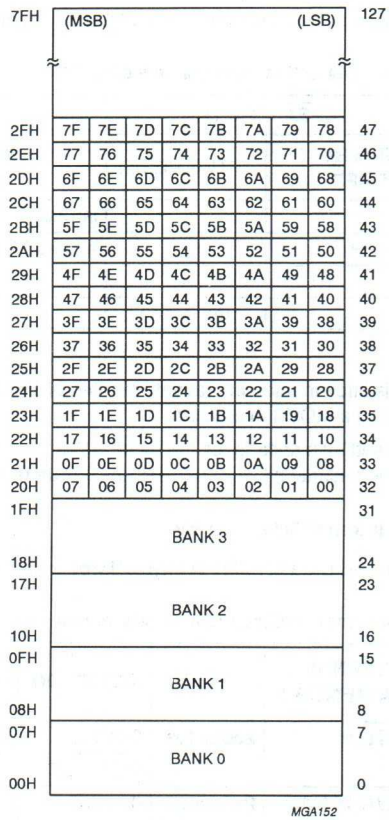



Fig.5 Internal MAIN RAM bit addresses.

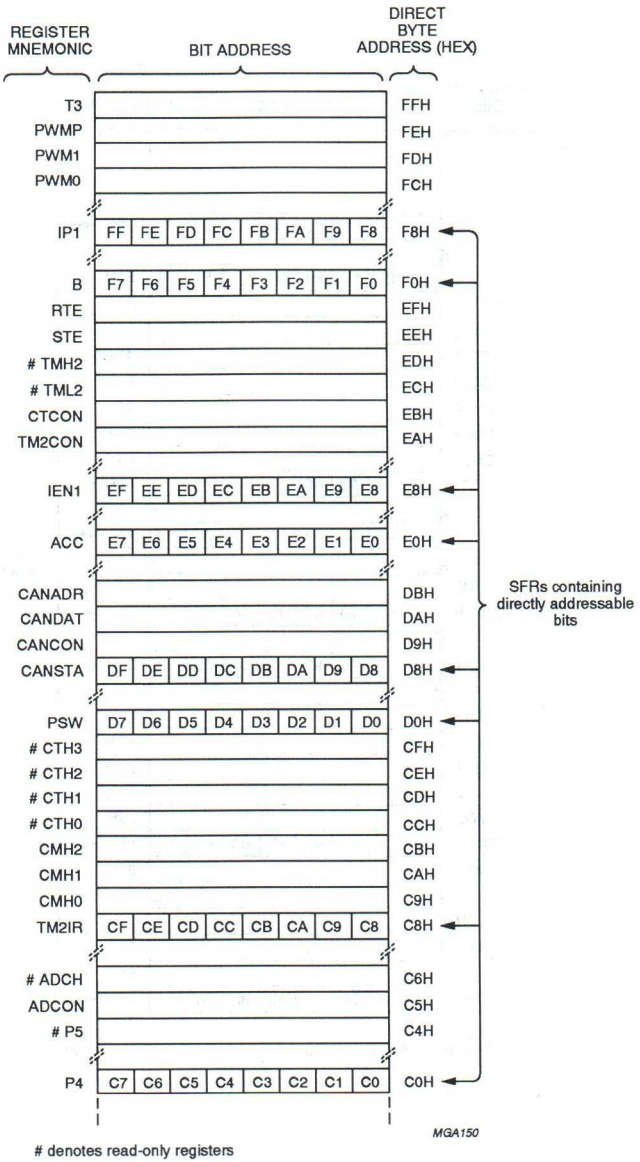



Fig.6 Special Function Register memory map (a).

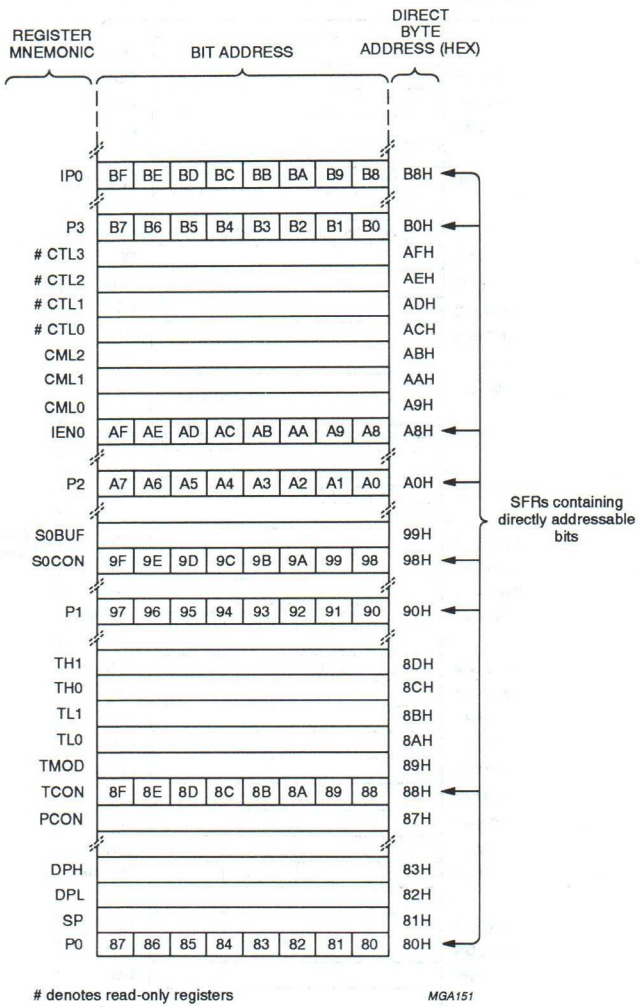



Fig.7 Special Function Register memory map (b).

## 8 I/O PORT STRUCTURE

The P8xC592 has six 8-bit parallel ports: Port 0 to Port 5. In addition to the standard 8-bit parallel ports, the I/O facilities also include a number of special I/O lines. The use of a Port 1, Port 3 or Port 4 pins as an alternative function is carried out automatically provided the associated SFR bit is set HIGH.

**Table 5 Default** Port functions

| PORT | TYPE | FUNCTION | REMARKS |
|---|---|---|---|
| Port 0 | I/O | The same as in the 80C51 | Except for the additional functions of P1.6 and P1.7. |
| Port 1 | I/O | | |
| Port 2 | I/O | | |
| Port 3 | I/O | | |
| Port 4 | I/O | Parallel I/O port | Parallel I/O function is identical to Port1, 2 and 3. |
| Port 5 | I | Parallel input port with an input function only | May be used as normal inputs if the ADC function is inoperative. |

**Table 6 Alternative** Port functions

| PORT | TYPE | FUNCTION | REMARKS |
|---|---|---|---|
| Port 0 | I/O | Multiplexed Low-order address and Data bus for external memory (AD7 to AD0) | Provides the multiplexed Low-order address and data bus used for expanding the P8xC592 with standard memories and peripherals. |
| Port 1 | I/O | Capture timer inputs for Timer T2 (CT0I to CT3I), or External interrupt request inputs (INT2 to INT5) | External interrupt request inputs, if capture information is not utilized. |
| | | T2 event input (T2) | External counter input. |
| | | T2 timer reset input (RT2) | External counter reset input. |
| | | CAN transmitter output 0 (CTX0) | CTX0 and CTX1 outputs of the CAN interface (note 1). |
| | | CAN transmitter output 1 (CTX1) | |
| Port 2 | I/O | High-order address byte for external memory (A08 to A15) | Port 2 provides the High-order address bus when the P8xC592 is expanded with external Program Memory and/or external Data Memory. |
| Port 3 | I/O | Serial Input Port (RXD) | Receiver input of serial port SIO0 (UART). |
| | | Serial Output Port (TXD) | Transmitter output of serial port SIO0 (UART). |
| | | External interrupt ($\overline{\text{INT0}}$) | External interrupt request inputs. |
| | | External interrupt ($\overline{\text{INT1}}$) | |
| | | Timer 0 external input (T0) | Counter inputs. |
| | | Timer 1 external input (T1) | |
| | | External data memory Write strobe ($\overline{\text{WR}}$) | Control signal to write to external Data Memory. |
| | | External data memory Read strobe ($\overline{\text{RD}}$) | Control signal to read from external Data Memory. |
| Port 4 | I/O | Compare and Set/Reset outputs (CMSR0 to CMSR5) | Can be configured to provide signals indicating a match between Timer counter T2 and its compare registers. |
| | | Compare and toggle outputs (CMT0, CMT1) | |
| Port 5 | I | Input channels to ADC (ADC7 to ADC0) | Port 5 may be used in conjunction with the ADC interface (note 2). |

**Notes to the alternative Port functions**

1. Port lines P1.6 and P1.7 may be selected as CTX0 and CTX1 outputs of the serial port SIO1 (CAN). After reset P1.6 and P1.7 may be used as normal I/O ports, if the CAN interface is not used.
2. Unused analog inputs can be used as digital inputs. As Port 5 lines may be used as inputs to the ADC, these digital inputs have an inherent hysteresis to prevent the input logic from drawing too much current from the power lines when driven by analog signals.
   Channel-to-channel crosstalk should be taken into consideration when both digital and analog signals are simultaneously input to Port 5 (see Chapter 20).

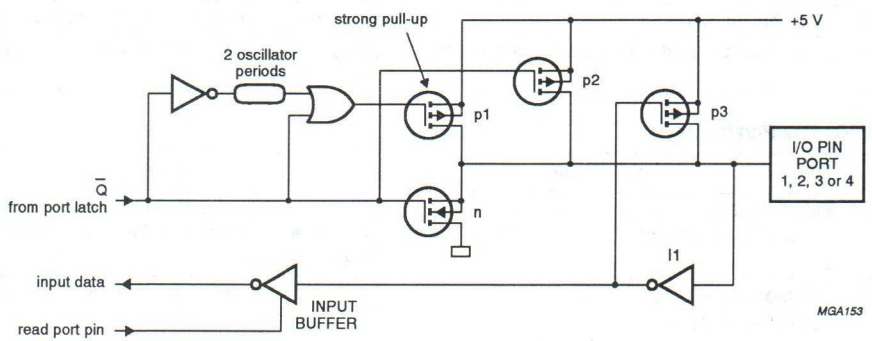



Fig.8 I/O buffers in the P8xC592 (P1.0 to P1.5, Ports 2, 3, and 4).

## 9 PULSE WIDTH MODULATED OUTPUTS (PWM)

Two Pulse Width Modulated (PWM) output channels are available with the P8xC592. These channels provide output pulses of programmable length and interval. The repetition frequency is defined by an 8-bit prescaler PWMP which generates the clock for the counter. Both the prescaler and counter are common to both PWM channels. The 8-bit counter counts modulo 255 i.e. from 0 to 254 inclusive. The value of the 8-bit counter is compared to the contents of two registers: PWM0 and PWM1.

Provided the contents of either of these registers is greater than the counter value, the output of $\overline{\text{PWM0}}$ or $\overline{\text{PWM1}}$ is set LOW. If the contents of these register are equal to, or less than the counter value, the output will be HIGH. The pulse-width-ratio is therefore defined by the contents of the register PWM0 and PWM1. The pulse-width-ratio is in the range of 0 to $^{255}/_{255}$ and may be programmed in increments of $^{1}/_{255}$.

The repetition frequency $f_{PWM}$, at the $\overline{\text{PWMn}}$ outputs is given by:

$$f_{PWM} = \frac{f_{CLK}}{2 \times (PWMP + 1) \times 255}$$

When using an oscillator frequency of 16 MHz, for example, the above formula would give a repetition frequency range of 123 Hz to 31.4 kHz.

By loading the PWM registers with either 00H or FFH, the PWM outputs can be retained at a constant HIGH or LOW level respectively. When loading FFH to the PWM registers, the 8-bit counter will never actually reach this (FFH) value.
Both output pins $\overline{\text{PWMn}}$ are driven by push-pull drivers, and are not shared with any other function.

### 9.1 Prescaler frequency control register (PWMP)

**Table 7** Prescaler frequency control register (address FEH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| PWMP.7 | PWMP.6 | PWMP.5 | PWMP.4 | PWMP.3 | PWMP.2 | PWMP.1 | PWMP.0 |

**Table 8** Description of PWMP bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7<br>to<br>0 | PWMP.7<br>to<br>PWMP.0 | **Prescaler division factor.**<br>The Prescaler division factor = (PWMP) + 1. |

### 9.2 Pulse Width Register 0 (PWM0)

**Table 9** Pulse Width Register (address FCH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| PWM0.7 | PWM0.6 | PWM0.5 | PWM0.4 | PWM0.3 | PWM0.2 | PWM0.1 | PWM0.0 |

**Table 10** Description of PWM0 bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7<br>to<br>0 | PWM0.7<br>to<br>PWM0.0 | **Pulse width ratio.**<br>LOW/HIGH ratio of $\overline{PWMn}$ signals $= \frac{(PWMn)}{255 - (PWMn)}$ |

### 9.3 Pulse Width Register 1 (PWM1)

**Table 11** Pulse width register (address FDH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| PWM1.7 | PWM1.6 | PWM1.5 | PWM1.4 | PWM1.3 | PWM1.2 | PWM1.1 | PWM1.0 |

**Table 12** Description of PWM1 bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7<br>to<br>0 | PWM1.7<br>to<br>PWM1.0 | **Pulse width ratio.**<br>LOW/HIGH ratio of $\overline{PWMn}$ signals $= \frac{(PWMn)}{255 - (PWMn)}$ |

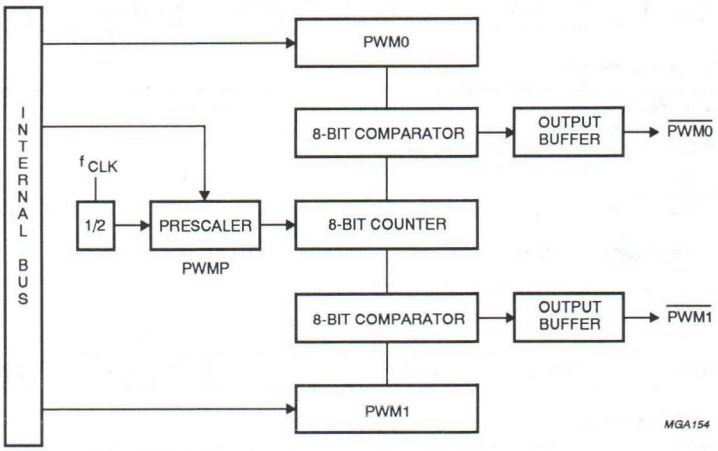



Fig.9 Functional diagram of Pulse Width Modulated outputs.

## 10 ANALOG-TO-DIGITAL CONVERTER (ADC)

The analog input circuitry consists of an 8-input analog multiplexer and an ADC with 10-bit resolution. The analog reference voltage and analog power supplies are connected via separate input pins. The conversion takes 50 machine cycles i.e. 37.5 μs at 16 MHz oscillator frequency. The input voltage swing is from 0 V to $AV_{DD}$. The ADC is controlled using the ADCON control register. Register bits ADCON.0 to ADCON.2 select the input channels of the analog multiplexer (see Fig.10). The completion of the 10-bit analog-to-digital conversion is flagged by ADCI in the ADCON register and the result is stored in the SFR ADCH (upper 8-bits) and the 2 lower bits (ADC.1 and ADC.0) in register ADCON.

An analog-to-digital conversion in progress is unaffected by an external or software ADC start. The result of a completed conversion remains unchanged provided ADCI = HIGH. While ADCI or ADCS are HIGH, a new ADC START will be blocked and consequently lost. An analog-to-digital conversion already in progress is aborted when the Idle or Power-down mode is entered.

The result of a completed conversion (ADCI = HIGH) remains unaffected during the Idle mode.
The LOW-to-HIGH transition of STADC is recognized at the end of a machine cycle and the conversion commences at the beginning of the next cycle. When a conversion is initiated by software, the conversion starts at the beginning of the machine cycle following the instruction that sets ADCS.

The next two machine cycles are used to initiate the converter. At the end of this first cycle, the ADCS status flag is set to HIGH while the conversion is in progress. Sampling of the analog input commences at the end of the second cycle.

During the next eight machine cycles, the voltage at the previously selected pin of Port 5 is sampled and this input voltage should be stable in order to obtain a useful sample. In any case, the input voltage slew rate must be less than 10 V/ms (5 V conversion range) in order to prevent an undefined result. The conversion takes four machine cycles per bit.

### 10.1 ADC Control register (ADCON)

**Table 13** ADC Control register (address C5H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| ADC.1 | ADC.0 | ADEX | ADCI | ADCS | AADR2 | AADR1 | AADR0 |

**Table 14** Description of the ADCON bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | ADC.1 | Bit 1 of ADC converted value. |
| 6 | ADC.0 | Bit 0 of ADC converted value. |
| 5 | ADEX | Enable external start of conversion by STADC. If ADEX is:<br>LOW, then conversion cannot be started externally by STADC (only by software by setting ADCS)<br>HIGH, then conversion can be started externally by a rising edge on STADC or externally. |
| 4 | ADCI | **ADC interrupt flag.** This flag is set when an analog-to-digital conversion result is ready to be read. If enabled, an interrupt is invoked. The flag must be cleared by software.<br>It cannot be set by software (see Table 15). |
| 3 | ADCS | **ADC start and status.** Setting this bit starts an analog-to-digital conversion. It may be set by software or by the external signal STADC. The ADC logic ensures that this signal is HIGH while the ADC is busy. On completion of the conversion, ADCS is reset at the same time the interrupt flag ADCI is set. ADCS can not be reset by software (see Table 15). |
| 2 | AADR2 | **Analog input select**. This binary coded address selects one of the eight analog port pins of P5 to be input to the converter. It can only be changed when ADCI and ADCS are both LOW. AADR2 is the MSB. (e.g. 100B selects the analog input channel ADC4) |
| 1 | AADR1 | |
| 0 | AADR0 | |

**Table 15** ADCI and ADCS operating modes

If ADCI is cleared by software while ADCS is set at the same time a new analog-to-digital conversion with the same channel-number may be started. It is recommended to reset ADCI before ADCS is set.

| ADCI | ADCS | OPERATION |
|---|---|---|
| 0 | 0 | ADC not busy, a conversion can be started. |
| 0 | 1 | ADC busy, start of a new conversion is blocked. |
| 1 | X (don't care) | Conversion completed; see note 1. |

**Note**

1. Start of a new conversion requires ADCI = 0.

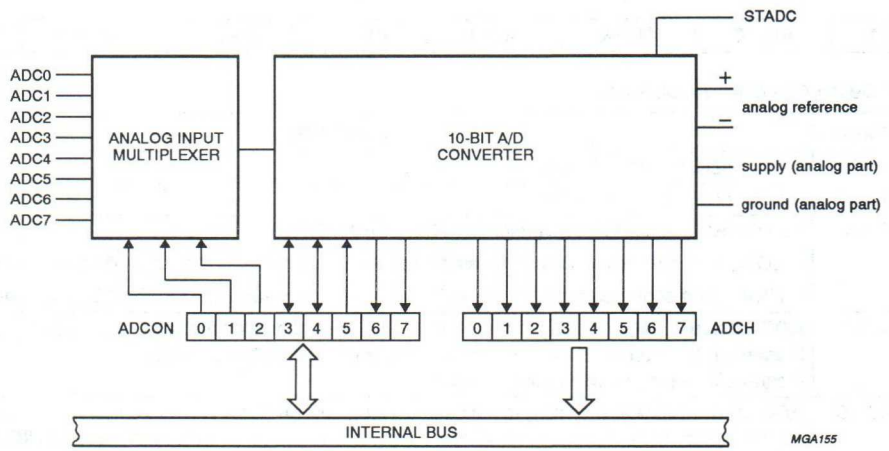



Fig.10 Functional diagram of analog input.

## 11 TIMERS/COUNTERS

The P8xC592 contains:

- Three 16-bit timer/event counters: Timer 0, Timer 1 and Timer T2
- One 8-bit timer, T3 (Watchdog WDT).

### 11.1 Timer 0 and Timer 1

Timer 0 and Timer 1 may be programmed to carry out the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests.

Timer 0 and Timer 1 can be programmed independently to operate in 3 modes:

Mode 0 8-bit timer or 8-bit counter each with divide-by-32 prescaler.

Mode 1 16-bit timer-interval or event counter.

Mode 2 8-bit timer-interval or event counter with automatic reload upon overflow.

Timer 0 can be programmed to operate in an additional mode as follows:

Mode 3 one 8-bit time-interval or event counter and one 8-bit timer-interval counter.

When Timer 0 is in Mode 3, Timer 1 can be programmed to operate in Modes 0, 1 or 2 but cannot set an interrupt flag or generate an interrupt. However, the overflow from Timer 1 can be used to pulse the Serial Port baud-rate generator.

The frequency handling range of these counters with a 16 MHz crystal is as follows:

- In the timer function, the timer is incremented at a frequency of 1.33 MHz ($1/12$ of the oscillator frequency)
- 0 Hz to an upper limit of 0.66 MHz ($1/24$ of the oscillator frequency) when programmed for external inputs.

Both internal and external inputs can be gated to the counter by a second external source for directly measuring pulse durations. When configured as a counter, the register is incremented on every falling edge on the corresponding input pin, T0 or T1.

The earliest moment, when the incremented register value can be read is during the second machine cycle following the machine cycle within which the incrementing pulse occurred.The counters are started and stopped under software control. Each one sets its interrupt request flag when it overflows from all HIGHs to all LOWs (or automatic reload value), with the exception of Mode 3 as previously described.

### 11.2 Timer T2 Capture and Compare Logic

Timer T2 is a 16-bit timer/counter which has capture and compare facilities (see Fig.11).

The 16-bit timer/counter is clocked via a prescaler with a programmable division factor of 1, 2, 4 or 8. The input of the prescaler is clocked with $1/12$ of the oscillator frequency, or by an external source connected to the T2 input, or it is switched off. The maximum repetition rate of the external clock source is $1/12 f_{CLK}$, twice that of Timer 0 and Timer 1. The prescaler is incremented on a rising edge. It is cleared if its division factor or its input source is changed, or if the timer/counter is reset.

T2 is readable 'on the fly', without any extra read latches; this means that software precautions have to be taken against misinterpretation at overflow from least to most significant byte while T2 is being read. T2 is not loadable and is reset by the RST signal or at the positive edge of the input signal RT2, if enabled. In the Idle mode the timer/counter and prescaler are reset and halted.

T2 is connected to four 16-bit Capture Registers: CT0, CT1, CT2 and CT3. A rising or falling edge on the inputs CT0I, CT1I, CT2I or CT3I (alternative function of Port 1) results in loading the contents of T2 into the respective Capture Registers and an interrupt request.

Using the Capture Register CTCON, these inputs may invoke capture and interrupt request on a positive edge, a negative edge or on both edges. If neither a positive nor a negative edge is selected for capture input, no capture or interrupt request can be generated by this input.

The contents of the Compare Registers CM0, CM1 and CM2 are continually compared with the counter value of Timer T2. When a match occurs, an interrupt may be invoked. A match of CM0 sets the bits 0 to 5 of Port 4, a CM1 match resets these bits and a CM2 match toggles bits 6 and 7 of Port 4, provided these functions are enabled by the STE/RTE registers. A match of CM0 and CM1 at the same time results in resetting bits 0 to 5 of Port 4. CM0, CM1 and CM2 are reset by the RST signal.

Port 4 can be read and written by software without affecting the toggle, set and reset signals. At a byte overflow of the least significant byte, or at a 16-bit overflow of the timer/counter, an interrupt sharing the same interrupt vector is requested. Either one or both of these overflows can be programmed to request an interrupt. All interrupt flags must be reset by software.

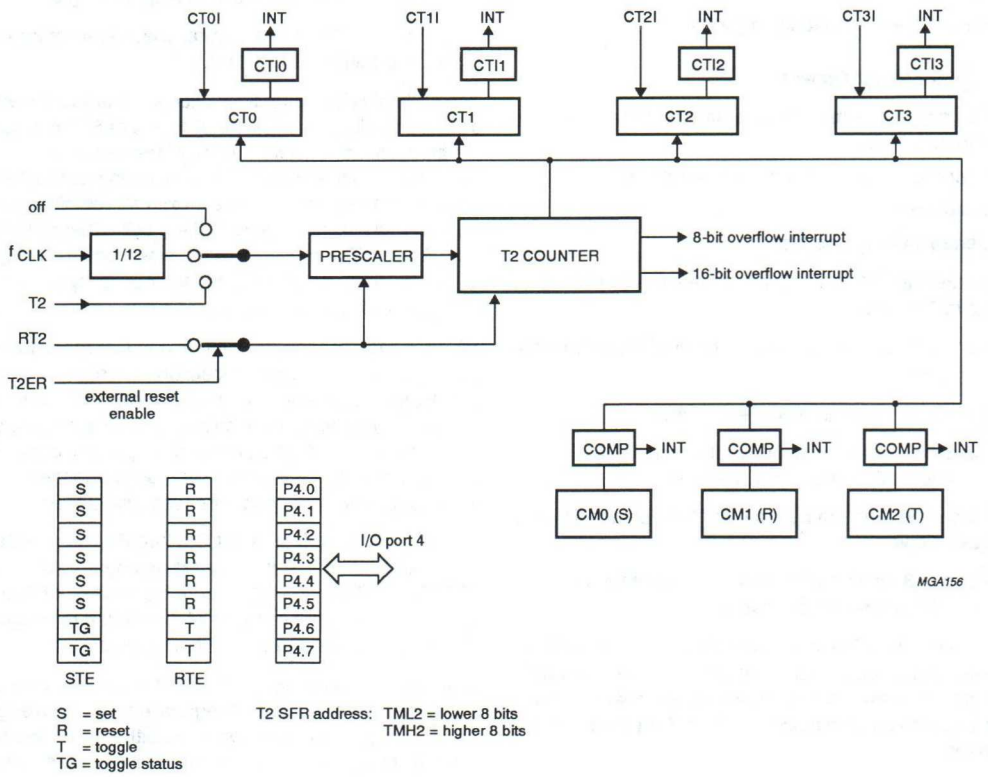



Fig.11 Block diagram of Timer T2 configuration.

### 11.2.1 COUNTER CONTROL REGISTER (TM2CON)

**Table 16** Counter Control register (address EAH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| T2IS1 | T2IS0 | T2ER | T2B0 | T2P1 | T2P0 | T2MS1 | T2MS0 |

**Table 17** Description of the TM2CON bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | T2IS1 | Timer 2 16-bit overflow interrupt select. |
| 6 | T2IS0 | Timer 2 byte overflow interrupt select. |
| 5 | T2ER | Timer 2 external reset enable. |
| 4 | T2B0 | Timer 2 byte overflow interrupt flag. |
| 3 | T2P1 | Timer 2 prescaler select (see Table 18). |
| 2 | T2P0 | |
| 1 | T2MS1 | Timer 2 mode select (see Table 19). |
| 0 | T2MS0 | |

**Table 18** Timer 2 prescaler select

| T2P1 | T2P0 | T2 CLOCK |
|---|---|---|
| 0 | 0 | Clock source |
| 0 | 1 | $\frac{1}{2}$ Clock source |
| 1 | 0 | $\frac{1}{4}$ Clock source |
| 1 | 1 | $\frac{1}{8}$ Clock source |

**Table 19** Timer 2 mode select

| T2MS1 | T2MS0 | MODE |
|---|---|---|
| 0 | 0 | Timer T2 is halted |
| 0 | 1 | T2 clock source = $\frac{1}{12}f_{CLK}$. |
| 1 | 0 | Test mode; do not use |
| 1 | 1 | T2 clock source = pin T2 |

### 11.2.2 CAPTURE CONTROL REGISTER (CTCON)

**Table 20** Capture Control register (address EBH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| CTN3 | CTP3 | CTN2 | CTP2 | CTN1 | CTP1 | CTN0 | CTP0 |

**Table 21** Description of the CTCON bits

| BIT | SYMBOL | FUNCTION | |
|---|---|---|---|
| | | CAPTURE | INTERRUPT ON |
| 7 | CTN3 | CT3I | negative edge |
| 6 | CTP3 | CT3I | positive edge |
| 5 | CTN2 | CT2I | negative edge |
| 4 | CTP2 | CT2I | positive edge |
| 3 | CTN1 | CT1I | negative edge |
| 2 | CTP1 | CT1I | positive edge |
| 1 | CTN0 | CT0I | negative edge |
| 0 | CTP0 | CT0I | positive edge |

### 11.2.3 Timer Interrupt Flag register (TM2IR)

**Table 22** Timer Interrupt Flag register (address C8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| T2OV | CMI2 | CMI1 | CMI0 | CTI3 | CTI2 | CTI1 | CTI0 |

**Table 23** Description of the TM2IR bits (see notes 1 and 2)

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | T2OV | T2: 16-bit overflow interrupt flag |
| 6 | CMI2 | CM2: interrupt flag |
| 5 | CMI1 | CM1: interrupt flag |
| 4 | CMI0 | CM0: interrupt flag |
| 3 | CTI3 | CT3: interrupt flag |
| 2 | CTI2 | CT2: interrupt flag |
| 1 | CTI1 | CT1: interrupt flag |
| 0 | CTI0 | CT0: interrupt flag |

**Notes**

1. Interrupt Enable IEN1 is used to enable/disable Timer 2 interrupts (see Section 14.1.2).
2. Interrupt Priority Register IP1 is used to determine the Timer 2 interrupt priority (see Section 14.1.4).

### 11.2.4 Set Enable register (STE)

**Table 24** Set Enable register (address EEH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| TG47 | TG46 | SP45 | SP44 | SP43 | SP42 | SP41 | SP40 |

**Table 25** Description of the STE bits (see notes 1 and 2)

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | TG47 | if HIGH then P4.7 is reset on the next toggle, if LOW P4.7 is set on the next toggle |
| 6 | TG46 | if HIGH then P4.6 is reset on the next toggle, if LOW P4.6 is set on the next toggle |
| 5 | SP45 | if HIGH then P4.5 is set on a match of CM0 and T2 |
| 4 | SP44 | if HIGH then P4.4 is set on a match of CM0 and T2 |
| 3 | SP43 | if HIGH then P4.3 is set on a match of CM0 and T2 |
| 2 | SP42 | if HIGH then P4.2 is set on a match of CM0 and T2 |
| 1 | SP41 | if HIGH then P4.1 is set on a match of CM0 and T2 |
| 0 | SP40 | if HIGH then P4.0 is set on a match of CM0 and T2 |

**Notes**

1. If STE.n is LOW then P4.n is not affected by a match of CM0 and T2 (n = 0, 1, 2, 3, 4, 5).
2. STE.6 and STE.7 are read only.

### 11.2.5 RESET/TOGGLE ENABLE REGISTER (RTE)

**Table 26** Reset/Toggle Enable register (address EFH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| TP47 | TP46 | RP45 | RP44 | RP43 | RP42 | RP41 | RP40 |

**Table 27** Description of the RTE bits (note 1)

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | TP47 | if HIGH then P4.7 toggles on a match of CM2 and T2 |
| 6 | TP46 | if HIGH then P4.6 toggles on a match of CM2 and T2 |
| 5 | RP45 | if HIGH then P4.5 is reset on a match of CM1 and T2 |
| 4 | RP44 | if HIGH then P4.4 is reset on a match of CM1 and T2 |
| 3 | RP43 | if HIGH then P4.3 is reset on a match of CM1 and T2 |
| 2 | RP42 | if HIGH then P4.2 is reset on a match of CM1 and T2 |
| 1 | RP41 | if HIGH then P4.1 is reset on a match of CM1 and T2 |
| 0 | RP40 | if HIGH then P4.0 is reset on a match of CM1 and T2 |

**Note**

1. If RTE.n is LOW then P4.n is not affected by a match of CM1 and T2 or CM2 and T2.
   For more information, refer to the 8051-based *"8-bit Microcontrollers Data Handbook IC20"*.

### 11.3 Watchdog Timer (T3)

In addition to Timer T2 and the standard timers (Timer 0 and Timer 1), a Watchdog Timer (WDT) comprising an 11-bit prescaler and an 8-bit timer (T3) is also provided (see Fig.12).

The timer T3 is incremented every 1.5 ms, derived from the oscillator frequency of 16 MHz by the following formula: $f_{timer} = \frac{f_{CLK}}{12 \times 2048}$

When a timer T3 overflow occurs, the microcontroller is reset and a reset-output-pulse is generated at pin RST. This short output pulse (3 machine cycles) may be suppressed if the RST pin is connected to a capacitor.

To prevent a system reset (by an overflow of the WDT), the user program has to reload T3 within periods that are shorter than the programmed Watchdog time interval.

If the processor suffers a hardware/software malfunction, the software will fail to reload the timer. This failure will produce a reset upon overflow thus preventing the processor running out of control.

The Watchdog Timer can only be reloaded if the condition flag WLE = PCON.4 has been previously set by software. At the moment the counter is loaded the condition flag is automatically cleared.

The timer interval between the timer's reloading and the occurrence of a reset depends on the reloaded value. For example, this may range from 1.5 ms to 0.375 s when using an oscillator frequency of 16 MHz.

In the Idle state the Watchdog Timer and reset circuitry remain active.

The Watchdog Timer (WDT) is controlled by the Enable Watchdog pin ($\overline{EW}$) (see Table 28).

**Table 28** $\overline{EW}$ controlling WDT and Power-down mode

| PIN $\overline{EW}$ | WDT | POWER-DOWN MODE |
|---|---|---|
| LOW | enabled | disabled |
| HIGH | disabled | enabled |

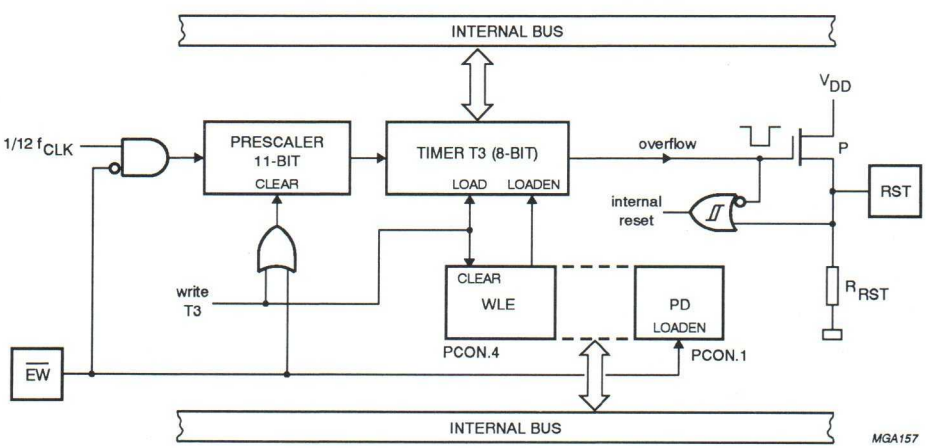



Fig.12 Functional diagram of T3 Watchdog Timer.

## 12 SERIAL I/O PORT: SIO0 (UART)

The Serial Port SIO0 is a full duplex (UART) serial I/O port i.e. it can transmit and receive simultaneously. This Serial Port is also receive-buffered. It can commence reception of a second byte before the previously received byte has been read from the receive register. However, if the first byte has still not been read by the time reception of the second byte is complete, one of these (first or second) bytes will be lost. The SIO0 receive and transmit registers are both accessed via the S0BUF SFR. Writing to S0BUF loads the transmit register, and reading S0BUF accesses to a physically separate receive register. SIO0 can operate in 4 modes:

Mode 0 Serial data is transmitted and received through RXD. TXD outputs the shift clock. 8 data bits are transmitted/received (LSB first). The baud rate is fixed at $1/12$ of the oscillator frequency.

Mode 1 10 bits are transmitted via TXD or received through RXD: a start bit (0), 8 data bits (LSB first), and a stop bit (1). On receive, the stop bit is put into RB8 of the S0CON SFR. The baud rate is variable.

Mode 2 11 bits are transmitted through TXD or received through RXD: a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). On transmit, the 9th data bit (TB8 in S0CON) can be assigned the value of 0 or 1. With nominal software, TB8 can be the parity bit (P in PSW). During a receive, the 9th data bit is stored in RB8 (S0CON), and the stop bit is ignored. The baud rate is programmable to either $1/32$ or $1/64$ of the oscillator frequency.

Mode 3 11 bits are transmitted through TXD or received through RXD: a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). Mode 3 is the same as Mode 2 except for the baud rate which is variable in Mode 3.

In all four modes, transmission is initiated by any instruction that writes to the S0BUF SFR.
Reception is initiated in Mode 0 when RI = 0 and REN = 1. In the other three modes, reception is initiated by the incoming start bit provided that REN = 1.

Modes 2 and 3 are provided for multiprocessor communications. In these modes, 9 data bits are received with the 9th bit written to RB8 (S0CON). The 9th bit is followed by the stop bit. The port can be programmed so that with receiving the stop bit, the Serial Port interrupt will be activated if, and only if RB8 = 1.

This feature is enabled by setting bit SM2 in S0CON. This feature may be used in multiprocessor systems.

For more information about how to use the UART in combination with the registers S0CON, PCON, IE, SBUF and the Timer register, refer to the 8051-based *"8-bit Microcontrollers Data Handbook IC20"*.

## 13 SERIAL I/O PORT: SIO1 (CAN)

SIO1 (CAN) provides the CAN (Controller Area Network) serial-bus data communication interface. SIO1 (CAN) replaces the SIO1 ($I^2C$) serial interface as provided in the microcontroller derivative P8xC552.

### 13.1 On-chip CAN-controller

CAN is the definition of a high performance communication protocol for serial data communication. The P8xC592 on-chip CAN-controller is a full implementation of the CAN 2.0A protocol. With the P8xC592 powerful local networks can be built, both for automotive and general industrial environments. This results in a much reduced wiring harness and enhanced diagnostic and supervisory capabilities.

### 13.2 CAN Features

- Multi-master architecture
- Bus access priority determined by the message identifier
- 2032 message identifier ($2^{11}$ standard frame CAN 2.0A)
- Guaranteed latency time for high priority messages
- Powerful error handling capability
- Data length from 0 up to 8 bytes
- Multicast and broadcast message facility
- Non destructive bit-wise arbitration
- Non-return-to-zero (NRZ) coding/decoding with bit-stuffing
- Programmable transfer rate (up to 1 Mbit/s)
- Programmable output driver configuration
- Suitable for use in a wide range of networks including the SAE's network classes A, B and C
- DMA providing high-speed on-chip data exchange
- Bus failure management facility
- $\frac{1}{2}AV_{DD}$ reference voltage.

### 13.3 Interface between CPU and CAN

The internal interface between the P8xC592's CPU and on-chip CAN-controller is achieved via the following four SFRs (see Fig.13):

- CANADR, to point to a register of the CAN-controller
- CANDAT, to read or write data
- CANCON, to read interrupt flags and to write commands
- CANSTA, to read status information and to write DMA pointer.

Additionally, the DMA-logic allows a high-speed data exchange between the CAN-controller and the CPU's on-chip MAIN RAM. For more information, see Section 13.5.15 "Handling of the CPU-CAN interface".

### 13.4 Hardware blocks of the CAN-controller

The P8xC592 CAN-controller contains all necessary hardware for high performance serial network communications (see Fig.14 and Table 29).

It controls the communication flow through the area network using the CAN-protocol. The CAN-controller meets the following automotive requirements:

- Short message length
- Bus access priority, determined by the message identifier
- Powerful error handling capability
- Configuration flexibility to allow area network expansion
- Guaranteed latency time for urgent messages;
  - The **latency time** defines the period between the initiation (Transmission Request) and the start of the transmission on the bus. The latency time strongly depends on a large variety of bus-related conditions. In the case of a message being transmitted on the bus and one distortion, the latency time can be up to 149 bit times (worst case). For more information see Chapter 22 "CAN application information".

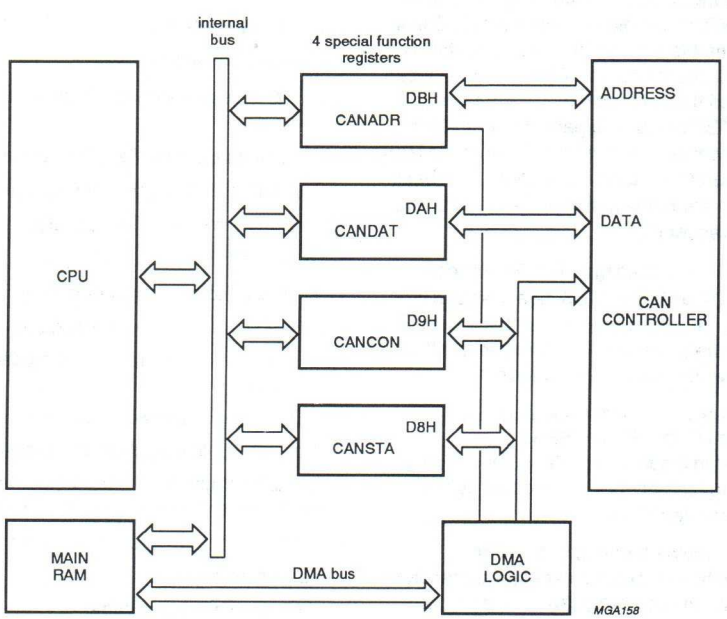



Fig.13 Interface between CPU and CAN-controller.

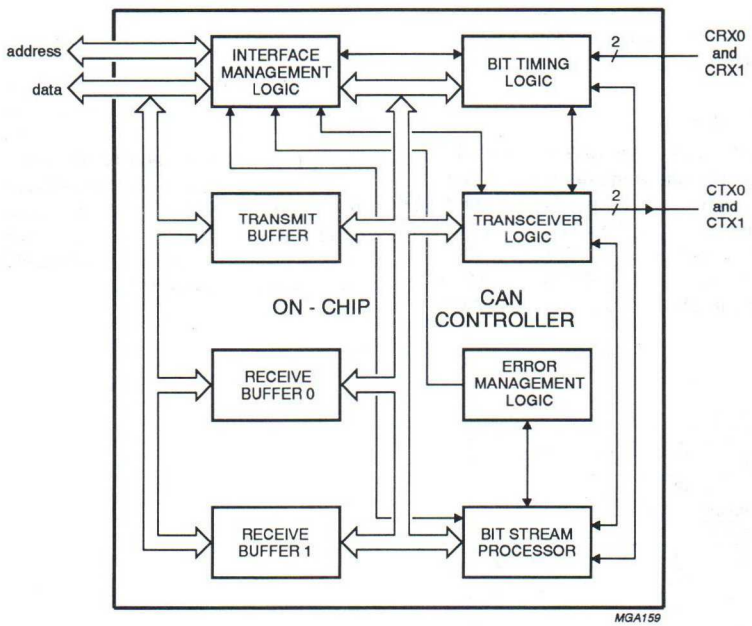



Fig.14 Block diagram of the P8xC592 on-chip CAN-controller.

**Table 29** Hardware blocks of the CAN-controller (see Fig.14)

| NAME | BLOCK | DESCRIPTION |
|---|---|---|
| Interface Management Logic | IML | Interprets commands from the CPU, allocates the message buffers (TBF, RBF0 and RBF1) and provides interrupts and status information to the microcontroller. |
| Transmit Buffer | TBF | 10 bytes memory into which the CPU writes messages which are to be transmitted over the CAN network. |
| Receive Buffers (0 and 1) | RBF0<br>RBF1 | RBF0 and RBF1 are each 10 bytes memories which are alternatively used to store messages received from the CAN network.<br>The CPU can process one message while another is being received. |
| Bit Stream Processor | BSP | Is a sequencer, controlling the data stream between the Transmit Buffer, Receive Buffers (parallel data) and the CAN-bus (serial data). |
| Bit Timing Logic | BTL | Synchronizes the CAN-controller to the bitstream on the CAN-bus. |
| Transceiver Control Logic | TCL | Controls the output driver. |
| Error Management Logic | EML | Performs the error confinement according to the CAN-protocol. |

## 13.5 Control Segment and Message Buffer description

The CAN-controller appears to the CPU as a memory-mapped peripheral, guaranteeing the independent operation of both parts.

### 13.5.1 Address allocation

The address area of the CAN-controller consists of the Control Segment and the message buffers. The Control Segment is programmed during an initialization down-load in order to configure communication parameters (e.g. bit timing). The communication over the CAN-bus is also controlled via this segment by the CPU. A message which is to be transmitted, must be written to the Transmit Buffer. After a successful reception the CPU may read the message from the Receive Buffer and then release it for further use.

### 13.5.2 Control Segment layout

The exchange of status, control and command signals between the CPU and the CAN-controller is performed in the control segment. The layout of this segment is shown in Fig.15. After an initial down-load, the contents of the registers Acceptance Code, Acceptance Mask, Bus Timing 0, Bus Timing 1 and Output Control should not be changed. These registers may only be accessed when the Reset Request bit in the Control Register is set HIGH (see Tables 30, 31 and 32).

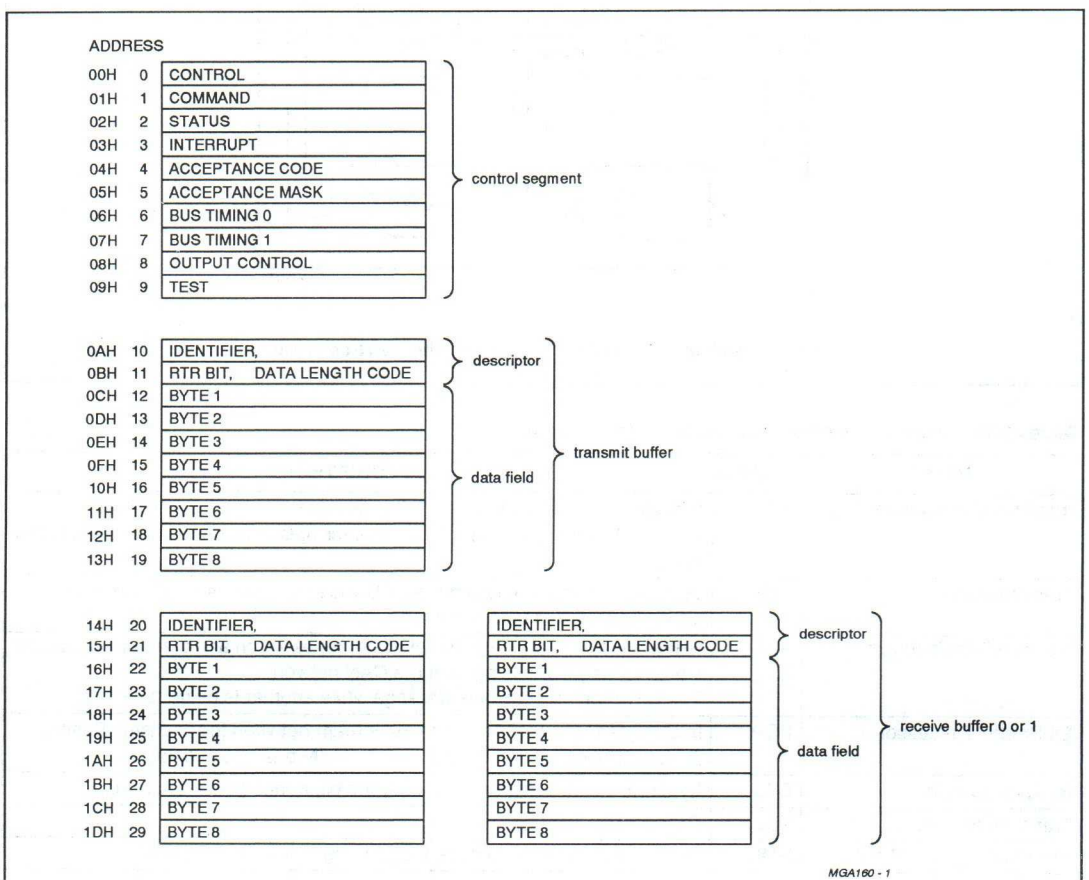



Fig.15 CAN-controller internal address allocation.

**Table 30** CPU/CAN Register map

| BIT | | | | | | | |
|---|---|---|---|---|---|---|---|
| **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| **Control Segment** | | | | | | | |
| ADDRESS 0: CONTROL REGISTER | | | | | | | |
| TM | S | RA | OIE | EIE | TIE | RIE | RR |
| ADDRESS 1: COMMAND REGISTER | | | | | | | |
| RX0A | RX1A | WUM | SLP | COS | RRB | AT | TR |
| ADDRESS 2: STATUS REGISTER | | | | | | | |
| BS | ES | TS | RS | TCS | TBS | DO | RBS |
| ADDRESS 3: INTERRUPT REGISTER | | | | | | | |
| Reserved | Reserved | Reserved | WUI | OI | EI | TI | RI |
| ADDRESS 4: ACCEPTANCE CODE REGISTER | | | | | | | |
| AC.7 | AC.6 | AC.5 | AC.4 | AC.3 | AC.2 | AC.1 | AC.0 |
| ADDRESS 5: ACCEPTANCE MASK REGISTER | | | | | | | |
| AM.7 | AM.6 | AM.5 | AM.4 | AM.3 | AM.2 | AM.1 | AM.0 |
| ADDRESS 6: BUS TIMING REGISTER 0 | | | | | | | |
| SJW.1 | SJW.0 | BRP.5 | BRP.4 | BRP.3 | BRP.2 | BRP.1 | BRP.0 |
| ADDRESS 7: BUS TIMING REGISTER 1 | | | | | | | |
| SAM | TSEG2.2 | TSEG2.1 | TESG2.0 | TSEG1.3 | TSEG1.2 | TSEG1.1 | TSEG1.0 |
| ADDRESS 8: OUTPUT CONTROL REGISTER | | | | | | | |
| OCTP1 | OCTN1 | OCPOL1 | OCTP0 | OCTN0 | OCPOL0 | OCMODE1 | OCMODE0 |
| ADDRESS 9: TEST REGISTER (note 1) | | | | | | | |
| Reserved | Reserved | Map Internal Register | Connect RX Buffer 0 CPU | Connect TX Buffer CPU | Access Internal Bus | Normal RAM Connect | Float Output Driver |

| BIT | | | | | | | |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Transmit Buffer** | | | | | | | |
| ADDRESS 10: IDENTIFIER | | | | | | | |
| ID.10 | ID.9 | ID.8 | ID.7 | ID.6 | ID.5 | ID.4 | ID.3 |
| ADDRESS 11: RTR, DATA LENGTH CODE | | | | | | | |
| ID.2 | ID.1 | ID.0 | RTR | DLC.3 | DLC.2 | DLC.1 | DLC.0 |
| ADDRESS 12 TO 19: BYTES 1 TO 8 | | | | | | | |
| Data | Data | Data | Data | Data | Data | Data | Data |
| **Receive Buffer 0 and 1** | | | | | | | |
| ADDRESS 20: IDENTIFIER | | | | | | | |
| ID.10 | ID.9 | ID.8 | ID.7 | ID.6 | ID.5 | ID.4 | ID.3 |
| ADDRESS 21: RTR, DATA LENGTH CODE | | | | | | | |
| ID.2 | ID.1 | ID.0 | RTR | DLC.3 | DLC.2 | DLC.1 | DLC.0 |
| ADDRESS 22 TO 29: BYTES 1 TO 8 | | | | | | | |
| Data | Data | Data | Data | Data | Data | Data | Data |

**Note**

1. The Test Register is used for production testing only.

### 13.5.3 CONTROL REGISTER (CR)

The contents of the Control Register are used to change the behaviour of the CAN-controller. Control bits may be set or reset by the CPU which uses the Control Register as a read/write memory.

**Table 31** Control Register (address 0)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| TM | S | RA | OIE | EIE | TIE | RIE | RR |

**Table 32** Description of the CR bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | TM | **Test Mode** (note 1).If the value of TM is: |
| | | HIGH (enabled), then the CAN-controller enters Test Mode (normal operations impossible). |
| | | LOW (disabled), then the CAN-controller is in normal operating mode. |
| 6 | S | **Sync** (note 2). If the value of S is: |
| | | HIGH (2 edges), then bus-line transitions from recessive-to-dominant and vice-versa are used for resynchronization (see Sections 13.5.20 and 13.6). |
| | | LOW (1 edge), then the only transitions from recessive-to-dominant are used for resynchronization. |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 5 | RA | **Reference Active** (notes 2). If the value of RA is:<br>HIGH (output), then the pin REF is an $\frac{1}{2}AV_{DD}$ reference output.<br>LOW (input), then a reference voltage may be input. |
| 4 | OIE | **Overrun Interrupt Enable**. If the value of OIE is:<br>HIGH (enabled) and the Data Overrun bit is set (see Section 13.5.5) then the CPU receives an Overrun Interrupt signal.<br>LOW (disabled), then the CPU receives no Overrun Interrupt signal from the CAN-controller. |
| 3 | EIE | **Error Interrupt Enable**. If the value of EIE is:<br>HIGH (enabled) and the Error or Bus Status change (see Section 13.5.5) then the CPU receives an Error Interrupt signal.<br>LOW (disabled), then the CPU receives no Error Interrupt signal. |
| 2 | TIE | **Transmit Interrupt Enable**. If the value of TIE is:<br>HIGH (enabled) and when a message has been successfully transmitted or the Transmit Buffer is accessible again, (e.g. after an Abort Transmission command), then the CAN-controller transmits a Transmit Interrupt signal to the CPU.<br>LOW (disabled), then there is no transmission of the Transmit Interrupt signal by the CAN-controller to the CPU. |
| 1 | RIE | **Receive Interrupt Enable**. If the value of RIE is:<br>HIGH (enabled) and when a message has been received without errors, then the CAN-controller transmits a Receive Interrupt signal to the CPU.<br>LOW (disabled), then there is no transmission of the Receive Interrupt signal by the CAN-controller to the CPU. |
| 0 | RR | **Reset Request** (note 3). If the value of RR is:<br>HIGH (present), then detection of a Reset Request results in the CAN-controller aborting the current transmission/reception of a message entering the reset state synchronously to the system clock ($t_{SCL}$, see Section 13.5.9).<br>LOW (absent), on the HIGH-to-LOW transition of the Reset Request bit, the CAN-controller returns to its normal operating state. |

**Notes to the description of the CR bits**

1. The test mode is intended for factory testing and not for customer use.
2. A modification of the bits Reference Active and Sync is only possible with Reset Request = HIGH (present). It is allowed to set these bits while Reset Request is changed from a HIGH level to a LOW level. After an external reset (pin RST = HIGH) the Reference Active bit is set HIGH (output), the Sync bit is undefined.
3. During an external reset (RST = HIGH) or when the Bus Status bit is set HIGH (Bus-OFF), the IML forces the Reset Request HIGH (present). After the Reset Request bit is set LOW (absent) the CAN-controller will wait for:
   a) One occurrence of the Bus-Free signal (11 recessive bits, see Section 13.6.9.6), if the preceding reset (Reset Request = HIGH) has been caused by an external reset or a CPU initiated reset.
   b) 128 occurrences of Bus-Free, if the preceding reset (Reset Request = HIGH) has been caused by a CAN-controller initiated Bus-OFF, before re-entering the Bus-On mode, see Section 13.6.9.
   c) When Reset Request is set HIGH (present), for whatever reason, the Control, Command, Status and Interrupt bits are affected, see Table 40. The registers at addresses 4 to 8 are only accessible when the Reset Request is set HIGH (present).

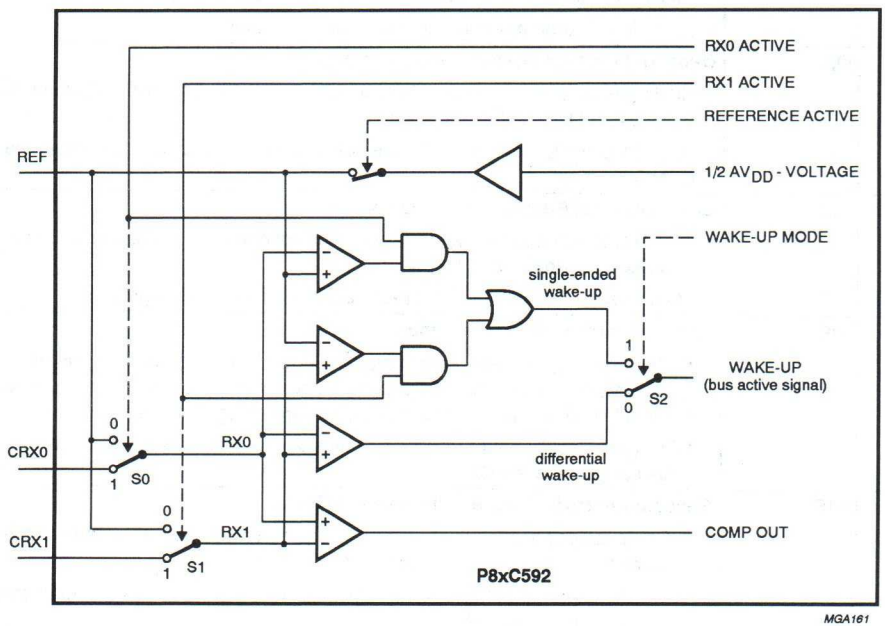



Fig.16 Configurable CAN receiver.

### 13.5.4 COMMAND REGISTER (CMR)

A command bit initiates an action within the transfer layer of the CAN-controller. The Command Register appears to the CPU as a read/write memory, except for the bits CMR.0 (TR) to CMR.3 (COS), which return a HIGH if being read.

**Table 33** Command Register (address 1)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| RX0A | RX1A | WUM | SLP | COS | RRB | AT | TR |

**Table 34** Description of the CMR bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | RX0A | **RX0 Active**. See Table 35; note 1. |
| 6 | RX1A | **RX1 Active**. See Table 35; note 1. |
| 5 | WUM | **Wake-up Mode** (note 2). If the value of WUM is:<br>HIGH (single ended), then the difference of the RX signals to the internal reference voltage $\frac{1}{2}AV_{DD}$ is used for wake up.<br>LOW (differential), then the differential signal between RX0 and RX1 is used for wake up. |
| 4 | SLP | **Sleep** (note 3). If the value of SLP is:<br>HIGH (sleep), then the CAN-controller enters sleep mode if no CAN interrupt is pending and there is no bus activity.<br>LOW (wake up), then the CAN-controller functions normally. |
| 3 | COS | **Clear Overrun Status** (note 4). If the value of COS is:<br>HIGH (clear), then the Data Overrun status bit is set to LOW (see Table 37).<br>LOW (no action), then there is no action. |
| 2 | RRB | **Release Receive Buffer** (note 5). If the value of RRB is:<br>HIGH (released), then the Receive Buffer attached to the CPU is released.<br>LOW (no action), then there is no action. |
| 1 | AT | **Abort Transmission** (note 6). If the value of AT is:<br>HIGH (present) and if not already in progress, a pending Transmission Request is cancelled.<br>LOW (absent), then there is no action. |
| 0 | TR | **Transmission Request** (note 7). If the value of TR is:<br>HIGH (present), then a message shall be transmitted.<br>LOW (absent), then there is no action. |

**Notes to the description of the CMR bits**

1. The RX0/RX1 Active bits, if being read, reflect the status of the respective switches (see Fig.16). It is recommended to change the switches only during the reset state (Reset Request = HIGH).
2. The Wake-Up Mode bit should be set at the same time as the Sleep bit. The differential wake up mode is useful if both bus wires are fully functioning; it minimizes the amount of wake ups due to noise. The single ended wake up mode is recommended if a wake up must be possible even if one bus wire is already or may become disturbed (see Fig.16).
3. The CAN-controller will enter sleep mode, if the Sleep bit is set HIGH (sleep) there is no bus activity and no interrupt is pending. The CAN-controller will wake up after the Sleep bit is set LOW (wake up) or when there is bus activity. On wake up, a Wake-Up Interrupt (see Section 13.5.6) is generated (see also Chapter 15). A CAN-controller which is sleeping and then awaken by bus activity will not be able to receive this message until it detects a Bus-Free signal (see Section 13.6.9.6). The Sleep bit, if read, reflects the status of the CAN-controller.
4. This command bit is used to acknowledge the Data Overrun condition signalled by the Data Overrun status bit. Command is given only after releasing both receive buffers. The stored messages have to be rejected. The command bit is set simultaneously with setting of the Release Receive Buffer command bit the second time.
5. After reading the contents of the Receive Buffer (RBF0 or RBF1) the CPU must release this buffer by setting Release Receive Buffer bit HIGH (released). This may result in another message becoming immediately available. To prevent the RRB command being executed only once, the minimum wait time between two successive RRB commands is 3 system clock cycles ($t_{SCL}$, see Section 13.5.9).
6. The Abort Transmission bit is used when the CPU requires the suspension of the previously requested transmission, e.g. to transmit an urgent message. A transmission already in progress is not stopped. In order to see if the original message had been either transmitted successfully or aborted, the Transmission Complete Status bit should be checked. This should be done after the Transmit Buffer Access bit has been set HIGH (released) or a Transmit Interrupt has been generated (see Section 13.5.6).
7. If the Transmission Request bit was set HIGH in a previous command, it cannot be cancelled by setting the Transmission Request bit LOW (absent). Cancellation of the requested transmission may be performed by setting the Abort Transmission bit HIGH (present).

**Table 35** Combination of bits RX0A and RX1A (see Fig.16)

| CONTROL | | RX0 | RX1 |
|---|---|---|---|
| RX0A | RX1A | | |
| 1 | 1 | CRX0 | CRX1 |
| 1 | 0 | CRX0 | $\frac{1}{2}AV_{DD}$ |
| 0 | 1 | $\frac{1}{2}AV_{DD}$ | CRX1 |
| 0 | 0 | No action | |

### 13.5.5 STATUS REGISTER (SR)

The contents of the Status Register reflects the status of the CAN-controller. The Status Register appears to the CPU as a read only memory.

**Table 36** Status Register (address 2)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| BS | ES | TS | RS | TCS | TBS | DO | RBS |

**Table 37** Description of the SR bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | BS | **Bus Status** (note 1). If the value of BS is:<br>HIGH (Bus-OFF), then the CAN-controller is not involved in bus activities.<br>LOW (Bus-ON), then the CAN-controller is involved in bus activities. |
| 6 | ES | **Error Status**. If the value of ES is:<br>HIGH (error), then at least one of the Error Counters (see Section 13.6.10) has reached the CPU Warning limit.<br>LOW (ok), then both Error Counters have not reached the warning limit. |
| 5 | TS | **Transmit Status** (note 2). If the value of TS is:<br>HIGH (transmit), then the CAN-controller is transmitting a message.<br>LOW (idle), then no message is transmitted. |
| 4 | RS | **Receive Status** (note 2). If the value of RS is:<br>HIGH (receive), then the CAN-controller is receiving a message.<br>LOW (idle), then no message is received. |
| 3 | TCS | **Transmission Complete Status** (note 3). If the value of TCS is:<br>HIGH (complete), then last requested transmission has been successfully completed.<br>LOW (incomplete), then previously requested transmission is not yet completed. |
| 2 | TBS | **Transmit Buffer Access** (note 3). If the value of TBS is:<br>HIGH (released), then the CPU may write a message into the TBF.<br>LOW (locked), then the CPU cannot access the Transmit Buffer. A message is either waiting for transmission or is in the process of being transmitted. |
| 1 | DO | **Data Overrun** (note 4). If the value of DO is:<br>HIGH (overrun), then both Receive Buffers are full and the first byte of another message should be stored.<br>LOW (absent), then no data overrun has occurred since the Clear Overrun command was given. |
| 0 | RBS | **Receive Buffer Status** (note 5). If the value of RBS is<br>HIGH (full), then this bit is set when a new message is available.<br>LOW (empty), then no message has become available since the last Release Receive Buffer command bit was set. |

**Notes to the description of the SR bits**

1. When the Bus Status bit is set HIGH (Bus-OFF), the CAN-controller will set the Reset Request bit HIGH (present). It will stay in this state until the CPU sets the Reset Request bit LOW (absent). Once this is completed the CAN-controller will wait the minimum protocol-defined time (128 occurrences of the Bus-Free signal) before setting the Bus Status bit LOW (Bus-ON), the Error Status bit LOW (ok) and resetting the Error Counters. During Bus-OFF the output drivers are switched off (floating); external transceiver circuits should output a recessive level in this case.
2. If both the Receive Status and Transmit Status bits are LOW (idle) the CAN-bus is idle.
3. If the CPU tries to write to the Transmit Buffer when the Transmit Buffer Access bit is LOW (locked), the written bytes will not be accepted and will be lost without this being signalled. The Transmission Complete Status bit is set LOW (incomplete) whenever the Transmission Request bit is set HIGH (present). If an Abort Transmission command is issued, the Transmit Buffer will be released. If the message, which was requested and then aborted, was not transmitted, the Transmission Complete Status bit will remain LOW.
4. If Data Overrun = HIGH (overrun) is detected, the currently received message is dropped. A transmitted message, granted acceptance, is also stored in a Receive Buffer. This occurs because it is not known if the CAN-controller will lose arbitration and so become a receiver of the message. If no Receive Buffer is available, Data Overrun is signalled. However, this transmitted and accepted message does neither cause a Receive Interrupt nor set the Receive Buffer Status bit to HIGH (full). Also, a Data Overrun does not cause the transmission of an Overload Frame (see Sections 13.6.1 and 13.6.5).
5. If the command bit Release Receive Buffer is set HIGH (released) by the CPU, the Receive Buffer Status bit is set LOW (empty) by IML. When a new message is stored in any of the receive buffers, the Receive Buffer Status bit is set HIGH (full) again.

### 13.5.6 INTERRUPT REGISTER (IR)

The Interrupt Register allows the identification of an interrupt source. When one or more bits of this register are set, a CAN interrupt (SI01) will be indicated to the CPU. All bits are reset by the CAN-controller after this register is read by the CPU. This register appears to the CPU as a read only memory.

**Table 38** Interrupt Register (address 3)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | WUI | OI | EI | TI | RI |

**Table 39** Description of the IR bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | – | Reserved. |
| 6 | – | Reserved. |
| 5 | – | Reserved. |
| 4 | WUI | **Wake-Up Interrupt**. The value of WUI is set to:<br>HIGH (set), when the sleep mode is left. See Section 13.5.4.<br>LOW (reset), by a read access of the Interrupt Register by the CPU. |
| 3 | OI | **Overrun Interrupt** (note 1). The value of OI is set to:<br>HIGH (set), if both Receive Buffers contain a message and the first byte of another message should be stored (passed acceptance), and the Overrun Interrupt Enable is HIGH (enabled).<br>LOW (reset), by a read access of the Interrupt Register by the CPU. |
| 2 | EI | **Error Interrupt**. The value of EI is set to:<br>HIGH (set), on a change of either the Error Status or Bus Status bits, if the Error Interrupt Enable is HIGH (enabled). See Section 13.5.5.<br>LOW (reset), by a read access of the Interrupt Register by the CPU. |
| 1 | TI | **Transmit Interrupt**. The value of TI is set to:<br>HIGH (set), on a change of the Transmit Buffer Access from LOW to HIGH (released) and Transmit Interrupt Enable is HIGH (enabled).<br>LOW (reset), after a read access of the Interrupt Register by the CPU. |
| 0 | RI | **Receive Interrupt** (note 2). The value of RBS is set to:<br>HIGH (set), when a new message is available in the Receive Buffer and the Receive Interrupt Enable bit is HIGH (enabled).<br>LOW (reset) automatically by a read access of Interrupt Register by the CPU. |

**Notes**

1. Overrun Interrupt bit (if enabled) and Data Overrun bit (see Section 13.5.5) are set at the same time.
2. Receive Interrupt bit (if enabled) and Receive Buffer Status bit (see Section 13.5.5) are set at the same time.

**Table 40** Effects of setting the Reset Request bit HIGH (present)

| TYPE | BIT | SYMBOL | FUNCTION | EFFECT |
|---|---|---|---|---|
| Control | CR.7 | TM | Test Mode | LOW (disabled) |
| | CR.5 | RA | Reference Active | HIGH (output); note 1 |
| Command | CMR.7 | RX0A | RX0 Active | HIGH (RX0 = CRX0); note 1 |
| | CMR.6 | RX1A | RX1 Active | HIGH (RX1 = CRX1); note 1 |
| | CMR.4 | SLP | Sleep | LOW (wake-up) |
| | CMR.3 | COS | Clear Overrun Status | HIGH (clear) |
| | CMR.2 | RRB | Release Receive Buffer | HIGH (released) |
| | CMR.1 | AT | Abort Transmission | LOW (absent) |
| | CMR.0 | TR | Transmission Request | LOW (absent) |
| Status | SR.7 | BS | Bus Status | LOW (Bus-On); note 1 |
| | SR.6 | ES | Error Status | LOW (no error); note 1 |
| | SR.5 | TS | Transmit Status | LOW (idle) |
| | SR.4 | RS | Receive Status | LOW (idle) |
| | SR.3 | TCS | Transmission Complete Status | HIGH (complete) |
| | SR.2 | TBS | Transmit Buffer Access | HIGH (released) |
| | SR.1 | DO | Data Overrun | LOW (absent) |
| | SR.0 | RBS | Receive Buffer Status | LOW (empty) |
| Interrupt | IR.3 | OI | Overrun Interrupt | LOW (reset) |
| | IR.1 | TI | Transmit Interrupt | LOW (reset) |
| | IR.0 | RI | Receive Interrupt | LOW (reset) |

**Note**

1. Only after an external reset; see note 5 to Table 37 "Description of the SR bits".

### 13.5.7 Acceptance Code Register (ACR)

The Acceptance Code Register is part of the acceptance filter of the CAN-controller. This register can be accessed (read/write), if the Reset Request bit is set HIGH (present).

When a message is received which passes the acceptance test and if there is an empty Receive Buffer, then the respective Descriptor and Data Field (see Fig.15) are sequentially stored in this empty buffer.

In the event that there is no empty Receive Buffer, the Data Overrun bit is set HIGH (overrun); see Sections 13.5.5 and 13.5.6.

When the complete message has been correctly received the following occurs:

- The Receive Buffer Status bit is set HIGH (full)
- If the Receive Interrupt Enable bit is set HIGH (enabled), the Receive Interrupt is set HIGH (set).

During transmission of a message which passes the acceptance test, the message is also written to its own Receive Buffer. If no Receiver Buffer is available, Data Overrun is signalled because it is not known at the start of a message whether the CAN-controller will lose arbitration and so become a receiver of the message.

**Table 41** Acceptance Code Register (address 4)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| AC.7 | AC.6 | AC.5 | AC.4 | AC.3 | AC.2 | AC.1 | AC.0 |

**Table 42** Description of the ACR bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 to 0 | AC.7 to AC.0 | **Acceptance Code.** The Acceptance Code bits (AC.7 to AC.0) and the eight most significant bits of the message's Identifier (ID.10 to ID.3) must be equal to those bit positions which are marked relevant by the Acceptance Mask bits (AM.7 to AM.0).<br>The acceptance is given, if the following equation is satisfied:<br>(ID10 ... ID.3) = [(AC.7 ... AC.0) or (AM.7 ... AM.0)] = 1111 1111 B. |

### 13.5.8 Acceptance Mask Register (AMR)

The Acceptance Mask Register is part of the acceptance filter of the CAN-controller.
This register can be accessed (read/write) if the Reset Request bit is set HIGH (present).

The Acceptance Mask Register qualifies which of the corresponding bits of the acceptance code are 'relevant' or 'don't care' for acceptance filtering.

**Table 43** Acceptance Mask Register (address 5)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| AM.7 | AM.6 | AM.5 | AM.4 | AM.3 | AM.2 | AM.1 | AM.0 |

**Table 44** Description of the AMR bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 to 0 | AM.7 to AM.0 | **Acceptance Mask.** If the Acceptance Mask bit is:<br>HIGH (don't care), then this bit position is 'don't care' for the acceptance of a message.<br>LOW (relevant), then this bit position is 'relevant' for acceptance filtering. |

### 13.5.9 Bus Timing Register 0 (BTR0)

The contents of Bus Timing Register 0 defines the values of the Baud Rate Prescaler (BRP) and the Synchronization Jump Width (SJW).

This register can be accessed (read/write) if the Reset Request bit is set HIGH (present).
For further information on bus timing, see Sections 13.5.10 and 13.5.18.

**Table 45** Bus Timing Register 0 (address 6)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SJW.1 | SJW.0 | BRP.5 | BRP.4 | BRP.3 | BRP.2 | BRP.1 | BRP.0 |

**Table 46** Description of the BTR0 bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | SJW.1 | **Synchronization Jump Width.** To compensate for phase shifts between clock oscillators of different bus controllers, any bus controller must resynchronize on any relevant signal edge of the current transmission. The synchronization jump width defines the maximum number of clock cycles a bit period may be shortened or lengthened by one resynchronization: $t_{SJW} = t_{SCL}(2SJW.1 + SJW.0 + 1)$. |
| 6 | SJW.0 | |
| 5 | BRP.5 | **Baud Rate Prescaler.** The period of the system clock $t_{SCL}$ is programmable and determines the individual bit timing.The system clock is calculated using the following equation: $t_{SCL} = 2t_{CLK}(32BRP.5 + 16BRP.4 + 8BRP.3 + 4BRP.2 + 2BRP.1 + BRP.0 + 1)$. Where $t_{CLK}$ = time period of the P8xC592 oscillator. |
| 4 | BRP.4 | |
| 3 | BRP.3 | |
| 2 | BRP.2 | |
| 1 | BRP.1 | |
| 0 | BRP.0 | |

### 13.5.10 Bus Timing Register 1(BTR1)

The contents of Bus Timing Register 1 defines the length of the bit period, the location of the sample point and the number of samples to be taken at each sample point.

This register can be accessed (read/write) if the Reset Request bit is set HIGH (present).For further information on bus timing, see Sections 13.5.9 and 13.5.18.

**Table 47** Bus Timing Register 1 (address 7)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SAM | TSEG2.2 | TSEG2.1 | TSEG2.0 | TSEG1.3 | TSEG1.2 | TSEG1.1 | TSEG1.0 |

**Table 48** Description of the BTR1 bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | SAM | **Sampling**. If the bit SAM is:<br>HIGH (3 samples), then three samples are taken. This is recommended for slow/medium speed buses (SAE class A and B) where filtering of spikes on the bus-line is beneficial (see Section 13.5.19.6)<br>LOW (1 sample), the bus is sampled once. This is recommended for high speed buses (SAE class C). |
| 6 | TSEG2.2 | **Time Segment 1 (TSEG1) and Time Segment 2 (TSEG2)**. |
| 5 | TSEG2.1 | TSEG1 determines the number of clock cycles per bit period and the location of the sample point |
| 4 | TSEG2.0 | $t_{TSEG1} = t_{SCL}(8TSEG1.3 + 4TSEG1.2 + 2TSEG1.1 + TSEG1.0 + 1)$. |
| 3 | TSEG1.3 | TSEG2 determines the number of clock cycles per bit period and the location of the sample point: |
| 2 | TSEG1.2 | $t_{TSEG2} = t_{SCL}(4TSEG2.2 + 2TSEG2.1 + TSEG2.0 + 1)$. |
| 1 | TSEG1.1 | |
| 0 | TSEG1.0 | |

### 13.5.11 Output Control Register (OCR)

The Output Control Register allows, under software control, the set-up of different output driver configurations. This register can be accessed (read/write) if the Reset Request bit is set HIGH (present). If the CAN-controller is in the sleep mode (Sleep = HIGH) a recessive level is output on the CTX0 and CTX1 pins. If the CAN-controller is in the reset state (Reset Request = HIGH) the output drivers are floating.

Tables 50 and 51, show the relationship between the bits of the Output Control Register and the two serial output pins CTX0 and CTX1 of the P8xC592 CAN-controller, connected to the serial bus (see Fig.14).

**Table 49** Output Control Register (address 8)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| OCTP1 | OCTN1 | OCPOL1 | OCTP0 | OCTN0 | OCPOL0 | OCMODE1 | OCMODE0 |

**Table 50** Description of the OCR bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | OCTP1 | See Tables 51 and 52. |
| 6 | OCTN1 | |
| 5 | OCPOL1 | |
| 4 | OCTP0 | |
| 3 | OCTN0 | |
| 2 | OCPOL0 | |
| 1 | OCMODE1 | **Output Mode.** |
| 0 | OCMODE0 | These bits select the output mode; see Table 51. |

**Table 51** Description of the Output Mode bits

| OCMODE1 | OCMODE0 | DESCRIPTION |
|---|---|---|
| 1 | 0 | **Normal Output Mode**. The bit sequence (TXD) is sent via CTX0, CTX1. TXD is the data bit to be transmitted. The voltage levels on the output driver pins CTX0 and CTX1 depend on both the driver characteristic programmed by OCTPx, OCTNx (float, pull-up, pull-down, push-pull) and the output polarity programmed by OCPOLx (see Fig.17). |
| 1 | 1 | **Clock Output Mode**. For the CTX0 pin this is the same as in Normal Output Mode (CTX0: bit sequence). However, the data stream to CTX1 is replaced by the transmit clock (TXCLK). The rising edge of the transmit clock (non-inverted) marks the beginning of a bit period. The clock pulse width is $t_{SCL}$. |
| 0 | 0 | **Bi-phase Output Mode**. In contrast to Normal Output Mode the bit representation is time variant and toggled. If the bus controllers are galvanically decoupled from the bus-line by a transformer, the bit stream is not allowed to contain a DC component. This is achieved by the following scheme. During recessive bits all outputs are deactivated (floating). Dominant bits are sent alternately on CTX0 and CTX1, i.e. the first dominant bit is sent on CTX0, the second is sent on CTX1, and the third one is sent on CTX0 again, etc. |
| 0 | 1 | **Test Output Mode.** For the CTX0 pin this is the same as in Normal Output Mode (CTX0: bit sequence). To measure the delay time of the transmitter and receiver this mode connects the output of the input comparator (COMP OUT) with the input of the output driver CTX1. This mode is used for production testing only. |

**Table 52** Output pin set-up

| DRIVE | OCTPx | OCTNx | OCPOLx | TXD | TPx(1) | TNx(2) | CTXx(3) |
|---|---|---|---|---|---|---|---|
| Float | 0 | 0 | 0 | 0 | OFF | OFF | float |
| | 0 | 0 | 0 | 1 | OFF | OFF | float |
| | 0 | 0 | 1 | 0 | OFF | OFF | float |
| | 0 | 0 | 1 | 1 | OFF | OFF | float |
| Pull-down | 0 | 1 | 0 | 0 | OFF | ON | LOW |
| | 0 | 1 | 0 | 1 | OFF | OFF | float |
| | 0 | 1 | 1 | 0 | OFF | OFF | float |
| | 0 | 1 | 1 | 1 | OFF | ON | LOW |
| Pull-up | 1 | 0 | 0 | 0 | OFF | OFF | float |
| | 1 | 0 | 0 | 1 | ON | OFF | HIGH |
| | 1 | 0 | 1 | 0 | ON | OFF | HIGH |
| | 1 | 0 | 1 | 1 | OFF | OFF | float |
| Push/Pull | 1 | 1 | 0 | 0 | OFF | ON | LOW |
| | 1 | 1 | 0 | 1 | ON | OFF | HIGH |
| | 1 | 1 | 1 | 0 | ON | OFF | HIGH |
| | 1 | 1 | 1 | 1 | OFF | ON | LOW |

**Notes**

1. TPx is the on-chip output transistor x, connected to $V_{DD}$; x = 0 or 1.
2. TNx is the on-chip output transistor x, connected to $CV_{SS}$; x = 0 or 1.
3. CTXx is the serial output level on CTX0 or CTX1. It is required that the output level on the CAN-bus is dominant with TXD = 0 and recessive with TXD = 1, see Section 13.6.1.1 "Bit representation".

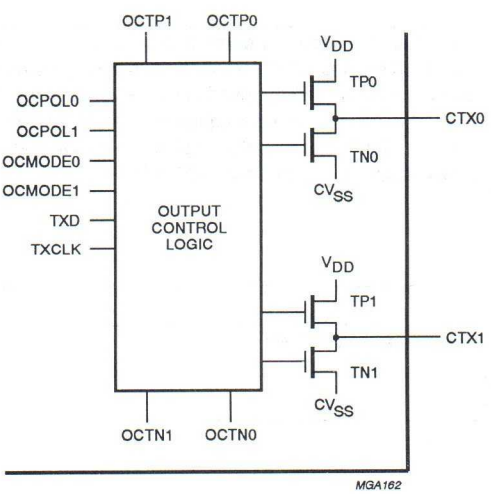



Fig.17 Configurable CAN Transmitter.

### 13.5.12 TEST REGISTER (TR)

The Test Register is used for production testing only.

**Table 53** Test Register (address 9)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| Reserved | Reserved | Map Internal Register | Connect RX Buffer 0 CPU | Connect TX Buffer CPU | Access Internal Bus | Normal RAM Connect | Float Output Driver |

### 13.5.13 TRANSMIT BUFFER LAYOUT

The global layout of the Transmit Buffer is shown in Fig.15. This buffer serves to store a message from the CPU to be transmitted by the CAN-controller. It is subdivided into Descriptor and Data Field. The Transmit Buffer can be written to and read from by the CPU.

#### *13.5.13.1 Descriptor*

**Table 54** Descriptor Byte 1 Register (DSCR1, address 10)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| ID.10 | ID.9 | ID.8 | ID.7 | ID.6 | ID.5 | ID.4 | ID.3 |

**Table 55** Descriptor Byte 2 Register (DSCR2, address 11)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| ID.2 | ID.1 | ID.0 | RTR | DLC.3 | DLC.2 | DLC.1 | DLC.0 |

**Table 56** Description of the ID.n bits in DSCR1 and DSCR2

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| **DSCR1** | | |
| 7 | ID.10 | **Identifier.** The Identifier consists of 11 bits (ID.10 to ID.0). ID.10 is the most significant bit, which is transmitted first on the bus during the arbitration process. The Identifier acts as the messages' name, used in a receiver for acceptance filtering, and also determines the bus access priority during the arbitration process. The lower the binary value of the Identifier the higher the priority. This is due to the larger number of leading dominant bits during arbitration (see Section 13.6.7). |
| 6 | ID.9 | |
| 5 | ID.8 | |
| 4 | ID.7 | |
| 3 | ID.6 | |
| 2 | ID.5 | |
| 1 | ID.4 | |
| 0 | ID.3 | |
| **DSCR2** | | |
| 7 | ID.2 | **Identifier**. See DSCR1. |
| 6 | ID.1 | |
| 5 | ID.0 | |

**Table 57** Description of the other DSCR2 bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 4 | RTR | **Remote Transmission Request**. If the RTR bit is:<br>HIGH (remote), then the Remote Frame will be transmitted by the CAN-controller.<br>LOW (data), then the Data Frame will be transmitted by the CAN-controller. |
| 3 | DLC.3 | **Data Length Code (DLC).** The number of bytes (Data Byte Count) in the Data Field of a message is coded by the Data Length Code. At the start of a Remote Frame transmission the Data Length Code is not considered due to the RTR bit being HIGH (remote). This forces the number of transmitted/received data bytes to be a logic 0. Nevertheless, the Data Length Code must be specified correctly to avoid bus errors, if two CAN-controllers start a Remote Frame transmission simultaneously. The range of the Data Byte Count is 0 to 8 bytes and coded as follows:<br>Data Byte Count = 8DLC.3 + 4DLC.2 + 2DLC.1 + DLC.0.<br>For reasons of compatibility no Data Byte Counts other than 0,1,2,....8 should be used. |
| 2 | DLC.2 | |
| 1 | DLC.1 | |
| 0 | DLC.0 | |

*13.5.13.2 Data Field*

The number of transferred data bytes is determined by the Data Length Code. The first bit transmitted is the most significant bit of data byte 1 at address 12.

13.5.14 RECEIVE BUFFER LAYOUT

The layout of the Receive Buffer and the individual bytes correspond to the definitions given for the Transmit Buffer layout, except that the addresses start at 20 instead of 10 (see Fig.15).

13.5.15 HANDLING OF THE CPU-CAN INTERFACE

Via the four special registers CANADR, CANDAT, CANCON and CANSTA the CPU has access to the CAN-controller and also to the DMA-logic. Note that CANCON and CANSTA have different meanings for a Read and Write access.

**Table 58** The SFRs between CPU and CAN

Reserved bits are read as HIGH. R = Read; W = Write; R/W = Read/Write.

| ADDRESS | ACCESS | BIT | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **CANADR** | | | | | | | | | |
| DBH | R/W | DMA | Reserved | AutoInc | CANA4 | CANA3 | CANA2 | CANA1 | CANA0 |
| **CANDAT** | | | | | | | | | |
| DAH | R/W | CAND7 | CAND6 | CAND5 | CAND4 | CAND3 | CAND2 | CAND1 | CAND0 |
| **CANCON; Do not use a RMW instruction** | | | | | | | | | |
| D9H | R | Reserved | Reserved | Reserved | WUI | OI | EI | TI | RI |
| | W | RX0A | RX1A | WUM | SLP | COS | RRB | AT | TR |
| **CANSTA; The bit addresses of CANSTA (7 to 0) are DFH to D8H; do not use a RMW instruction** | | | | | | | | | |
| DFH to D8H | R | BS | ES | TS | RS | TCS | TBS | DO | RBS |
| | W | RAMA7 | RAMA6 | RAMA5 | RAMA4 | RAMA3 | RAMA2 | RAMA1 | RAMA0 |

*13.5.15.1 Special Function Register CANADR*

CANADR is implemented as a read/write register.

**Table 59** SFR CANADR (address DBH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| DMA | – | AutoInc | CANA4 | CANA3 | CANA2 | CANA1 | CANA0 |

**Table 60** Description of the CANADR bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | DMA | DMA-logic controlled via bit CANADR.7 (see Section 13.5.17). |
| 6 | – | Reserved. |
| 5 | AutoInc | Auto Address Increment mode controlled via bit CANADR.5 (see Section 13.5.16). |
| 4 | CANA4 | The five least significant bits CANADR.4 to CANADR.0 define the address of one of the CAN-controller internal registers to be accessed via CANDAT. For instance, after an external hardware (e.g. power-on) reset CANADR contains the value 64H, and hence the CPU accesses (read/write) the Acceptance Code register of the CAN-controller, via the SFR CANDAT. |
| 3 | CANA3 | |
| 2 | CANA2 | |
| 1 | CANA1 | |
| 0 | CANA0 | |

*13.5.15.2 Special Function Register CANDAT*

CANDAT is implemented as a read/write register.

**Table 61** SFR CANDAT (address DAH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| CAND7 | CAND6 | CAND5 | CAND4 | CAND3 | CAND2 | CAND1 | CAND0 |

**Table 62** Description of the CANADR bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 to 0 | CAND7 to CAND0 | The SFR CANDAT appears as a port to the CAN-controller internal register (memory location) being selected by CANADR. Reading or writing CANDAT is effectively an access to that CAN-controller internal register, which is selected by CANADR. |

*13.5.15.3 Special Function Register CANCON*

**Table 63** SFR CANCON in **Read** access (address D9H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | WUI | OI | EI | TI | RI |

**Table 64** Description of the CANCON bits in **Read** access

When reading CANCON the Interrupt Register of the CAN-controller is accessed.

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | – | Reserved; bits are read as HIGH. |
| 6 | – | |
| 5 | – | |
| 4 | WUI | Wake-Up Interrupt (see Table 39). |
| 3 | OI | Overrun Interrupt (see Table 39). |
| 2 | EI | Error Interrupt (see Table 39). |
| 1 | TI | Transmit Interrupt (see Table 39). |
| 0 | RI | Receive Interrupt (see Table 39). |

**Table 65** SFR CANCON in **Write** access (address D9H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| RX0A | RX1A | WUM | SLP | COS | RRB | AT | TR |

**Table 66** Description of the CANCON bits in **Write** access

When writing to CANCON then the Command Register of the CAN-controller is accessed.

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | RX0A | RX0 Active (see Table 34). |
| 6 | RX1A | RX1 Active (see Table 34). |
| 5 | WUM | Wake-Up Mode (see Table 34). |
| 4 | SLP | Sleep (see Table 34). |
| 3 | COS | Clear Overrun Status (see Table 34). |
| 2 | RRB | Release Receive Buffer (see Table 34). |
| 1 | AT | Abort Transmission (see Table 34). |
| 0 | TR | Transmission Request (see Table 34). |

*13.5.15.4 Special Function Register CANSTA*

CANSTA is implemented as a bit-addressable read/write register.
The bit addresses of CANSTA (7 to 0) are DFH to D8H.

**Table 67** SFR CANCON in **Read** access (address DFH to D8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| BS | ES | TS | RS | TCS | TBS | DO | RBS |

**Table 68** Description of the CANCON bits in **Read** access
When reading CANSTA the Status Register of the CAN-controller is accessed.

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | BS | Bus Status (see Table 37). |
| 6 | ES | Error Status (see Table 37). |
| 5 | TS | Transmit Status (see Table 37). |
| 4 | RS | Receive Status (see Table 37). |
| 3 | TCS | Transmission Complete Status (see Table 37). |
| 2 | TBS | Transmit Buffer Access (see Table 37). |
| 1 | DO | Data Overrun (see Table 37). |
| 0 | RBS | Receive Buffer Status (see Table 37). |

**Table 69** SFR CANCON in **Write** access (address DFH to D8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| RAMA7 | RAMA6 | RAMA5 | RAMA4 | RAMA3 | RAMA2 | RAMA1 | RAMA0 |

**Table 70** Description of the CANSTA bits in **Write** access
Writing to CANSTA sets the address of the on-chip MAIN RAM (internal Data Memory) for a subsequent DMA transfer.

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 to 0 | RAMA7 to RAMA0 | – |

#### 13.5.16 AUTO ADDRESS INCREMENT

With the Auto Address Increment mode a fast stack-like reading and writing of CAN-controller internal registers is provided. If the bit CANADR.5 (AutoInc) is HIGH, the content of CANADR is incremented automatically after any read or write access to CANDAT. For instance, loading a message into the Transmit Buffer can be done by writing 2AH into CANADR and then moving byte by byte of the message to CANDAT. Incrementing CANADR beyond XX111111B resets the bit CANADR.5 (AutoInc) automatically (CANADR = XX000000B).

#### 13.5.17 HIGH SPEED DMA

The DMA-logic allows you to transfer a complete message (up to 10 bytes) between CAN-controller and MAIN RAM in 2 instruction cycles at maximum; up to 4 bytes are transferred in 1 instruction cycle. The performance of the CPU is strongly enhanced because this very fast transfer is carried out in the background.

A DMA transfer is achieved by first writing the RAM address (00H to FFH) into CANSTA and then setting the TX- or RX-Buffer address in CANDR and the bit CANADR.7 (DMA) simultaneously; the RAM address points to the location of the first byte to be transferred. Setting the DMA bit causes an automatic evaluation of the Data Length Code and then the transfer; for a TX-DMA transfer the Data Length Code is expected at the location 'RAM address +1'.

In order to program a TX-DMA transfer the value 8AH (address 10) has to be written into CANADR. Then a complete message, consisting of the 2-byte Descriptor and the Data Field (0 to 8 bytes), starting at location 'RAM address' is transferred to the TX-Buffer.

The RX-DMA transfer is very versatile. By writing a value in the range of 94H (address 20) up to 9DH (address 29) into CANADR the whole or a part of the received message, starting at the specified address, is transferred to the internal Data Memory. This allows e.g. to transfer the bytes of the Data Field only.

After a successful DMA transfer the DMA-bit is reset.

During a DMA transfer the CPU can process the next instruction. However, an access to the Data Memory, CANADR, CANDAT, CANCON or CANSTA is not allowed. After having set the DMA-bit, every interrupt is disabled until the end of the transfer. Note, that disadvantageous programming may lead to an interrupt response time of at most 10 instruction cycles. The shortest interrupt response time is achieved by using 2 consecutive 1-cycle instructions directly after setting the DMA-bit.

During the reset state (bit Reset Request is HIGH) a DMA transfer is not possible.

#### 13.5.18 BUS TIMING/SYNCHRONIZATION

The Bus Timing Logic (BTL) monitors the serial bus-line via the on-chip input comparator and performs the following functions (see Section 13.4):

- Monitors the serial bus-line level
- Adjusts the sample point, within a bit period (programmable)
- Samples the bus-line level using majority logic (programmable, 1 or 3 samples)
- Synchronization to the bit stream:
  - hard synchronization at the start of a message
  - resynchronization during transfer of a message.

The configuration of the BTL is performed during the initialization of the CAN-controller. The BTL uses the following three registers:

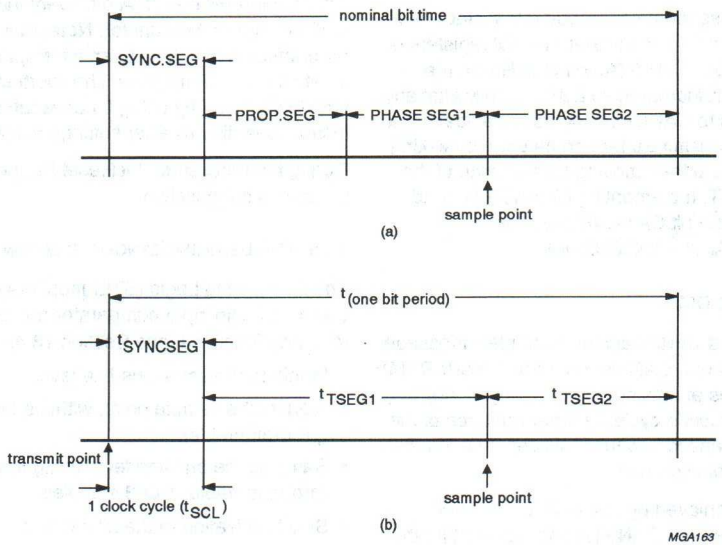
- Control Register (Sync)
- Bus Timing Register 0
- Bus Timing Register 1.

#### 13.5.19 BIT TIMING

A bit period is built up from a number of system clock cycles ($t_{SCL}$), see Section 13.5.9.
One bit period is the result of the addition of the programmable segments TSEG1 and TSEG2 and the general segment SYNCSEG.

##### *13.5.19.1 Synchronization Segment (SYNCSEG)*

The incoming edge of a bit is expected during this state; this state corresponds to one system clock cycle ($1 \times t_{SCL}$).




(a) As defined by the CAN-protocol.
(b) As implemented in the P8xC592's on-chip CAN-controller.

Fig.18 Bit period.

*13.5.19.2 Time Segment 1 (TSEG1)*

This segment determines the location of the sampling point within a bit period, which is at the end of TSEG1. TSEG1 is programmable from 1 to 16 system clock cycles (see Section 13.5.10).

The correct location of the sample point is essential for the correct functioning of a transmission. The following points must be taken into consideration:

- A Start-Of-Frame (see Section 13.6.2) causes all CAN-controllers to perform a 'hard synchronization' (see Section 13.5.20) on the first recessive-to-dominant edge.
  During arbitration, however, several CAN-controllers may simultaneously transmit. Therefore it may require twice the sum of bus-line, input comparator and the output driver delay times until the bus is stable.
  This is the propagation delay time.
- To avoid sampling at an incorrect position, it is necessary to include an additional synchronization buffer on both sides of the sample point.
  The main reasons for incorrect sampling are:
  - Incorrect synchronization due to spikes on the bus-line
  - Slight variations in the oscillator frequency of each CAN-controller in the network, which results in a phase error.
- Time Segment 1 consists of the segment for compensation of propagation delays and the synchronization buffer segment directly before the sample point (see Fig.18).

*13.5.19.3 Time Segment 2 (TSEG2)*

This time segment provides:

- Additional time at the sample point for calculation of the subsequent bit levels (e.g. arbitration)
- Synchronization buffer segment directly after the sample point.

TSEG2 is programmable from 1 to 8 system clock cycles (see Section 13.5.10).

*13.5.19.4 Synchronisation Jump Width (SJW)*

SJW defines the maximum number of clock cycles ($t_{SCL}$) a period may be reduced or increased by one resynchronization. SJW is programmable from 1 to 4 system clock cycles, see Section 13.5.2.

*13.5.19.5 Propagation Delay Time ($t_{prop}$)*

The Propagation Delay Time is:

$t_{prop}$ = 2 × (physical bus delay + input comparator delay + output driver delay).

$t_{prop}$ is rounded up to the nearest multiple of $t_{SCL}$.

*13.5.19.6 Bit Timing Restrictions*

Restrictions on the configuration of the bit timing are based on internal processing. The restrictions are:

- $t_{TSEG2} \geq 2t_{SCL}$
- $t_{TSEG2} \geq t_{SJW}$
- $t_{TSEG1} \geq t_{SEG2}$
- $t_{TSEG1} \geq t_{SJW} + t_{prop}$.

The three sample mode (SAM = HIGH) has the effect of introducing a delay of one system clock cycle on the bus-line. This must be taken into account for the correct calculation of TSEG1 and TSEG2:

- $t_{TSEG1} \geq t_{SJW} + t_{prop} + 2t_{SCL}$
- $t_{TSEG2} \geq 3t_{SCL}$.

13.5.20 SYNCHRONIZATION

Synchronization is performed by a state machine which compares the incoming edge with its actual bit timing and adapts the bit timing by hard synchronization or resynchronization.

This type of synchronization occurs only at the beginning of a message.

The CAN-controller synchronizes on the first incoming recessive-to-dominant edge of a message (being the leading edge of a message's Start-Of-Frame bit; see Section 13.6.2.

Resynchronization occurs during the transmission of a message's bit stream to compensate for:

- Variations in individual CAN-controller oscillator frequencies
- Changes introduced by switching from one transmitter to another (e.g. during arbitration).

As a result of resynchronization either $t_{TSEG1}$ may be increased by up to a maximum of $t_{SJW}$ or $t_{TSEG2}$ may be decreased by up to a maximum of $t_{SJW}$:

- $t_{TSEG1} \leq t_{SCL}$ [(TSEG1 + 1) + (SJW + 1)]
- $t_{TSEG2} \geq t_{SCL}$ [(TSEG2 + 1) – (SJW + 1)].

TSEG1, TSEG2 and SJW are the programmed numerical values.

The phase error (e) of an edge is given by the position of the edge relative to SYNCSEG, measured in system clock cycles ($t_{SCL}$).

The value of the phase error is defined as:

- e = 0, if the edge occurs within SYNCSEG
- e > 0, if the edge occurs within TSEG1
- e < 0, if the edge occurs within TSEG2.

The effect of resynchronization is:

- The same as that of a hard synchronization, if the magnitude of the phase error (e) is less or equal to the programmed value of $t_{SJW}$
- To increase a bit period by the amount of $t_{SJW}$, if the phase error is positive and the magnitude of the phase error is larger than $t_{SJW}$
- To decrease a bit period by the amount of $t_{SJW}$, if the phase error is negative and the magnitude of the phase error is larger than $t_{SJW}$.

#### *13.5.20.1 Synchronization Rules*

The synchronization rules are as follows:

- Only one synchronization within one bit time is used.
- An edge is used for synchronization only if the value detected at the previous sample point differs from the bus value immediately after the edge.
- Hard synchronization is performed whenever there is a recessive-to-dominant edge during Bus-Idle (see Section 13.6.6).
- All other edges (recessive-to-dominant and optionally dominant-to recessive edges if the Sync bit is set HIGH (see Section 13.5.3) which are candidates for resynchronization will be used with the following exception:
  - A transmitting CAN-controller will not perform a resynchronization as a result of a recessive-to-dominant edge with positive phase error, if only these edges are used for resynchronization. This ensures that the delay times of the output driver and input comparator do not cause a permanent increase in the bit time.

## 13.6 CAN 2.0A Protocol description

### 13.6.1 FRAME TYPES

The P8xC592's CAN-controller supports the four different CAN-protocol frame types for communication:

- Data Frame, to transfer data
- Remote Frame, request for data
- Error Frame, globally signal a (locally) detected error condition
- Overload Frame, to extend delay time of subsequent frames (an Overload Frame is not initiated by the P8xC592 CAN-controller).

#### *13.6.1.1 Bit representation*

There are two logical bit representations used in the CAN-protocol:

- A recessive bit on the bus-line appears only if all connected CAN-controllers send a recessive bit at that moment.
- Dominant bits always overwrite recessive bits i.e. the resulting bit level on the bus-line is dominant.

### 13.6.2 DATA FRAME

A Data Frame carries data from a transmitting CAN-controller to one or more receiving ones.
A Data Frame is composed of seven different bit-fields:

- Start-Of-Frame
- Arbitration Field
- Control Field
- Data Field (may have a length of zero)
- CRC Field (CRC = Cyclic Redundancy Code)
- Acknowledge Field
- End-Of-Frame.

#### *13.6.2.1 Start-Of-Frame bit*

Signals the start of a Data Frame or Remote Frame. It consists of a single dominant bit use for hard synchronization of a CAN-controller in receive mode.

#### *13.6.2.2 Arbitration Field*

Consists of the message Identifier and the RTR bit. In the case of simultaneous message transmissions by two or more CAN-controllers the bus access conflict is solved by bit-wise arbitration, which is active during the transmission of the Arbitration Field.

#### *13.6.2.3 Identifier*

This 11-bit field is used to provide information about the message, as well as the bus access priority. It is transmitted in the order ID.10 to ID.0 (LSB). The situation that the seven most significant bits (ID.10 to ID.4) are all recessive must not occur.

An Identifier does not define which particular CAN-controller will receive the frame because a CAN based communication network does not differentiate between a point-to-point, multicast or broadcast communication.

#### *13.6.2.4 RTR bit*

A CAN-controller, acting as a receiver for certain information may initiate the transmission of the respective data by transmitting a Remote Frame to the network, addressing the data source via the Identifier and setting the RTR bit HIGH (remote; recessive bus level). If the data source simultaneously transmits a Data Frame containing the requested data, it uses the same Identifier. No bus access conflict occurs due to the RTR bit being set LOW (data; dominant bus level) in the Data Frame.

#### *13.6.2.5 Control Field*

This field consists of six bits. It includes two reserved bits (for future expansions of the CAN-protocol), transmitted with a dominant bus level, and is followed by the Data Length Code (4 bits).
The number of bytes (destuffed; number of data bytes to be transmitted/received) in the Data Field is indicated by the Data Length Code. Admissible values of the Data Length Code, and hence the number of bytes in the (destuffed) Data Field, are {0, 1, ...., 8}. A logic 0 (logic 1) in the Data Length Code is transmitted as dominant (recessive) bus level, respectively.

#### *13.6.2.6 Data Field*

The data, stored within the Data Field of the Transmit Buffer, are transmitted according to the Data Length Code. Conversely, data of a received Data Frame will be stored in the Data Field of a Receive Buffer. The Data Field can contain from 0 up to 8 bytes. The most significant bit of the first data byte (lowest address) is transmitted/received first.

#### *13.6.2.7 Cyclic Redundancy Code Field (CRC)*

The CRC Field consists of the CRC Sequence (15 bits) and the CRC Delimiter (1 recessive bit). The Cyclic Redundancy Code (CRC) encloses the destuffed bit stream of the Start-Of-Frame, Arbitration Field, Data Field and CRC Sequence. The most significant bit of the CRC Sequence is transmitted/received first. This frame check sequence, implemented in the CAN-controller is derived from a cyclic redundancy code best suited for frames with a total bit count of less than 127 bits, see Section 13.6.8.3. With Start-Of-Frame (dominant bit) included in the code word, any rotation of the code word can be detected by the absence of the CRC Delimiter (recessive bit).

#### *13.6.2.8 Acknowledge Field (ACK)*

The Acknowledge Field consists of two bits, the Acknowledge Slot and the Acknowledge Delimiter, which are transmitted with a recessive level by the transmitter of the Data Frame. All CAN-controllers having received the matching CRC Sequence, report this by overwriting the transmitter's recessive bit in the Acknowledge Slot with a dominant bit. Thereby a transmitter, still monitoring the bus level recognizes that at least one receiver within the network has received a complete and correct message (i.e. no error was found). The Acknowledge Delimiter (recessive bit) is the second bit of the Acknowledge Field. As a result, the Acknowledge Slot is surrounded by two recessive bits: the CRC Delimiter and the Acknowledge Delimiter.

All nodes within a CAN network may use all the information coming to the network by all CAN-controllers (shared memory concept). Therefore, acknowledgement and error handling are defined to provide all information in a consistent way throughout this shared memory. Hence, there is no reason to discriminate different receivers of a message in the acknowledge field. If a node is disconnected from the network due to bus failure, this particular node is no longer part of the shared memory. To identify a 'lost node' additional and application specific precautions are required.

#### *13.6.2.9 End-Of-Frame*

Each Data Frame or Remote Frame is delimited by the End-Of-Frame bit sequence which consists of seven recessive bits (exceeds the bit stuff width by two bits). Using this method a receiver detects the end of a frame independent of a previous transmission error because the receiver expects all bits up to the end of the CRC Sequence to be coded by the method of bit-stuffing, see Section 13.6.7.3. The bit-stuffing logic is deactivated during the End-Of-Frame sequence.

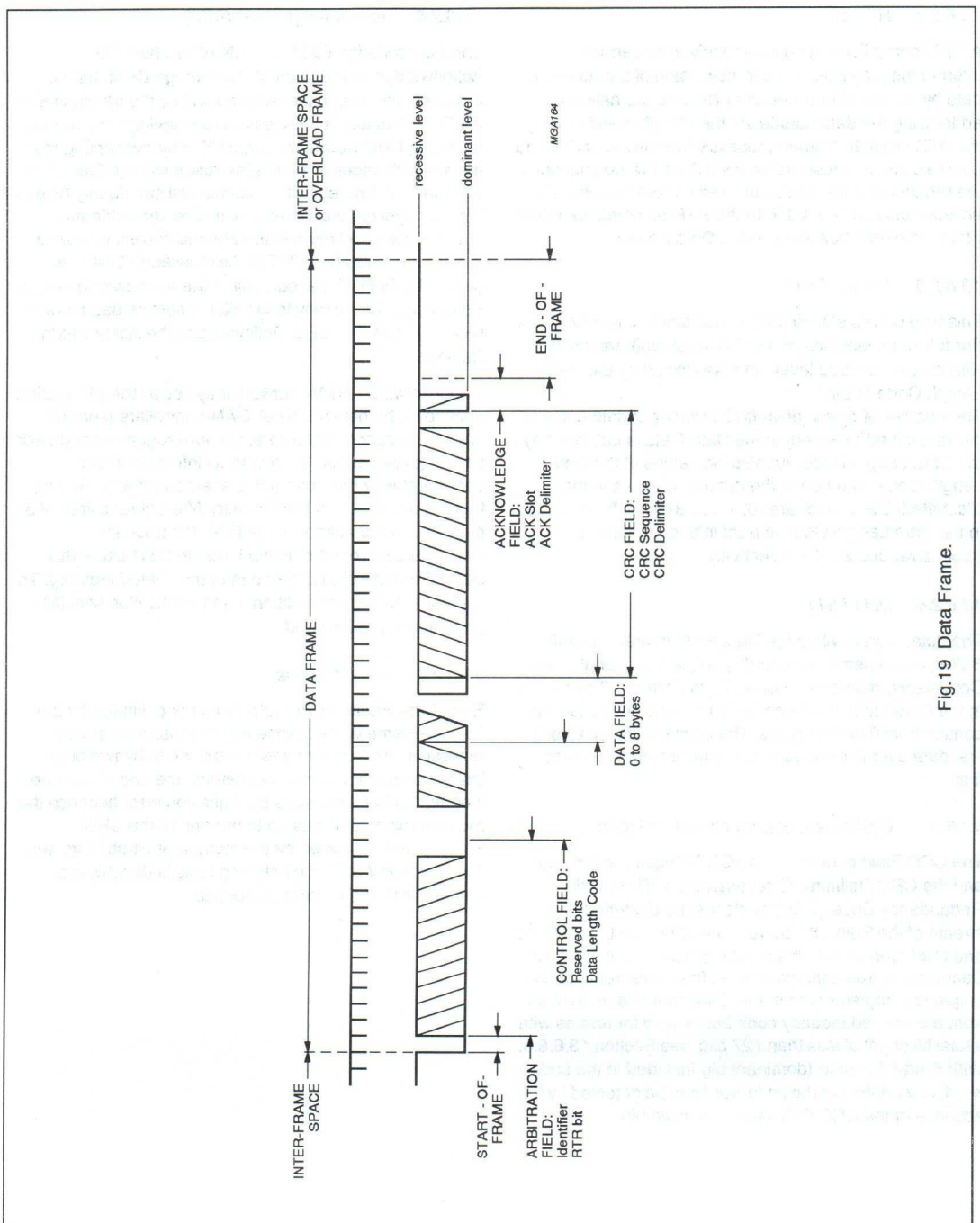



Fig.19 Data Frame.

### 13.6.3 REMOTE FRAME

A CAN-controller acting as a receiver for certain information may initiate the transmission of the respective data by transmitting a Remote Frame to the network, addressing the data source via the Identifier and setting the RTR bit HIGH (remote; recessive bus level). The Remote Frame is similar to the Data Frame with the following exceptions:

- RTR bit is set HIGH
- Data Length Code is ignored
- No Data Field contained.

Note that the value of the Data Length Code should be the one of the corresponding Data Frame, although it is ignored for a Remote Frame.

A Remote Frame is composed of six different bit fields:

- Start-of-Frame
- Arbitration Field
- Control Field
- CRC Field
- Acknowledge Field
- End-Of-Frame.

See Section 13.6.2 for more detailed explanation of the Remote Frame bit fields.

### 13.6.4 ERROR FRAME

The Error Frame consists of two different fields:

- The first field, accomplished by the superimposing of Error Flags contributed from different CAN-controllers
- The second field is the Error Delimiter.

#### *13.6.4.1 Error Flag*

There are two forms of an Error Flag:

- Active Error Flag, consists of six consecutive dominant bits.
- Passive Error Flag, consists of six consecutive recessive bits unless it is overwritten by dominant bits from other CAN-controllers.

An error-active CAN-controller (see Section 13.6.9) detecting an error condition signals this by transmission of an Active Error Flag. This Error Flag's form violates the bit-stuffing rule (see Section 13.6.7) applied to all fields, from Start-Of-Frame to CRC Delimiter, or destroys the fixed form of the fields Acknowledge Field or End-Of-Frame (see Fig.20).

Consequently, all other CAN-controllers detect an error condition and start transmission of an Error Flag. Therefore the sequence of dominant bits, which can be monitored on the bus, results from a superposition of different Error Flags transmitted by individual CAN-controllers. The total length of this sequence varies between six (minimum) and twelve (maximum) bits.

An error-passive CAN-controller (see Section 13.6.9) detecting an error condition tries to signal this by transmission of a Passive Error Flag. The error-passive CAN-controller waits for six consecutive bits with identical polarity, beginning at the start of the Passive Error Flag. The Passive Error Flag is complete when these six identical bits have been detected.

#### *13.6.4.2 Error Delimiter*

The Error Delimiter consists of eight recessive bits and has the same format as the Overload Delimiter. After transmission of an Error Flag, each CAN-controller monitors the bus-line until it detects a transition from a dominant-to-recessive bit level. At this point in time, every CAN-controller has finished sending its Error Flag and has additionally sent the first out of the 8 recessive bits of the Error Delimiter. Afterwards all CAN-controllers transmit the remaining recessive bits. After this event and an Intermission Field all error-active CAN-controllers within the network can start a transmission simultaneously.

If a detected error is signalled during transmission of a Data Frame or Remote Frame, the current message is spoiled and a retransmission of the message is initiated.

If a CAN-controller monitors any deviation of the Error Frame, a new Error Frame will be transmitted. Several consecutive Error Frames may result in the CAN-controller becoming error-passive and leaving the network unblocked.

In order to terminate an Error Flag correctly, an error-passive CAN-controller requires the bus to be Bus-Idle (see Section 13.6.6) for at least three bit periods (if there is a local error at an error-passive-receiver). Therefore a CAN-bus should not be 100% permanently loaded.

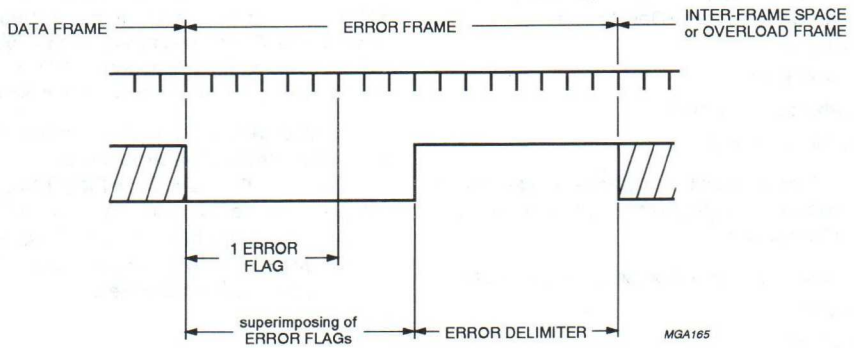



Fig.20 Error Frame.

### 13.6.5 Overload Frame

The Overload Frame consists of two fields:

- The Overload Flag
- The Overload Delimiter.

The transmission of an Overload Frame may only start:

- Condition 1; during the first bit period of an expected Intermission Field.
- Condition 2; one bit period after detecting the dominant bit during Intermission Field.

The P8xC592's on-chip CAN-controller will never initiate transmission of a condition 1 Overload Frame and will only react on a transmitted condition 2 Overload Frame, according to the CAN-protocol. No more than two Overload Frames are generated to delay a Data Frame or a Remote Frame. Although the overall form of the Overload Frame corresponds to that of the Error Frame, an Overload Frame does not initiate or require the retransmission of the preceding frame.

#### *13.6.5.1 Overload Flag*

The Overload Flag consists of six dominant bits and has a similar format to the Error Flag.

There are two conditions in the CAN-protocol which lead to the transmission of an Overload Flag:

- Condition 1; receiver circuitry requires more time to process the current data before receiving the next frame (receiver not ready).
- Condition 2; detection of a dominant bit during Intermission Field (see Section 13.6.6).

The Overload Flag's form corrupts the fixed form of the Intermission Field. All other CAN-controllers detecting the overload condition also transmit an Overload Flag (condition 2).

#### *13.6.5.2 Overload Delimiter*

The Overload Delimiter consists of eight recessive bits and takes the same form as the Error Delimiter. After transmission of an Overload Flag, each CAN-controller monitors the bus-line until it detects a transition from a dominant-to-recessive bit level. At this point in time, every CAN-controller has finished sending its Overload Flag and all CAN-controllers start simultaneously transmitting seven more recessive bits.

### 13.6.6 INTER-FRAME SPACE

Data Frames and Remote Frames are separated from preceding frames (all types) by an Inter-Frame Space, consisting of an Intermission Field and a Bus-Idle. Error-passive CAN-controllers also send a Suspend Transmission (see Section 13.6.9) after transmission of a message. Overload Frames and Error Frames are not preceded by an Inter-Frame Space.

#### *13.6.6.1 Intermission Field*

The Intermission Field consists of three recessive bits. During an Intermission period, no frame transmissions will be started by the P8xC592's on-chip CAN-controller. An Intermission is required to have a fixed time period to allow a CAN-controller to execute internal processes prior to the next receive or transmit task.

#### *13.6.6.2 Bus-Idle*

The Bus-Idle time may be of arbitrary length (min. 0 bit). The bus is recognized to be free and a CAN-controller having information to transmit may access the bus. The detection of a dominant bit level during Bus-Idle on the bus is interpreted as the Start-Of-Frame.

### 13.6.7 BUS ORGANIZATION

Bus organization is based on five basic rules described in the following subsections.

#### *13.6.7.1 Bus Access*

CAN-controllers only start transmission during the Bus-Idle state. All CAN-controllers synchronize on the leading edge of the Start-Of-Frame (hard synchronization).

#### *13.6.7.2 Bus Arbitration*

If two or more CAN-controllers simultaneously start transmitting, the bus access conflict is solved by a bit-wise arbitration process during transmission of the Arbitration Field.

During arbitration every transmitting CAN-controller compares its transmitted bit level with the monitored bus level. Any CAN-controller which transmits a recessive bit and monitors a dominant bus level immediately becomes the receiver of the higher-priority message on the bus without corrupting any information on the bus. Each message contains an unique Identifier and a RTR bit describing the type of data within the message. The Identifier together with the RTR bit implicitly define the message's bus access priority. During arbitration the most significant bit of the Identifier is transmitted first and the RTR bit last. The message with the lowest binary value of the Identifier and RTR bit has the highest priority. A Data Frame has higher priority than a Remote Frame due to its RTR bit having a dominant level.

For every Data Frame there is an unique transmitter. For reasons of compatibility with other CAN-bus controllers, use of the Identifier bit pattern ID = 1111111XXXXB (X being bits of arbitrary level) is forbidden.

The number of available different Identifiers:

$(2^{11} - 2^4) = 2032.$

#### *13.6.7.3 Coding/Decoding*

The following bit fields are coded using the bit-stuffing technique:

- Start-Of-Frame
- Arbitration Field
- Control Field
- Data Field
- CRC Sequence.

When a transmitting CAN-controller detects five consecutive bits of identical polarity to be transmitted, a complementary (stuff) bit is inserted into the transmitted bit-stream.

When a receiving CAN-controller has monitored five consecutive bits with identical polarity in the received bit streams of the above described bit fields, it automatically deletes the next received (stuff) bit. The level of the deleted stuff bit has to be the complement of the previous bits; otherwise a Stuff Error will be detected and signalled (see Section 13.6.8).

The remaining bit fields or frames are of fixed form and are not coded or decoded by the method of bit-stuffing.

The bit-stream in a message is coded according to the Non-Return-to-Zero (NRZ) method, i.e. during a bit period, the bit level is held constant, either recessive or dominant.

#### *13.6.7.4 Error Signalling*

A CAN-controller which detects an error condition, transmits an Error Flag. Whenever a Bit Error, Stuff Error, Form Error or an Acknowledgement Error is detected, transmission of an Error Flag is started at the next bit. Whenever a CRC Error is detected, transmission of an Error Flag starts at the bit following the Acknowledge Delimiter, unless an Error Flag for another error condition has already started. An Error Flag violates the bit-stuffing law or corrupts the fixed form bit fields. A violation of the bit-stuffing law affects any CAN-controller which detects the error condition. These devices will also transmit an Error Flag.

An error-passive CAN-controller (see Section 13.6.9) which detects an error condition, transmits a Passive Error Flag. A Passive Error Flag is not able to interrupt a current message at different CAN-controllers but this type of Error Flag may be ignored (overwritten) by other CAN-controllers. After having detected an error condition, an error-passive CAN-controller will wait for six consecutive bits with identical polarity and when monitoring them, interpret them as an Error Flag.

After transmission of an Error Flag, each CAN-controller monitors the bus-line until it detects a transition from a dominant-to-recessive bit level. At this point in time, every CAN-controller has finished transmitting its Error Flag and all CAN-controllers start transmitting seven additional recessive bits (Error Delimiter, see Section 13.6.4).

The message format of a Data Frame or Remote Frame is defined in such a way that all detectable errors can be signalled within the message transmission time and therefore it is very simple for the CAN-controllers to associate an Error Frame to the corresponding message and to initiate retransmission of the corrupted message. If a CAN-controller monitors any deviation of the fixed form of an Error Frame, it transmits a new Error Frame.

#### *13.6.7.5 Overload Signalling*

Some CAN-controllers (but not the one on-chip of the P8xC592) require to delay the transmission of the next Data Frame or Remote Frame by transmitting one or more Overload Frames. The transmission of an Overload Frame must start during the first bit of an expected Intermission Field. Transmission of Overload Frames which are reactions on a dominant bit during an expected Intermission Field, start one bit after this event.

Though the format of Overload Frame and Error Frame are identical, they are treated differently. Transmission of an Overload Frame during Intermission Field does not initiate the retransmission of any previous Data Frame or Remote Frame. If a CAN-controller which transmitted an Overload Frame monitors any deviation of its fixed form, it transmits an Error Frame.

### 13.6.8 Error Detection

The processes described in Sections 13.6.8.1 to 13.6.10.3 are implemented in the P8xC592's on-chip CAN-controller for error detection.

#### *13.6.8.1 Bit Error*

A transmitting CAN-controller monitors the bus on a bit-by-bit basis. If the bit level monitored is different from the transmitted one, a Bit Error is signalled.
The exceptions being:

- During the Arbitration Field, a recessive bit can be overwritten by a dominant bit. In this case, the CAN-controller interprets this as a loss of arbitration.
- During the Acknowledge Slot, only the receiving CAN-controllers are able to recognize a Bit Error.

#### *13.6.8.2 Stuff Error*

The following bit fields are coded using the bit-stuffing technique:

- Start-Of-Frame
- Arbitration Field
- Control Field
- Data Field
- CRC Sequence.

There are two possible ways of generating a Stuff Error:

- A disturbance generates more than the allowed five consecutive bits with identical polarity. These errors are detected by all CAN-controllers.
- A disturbance falsifies one or more of the five bits preceding the stuff bit. This error situation is not recognized as a Stuff Error by the receivers. Therefore, other error detection processes may detect this error condition such as:
  - CRC check, format violation at the receiving CAN-controllers, or
  - Bit Error detection by the transmitting CAN-controller.

#### *13.6.8.3 CRC Error*

To ensure the validity of a transmitted message all receivers perform a CRC check. Therefore, in addition to the (destuffed) information digits (Start-Of-Frame up to Data Field), every message includes some control digits (CRC Sequence; generated by the transmitting CAN-controller of the respective message) used for error detection.

The code used by all CAN-controllers is a (shortened) BCH code, extended by a parity check and has the following attributes:

- 127 bits as maximum length of the code.
- 112 bits as maximum number of information digits (max. 83 bits are used by the CAN-controller).
- Length of the CRC Sequence amounts to 15 bits.
- Hamming distance d = 6.

As a result, '(d–1)' random errors are detectable (some exceptions exist).

The CRC Sequence is determined (calculated) by the following procedure:

1. The destuffed bit stream consisting of Start-Of-Frame up to the Data Field (if present) is interpreted as polynomial with coefficients 0 or 1.
2. This polynomial is divided (modulo-2) by the following generator polynomial, which includes a parity check:

   $f(x) = (x^{14} + x^9 + x^8 + x^6 + x^5 + x^4 + x^2 + x + 1)$

   $(x + 1) = 1100010110011001$ B.
3. The remainder of this polynomial division is the CRC Sequence.

Burst errors are detected up to a length of 15 [degree of f(x)]. Multiple errors (number of disturbed bits at least d = 6) are not detected with a residual error probability of $2^{-15}$ ($3 \times 10^{-5}$) by CRC check only.

#### *13.6.8.4 Form Error*

Form Errors result from violations of the fixed form of the following bit fields:

- CRC Delimiter
- Acknowledge Delimiter
- End-Of-Frame
- Error Delimiter
- Overload Delimiter.

During the transmission of these bit fields an error condition is recognized if a dominant bit level instead of a recessive one is detected.

#### *13.6.8.5 Acknowledgement Error*

This is detected by a transmitter whenever it does not monitor a dominant bit during the Acknowledge Slot.

#### *13.6.8.6 Error detection by an Error Flag from another CAN-controller*

The detection of an error is signalled by transmitting an Error Flag. An Active Error Flag causes a Stuff Error, a Bit Error or a Form Error at all other CAN-controllers.

#### *13.6.8.7 Error Detection Capabilities*

Errors which occur at all CAN-controllers (global errors) are 100% detected. For local errors, i.e. for errors occurring at some CAN-controllers only, the shortened BCH code, extended by a parity check, has the following error detection capabilities:

- Up to five single Bit Errors are 100% detected, even if they are distributed randomly within the code.
- All single Bit Errors are detected if their total number (within the code) is odd.
- The residual error probability of the CRC check amounts to ($3 \times 10^{-5}$). As an error may be detected not only by CRC check but also by other detection processes described above the residual error probability is several magnitudes less than ($3 \times 10^{-5}$).

### 13.6.9 Error Confinement Definitions

#### *13.6.9.1 Bus-OFF*

A CAN-controller which has too many unsuccessful transmissions, relative to the number of successful transmissions, will enter the Bus-OFF state. It remains in this state, neither receiving nor transmitting messages until the Reset Request bit is set LOW (absent) and both Error Counters set to 0 (see Section 13.6.10).

#### *13.6.9.2 Acknowledge*

A CAN-controller which has received a valid message correctly, indicates this to the transmitter by transmitting a dominant bit level on the bus during the Acknowledge Slot, independent of accepting or rejecting the message.

#### *13.6.9.3 Error-Active*

An error-active CAN-controller in its normal operating state is able to receive and to transmit normally and also to transmit an Active Error Flag (see Section 13.6.10).

#### *13.6.9.4 Error-Passive*

An error-passive CAN-controller may transmit or receive messages normally. In the case of a detected error condition it transmits a Passive Error Flag instead of an Active Error Flag. Hence the influence on bus activities by an error-active CAN-controller (e.g. due to a malfunction) is reduced.

#### *13.6.9.5 Suspend Transmission*

After an error-passive CAN-controller has transmitted a message, it sends eight recessive bits after the Intermission Field and then checks for Bus-Idle. If during Suspend Transmission another CAN-controller starts transmitting a message the suspended CAN-controller will become the receiver of this message; otherwise being in Bus-Idle it may start to transmit a further message.

#### *13.6.9.6 Start-Up*

A CAN-controller which either was switched off or in the Bus-OFF state, must run a Start-Up routine in order to:

- Synchronize with other available CAN-controllers before starting to transmit. Synchronization is achieved, when 11 recessive bits, equivalent to Acknowledge Delimiter, End-Of-Frame and Intermission Field, have been detected (Bus-Free).
- Wait for other CAN-controllers without passing into the Bus-OFF state (due to a missing acknowledge), if there is no other CAN-controller currently available.

### 13.6.10 AIMS OF ERROR CONFINEMENT

#### *13.6.10.1 Distinction of short and long disturbances*

The CPU must be informed when there are long disturbances and when bus activities have returned to normal operation. During long disturbances, a CAN-controller enters the Bus-OFF state and the CPU may use default values.

Minor disturbances of bus activities will not effect a CAN-controller. In particular, a CAN-controller does not enter the Bus-OFF state or inform the CPU of a short bus disturbance.

#### *13.6.10.2 Detection and localization of hardware disturbances and defects*

The rules for error confinement are defined by the CAN-protocol specification (and implemented in the P8xC592's on-chip CAN-controller), in such a way that the CAN-controller, being nearest to the error-locus, reacts with a high probability the quickest (i.e. becomes error-passive or Bus-OFF). Hence errors can be localized and their influence on normal bus activities is minimized.

#### *13.6.10.3 Error Confinement*

All CAN-controllers contain a Transmit Error Counter and a Receive Error Counter, which registers errors during the transmission and the reception of messages, respectively.

If a message is transmitted or received correctly, the count is decreased. In the event of an error, the count is increased. The Error Counters have an non-proportional method of counting: an error causes a larger counter increase than a correctly transmitted/received message causes the count to decrease. Over a period of time this may result in an increase in error counts, even if there are fewer corrupted messages than uncorrupted ones. The level of the Error Counters reflect the relative frequency of disturbances. The ratio of increase/decrease depends on the acceptable ratio of invalid/valid messages on the bus and is hardware implemented to eight.

If one of the Error Counters exceeds the Warning Limit of 96 error points, indicating a significant accumulation of error conditions, this is signalled by the CAN-controller (Error Status, Error Interrupt).

A CAN-controller operates in the error-active mode until it exceeds 127 error points on one of its Error Counters. At this value it will enter the error-passive state. A transmit error which exceeds 255 error points results in the CAN-controller entering the Bus-OFF state.

## 14 INTERRUPT SYSTEM

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. Interrupt response latency is from 2.25 μs to 7.5 μs when using a 16 MHz crystal. The latency time strongly depends on the sequence of instructions executed directly after an interrupt request. During a CAN-DMA transfer the interrupt system is disabled (see Section 13.5.17). The P8xC592 acknowledges interrupt requests from fifteen sources as follows:

- $\overline{\text{INT0}}$ and $\overline{\text{INT1}}$: externally via pins 27 and 28 respectively
- Timer 0 and Timer 1: from the two internal counters
  - If the capture function remains unused and the Capture Register contents are 'don't care' then the corresponding input pins 'CTnI', with 'n = 0 ... 3', may be used as positive and/or negative edge triggered external interrupts INT2 to INT5. But note that they can not terminate the Idle mode because the Timer 2 is switched off then
- Timer T2, 8 separate interrupts:
  - 4 capture interrupts
  - 3 compare interrupts
  - an overflow interrupt
- ADC end-of-conversion interrupt
- CAN-controller interrupt
- UART serial I/O port interrupt.

Each interrupt vectors to a separate location in Program Memory for its service program. Each source can be individually enabled or disabled by a corresponding bit in the IEN0 or IEN1 register, moreover each interrupt may be programmed to a HIGH or LOW priority level using a corresponding bit in the IP0 or IP1 register. Also all enabled sources can be globally disabled or enabled. Both external interrupts can be programmed to be level-activated or transition-activated, and an active LOW level allows 'wire-ORing' of several interrupt sources to the input pin.

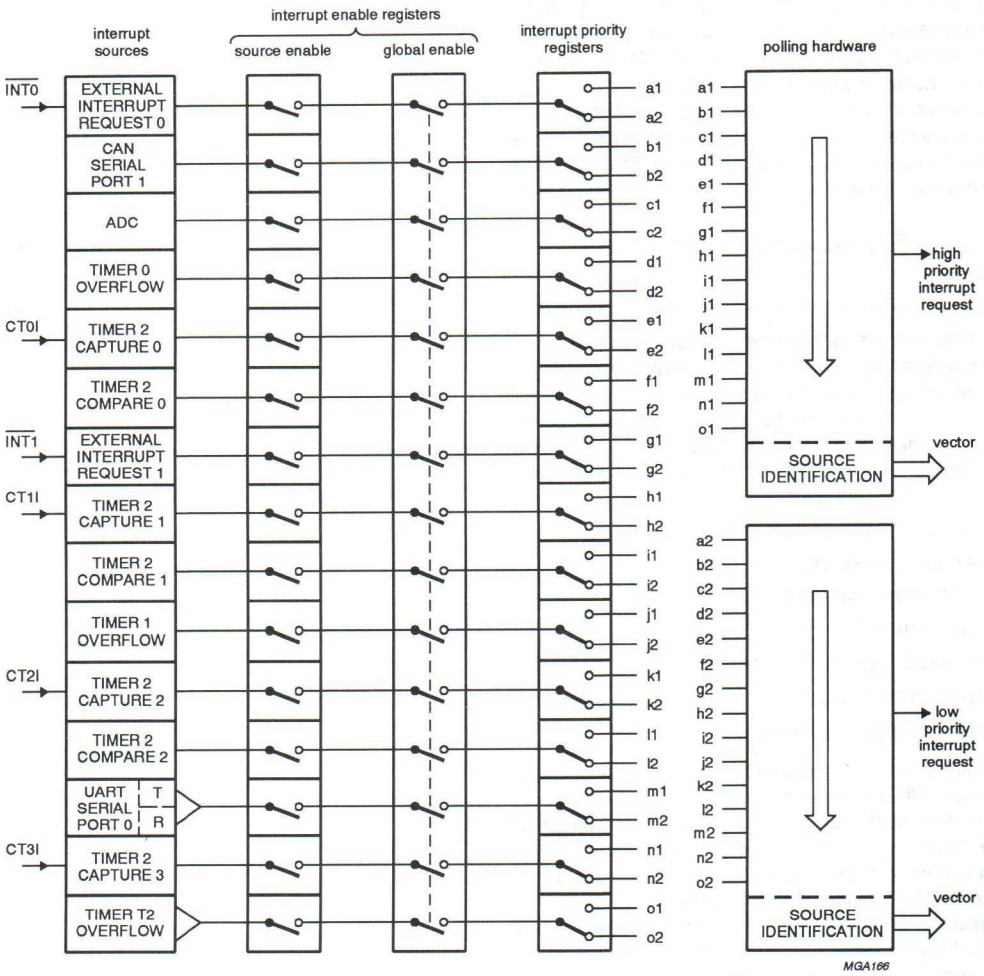



Fig.21 Interrupt system.

### 14.1 Interrupt Enable and Priority Registers

#### 14.1.1 INTERRUPT ENABLE REGISTER 0 (IEN0)

**Table 71** Interrupt Enable register 0 (address A8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| EA | EAD | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 |

**Table 72** Description of the IEN0 bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | EA | **General enable/disable control.** If bit EA is:<br>LOW, then no interrupt is enabled.<br>HIGH, then any individually enabled interrupt will be accepted. |
| 6 | EAD | Enable ADC interrupt. |
| 5 | ES1 | Enable SIO1 (CAN) interrupt. |
| 4 | ES0 | Enable SIO0 (UART) interrupt. |
| 3 | ET1 | Enable Timer 1 interrupt. |
| 2 | EX1 | Enable External 1 interrupt. |
| 1 | ET0 | Enable Timer 0 interrupt. |
| 0 | EX0 | Enable External 0 interrupt. |

#### 14.1.2 INTERRUPT ENABLE REGISTER 1 (IEN1)

**Table 73** Interrupt Enable register 0 (address E8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| ET2 | ECM2 | ECM1 | ECM0 | ECT3 | ECT2 | ECT1 | ECT0 |

**Table 74** Description of the IEN1 bits

Logic 0 = interrupt disabled; Logic 1 = interrupt enabled.

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | ET2 | Enable T2 overflow interrupt(s). |
| 6 | ECM2 | Enable T2 comparator 2 interrupt. |
| 5 | ECM1 | Enable T2 comparator 1 interrupt. |
| 4 | ECM0 | Enable T2 comparator 0 interrupt. |
| 3 | ECT3 | Enable T2 capture register 3 interrupt. |
| 2 | ECT1 | Enable T2 capture register 2 interrupt. |
| 1 | ECT1 | Enable T2 capture register 1 interrupt. |
| 0 | ECT0 | Enable T2 capture register 0 interrupt. |

### 14.1.3 INTERRUPT PRIORITY REGISTER 0 (IP0)

**Table 75** Interrupt Priority register 0 (address B8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | PAD | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 |

**Table 76** Description of the IP0 bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | – | Not used. |
| 6 | PAD | ADC interrupt priority level. |
| 5 | PS1 | SIO1 (CAN) interrupt priority level. |
| 4 | PS0 | SIO0 (UART) interrupt priority level. |
| 3 | PT1 | Timer 1 interrupt priority level. |
| 2 | PX1 | External interrupt 1 priority level. |
| 1 | PT0 | Timer 0 interrupt priority level. |
| 0 | PX0 | External interrupt 0 priority level. |

### 14.1.4 INTERRUPT PRIORITY REGISTER 1 (IP1)

**Table 77** Interrupt Priority register 1 (address F8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| PT2 | PCM2 | PCM1 | PCM0 | PCT3 | PCT2 | PCT1 | PCT0 |

**Table 78** Description of the IP1 bits

Logic 0 = low priority; Logic 1 = high priority.

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | PT2 | T2 overflow interrupt(s) priority level. |
| 6 | PCM2 | T2 comparator 2 priority interrupt level. |
| 5 | PCM1 | T2 comparator 1 priority interrupt level. |
| 4 | PCM0 | T2 comparator 0 priority interrupt level. |
| 3 | PCT3 | T2 capture register 3 priority interrupt level. |
| 2 | PCT2 | T2 capture register 2 priority interrupt level. |
| 1 | PCT1 | T2 capture register 1 priority interrupt level. |
| 0 | PCT0 | T2 capture register 0 priority interrupt level. |

### 14.2 Interrupt Vectors

The vector indicates the Program Memory location where the appropriate interrupt service routine starts (see Table 79).

**Table 79** Interrupt vectors

| SOURCE | BIT | VECTOR |
|---|---|---|
| External 0 | X0 | 0003H |
| Timer 0 overflow | T0 | 000BH |
| External 1 | X1 | 0013H |
| Timer 1 overflow | T1 | 001BH |
| Serial I/O 0 (UART) | S0 | 0023H |
| Serial I/O 1 (CAN) | S1 | 002BH |
| T2 capture 0 | CT0 | 0033H |
| T2 capture 1 | CT1 | 003BH |
| T2 capture 2 | CT2 | 0043H |
| T2 capture 3 | CT3 | 004BH |
| ADC completion | ADC | 0053H |
| T2 compare 0 | CM0 | 005BH |
| T2 compare 1 | CM1 | 0063H |
| T2 compare 2 | CM2 | 006BH |
| T2 overflow | T2 | 0073H |

### 14.3 Interrupt Priority

Each interrupt source can be either high priority or low priority. If both priorities are requested simultaneously, the processor will branch to the high priority vector. If there are simultaneous requests from sources of the same priority, then interrupts will be serviced in the following order:

X0, S1, ADC, T0, CT0, CM0, X1, CT1, CM1, T1, CT2, CM2, S0, CT3, T2.

A low priority interrupt routine can only be interrupted by a high priority interrupt. A high priority interrupt routine can not be interrupted.

## 15 POWER REDUCTION MODES

The P8xC592 has three software-selectable modes to reduce power consumption. These are:

- Sleep mode, affecting the CAN-controller only
- Idle mode, affecting the
  - CPU (halted)
  - Timer 2 (stopped and reset)
  - PWM0, PWM1 (reset, output = HIGH)
  - ADC (aborted if in progress)
- Power-down mode, affecting the whole P8xC592 device.

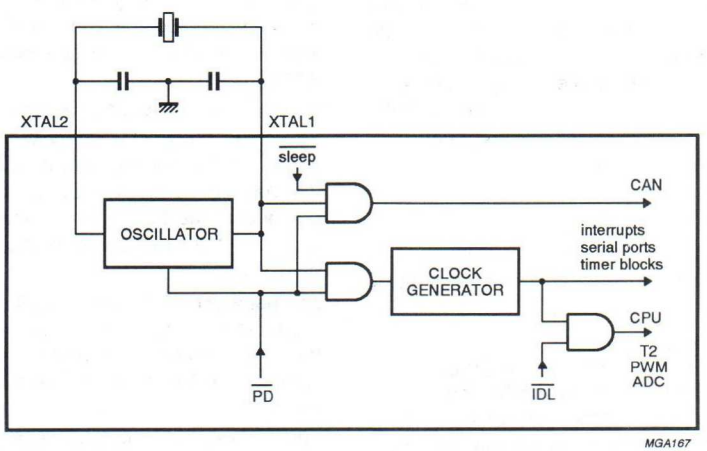



Fig.22 Internal Sleep, Idle and Power-down clock configuration.

### 15.1 Power Control Register (PCON)

**Table 80** Power Control Register (address 87H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SMOD | – | – | WLE | GF1 | GF0 | PD | IDL |

**Table 81** Description of the PCON bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| 7 | SMOD | **Double baud rate bit**. When set to logic 1 the baud rate is doubled when the serial port SIO0 is being used in Modes 1, 2 and 3. |
| 6 | – | Reserved. |
| 5 | – | |
| 4 | WLE | **Watchdog Load Enable**. This flag must be set by software prior to loading T3 (Watchdog timer). It is cleared when T3 is loaded. |
| 3 | GF1 | **General purpose flag bits.** |
| 2 | GF0 | |
| 1 | PD | **Power-down bit**. Setting this bit activates Power-down mode (note 1). It can only be set if input $\overline{EW}$ is HIGH. |
| 0 | IDL | **Idle mode bit**. Setting this bit activates the Idle mode (note 1). |

**Note**

1. If PD and IDL are set to HIGH at the same time, PD takes precedence. The reset value of PCON is 0XX00000B.

### 15.2 CAN Sleep Mode

In order to reduce power consumption of the P8xC592 the CAN-controller may be switched off (disconnecting the internal clock) by setting the CAN Command Register bit 4 (Sleep) HIGH. The CAN-controller leaves this Sleep mode by detecting either activity on the CAN-bus (dominant bit-level on CRX0/CRX1; see Chapter 5, Table 1) or by setting the Sleep bit to LOW. As the CPU can not only write to the Sleep bit, but can also read it, the CAN-controller status can be determined directly.

### 15.3 Idle Mode

The instruction that sets bit PCON.0 to HIGH is the last one executed in the normal operating mode before Idle mode is activated.

Once in the Idle mode, the CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator, RAM and all other registers maintain their data during Idle mode. The status of the external pins during Idle mode is shown in see Table 82.

There are three ways to terminate the Idle mode:

- Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware, provided that the interrupt source is active during Idle mode. After the interrupt is serviced, the program continues with the instruction immediately after the one, at which the interrupt request was detected.
- The flag bits GF0 and GF1 may be used to determine whether the interrupt was received during normal execution or during the Idle mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When Idle mode is terminated by an interrupt, the service routine can examine the status of the flag bits.
- Another way of terminating the Idle mode is an external hardware reset. Since the oscillator is still running, the reset signal is required to be active only for two machine cycles (24 oscillator periods) to complete the reset operation.
- The third way is the internally generated watchdog reset after an overflow of Timer 3.

### 15.4 Power-down Mode

The instruction that sets bit PCON.1 to HIGH, is the last one executed before entering the Power-down mode. In Power-down mode the oscillator of the P8xC592 is stopped. If the CAN-controller is in use, it is recommended to set it into Sleep mode before entering Power-down mode. However, setting PCON.1 to HIGH also sets the Sleep bit (CAN-controller Command Register bit 4) to HIGH.

The P8xC592 leaves Power-down mode either by a hardware reset or by a CAN Wake-Up interrupt (due to activity on the CAN-bus), if the SIO1 (CAN) interrupt source is enabled (contents of register IEN0 = 1X1XXXXXB).

A hardware reset affects the whole P8xC592, but leaves the contents of the on-chip RAM unchanged (CAN-controller-and CPU's SFRs are reset, see Section 13.5.2, Chapter 17 and Table 40). A CAN Wake-Up interrupt during Power-down mode causes a reset output pulse with a width of 6144 machine cycles (4.6 ms with $f_{CLK}$ = 16 MHz). All hardware except that for the CAN-controller of the P8xC592 is reset (i.e. the contents of all CAN-controller registers are preserved).

A capacitance connected to the RST pin can be used to lengthen the internally generated reset pulse. If the pulse exceeds 8192 machine cycles, the CAN-controller part is reset too.

**Table 82** Status of external pins during Idle and Power-down modes

| MODE | PROGRAM | ALE | $\overline{\text{PSEN}}$ | PORT0 | PORT1[1] | PORT2 | PORT3 | PORT4 | PWM0/ PWM1 |
|---|---|---|---|---|---|---|---|---|---|
| Idle | internal | 1 | 1 | port data | port data | port data | port data | port data | 1 |
| | external | 1 | 1 | floating | port data | address | port data | port data | 1 |
| Power-down | internal | 0 | 0 | port data | port data | port data | port data | port data | 1 |
| | external | 0 | 0 | floating | port data | port data | port data | port data | 1 |

**Note**

1. If the port pins P1.6 and P1.7 are used as the CAN transmitter outputs (CTX0 and CTX1), then during Sleep and Power-down mode these pins output a 'recessive' level (see Sections 13.5.2 and 13.5.11).

## 16 OSCILLATOR CIRCUITRY

The oscillator circuitry of the P8xC592 is a single-stage inverting amplifier in a Pierce oscillator configuration. The circuitry between XTAL1 and XTAL2 is basically an inverter biased to the transfer point. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuitry. Both are operated in parallel resonance. XTAL1 (pin 34) is the high gain amplifier input, and XTAL2 (pin 33) is the output (see Fig.23). If XTAL1 is driven from an external source, XTAL2 must be left open (see Fig.24).

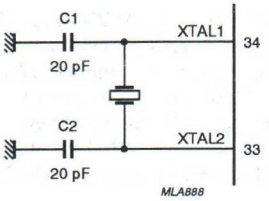



Fig.23 P8xC592 oscillator circuit.

## 17 RESET CIRCUITRY

The reset pin RST is connected to a Schmitt trigger for noise rejection (see Fig.25). A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods). The CPU responds by executing an internal reset. During reset ALE and $\overline{PSEN}$ output a HIGH level. In order to perform a correct reset, this level must not be affected by external elements.

Also with the P8xC592, the RST line can be pulled HIGH internally by a pull-up transistor activated by the Watchdog timer T3. The length of the output pulse from T3 is 3 machine cycles. A pulse of such short duration is necessary in order to recover from a processor or system fault as fast as possible.

During Power-down a reset could be generated internally via the CAN Wake-Up interrupt. Then the RST pin is pulled HIGH for 6144 machine cycles. In this case the CAN-controller is not reset.

If the Watchdog timer or the CAN Wake-Up interrupt is used to reset external devices, the usual capacitor arrangement for Power-on-reset (see Fig.26) should not be used.

However, the internal reset is forced, independent of the external level on the RST pin.

The MAIN RAM and AUXILIARY RAM are not affected. When $V_{DD}$ is turned on, the RAM content is indeterminate. A reset leaves the internal registers as shown in Table 83.

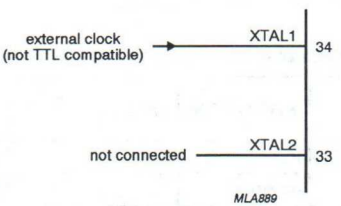



Fig.24 Driving P8xC592 from an external source.

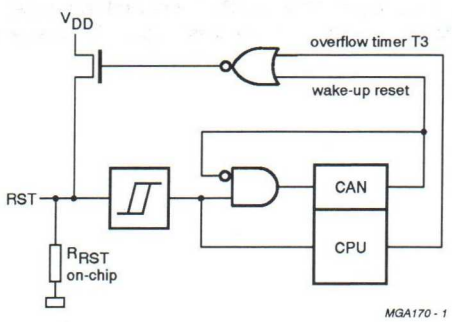



Fig.25 On-chip reset configuration.

**Table 83** Internal registers' contents after a reset

X = undefined state.

| REGISTER | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **CPU part** | | | | | | | | |
| ACC | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ADC0 | X | X | 0 | 0 | 0 | 0 | 0 | 0 |
| ADCH | X | X | X | X | X | X | X | X |
| B | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CML0 to CML2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CMH0 to CMH2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CTCON | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CTL0 to CTL3 | X | X | X | X | X | X | X | X |
| CTH0 to CTH3 | X | X | X | X | X | X | X | X |
| DPL | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DPH | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| IEN0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| IEN1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| IP0 | X | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| IP1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCH | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCL | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCON | 0 | X | X | 0 | 0 | 0 | 0 | 0 |
| PSW | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PWM0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCWM1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCWMP | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| P0 to P4 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| P5 | X | X | X | X | X | X | X | X |
| RTE | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| S0BUF | X | X | X | X | X | X | X | X |
| S0CON | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| REGISTER | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| CANSTA | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| CANCON | X | X | X | 0 | 0 | 0 | 0 | 0 |
| CANDAT | X | X | X | X | X | X | X | X |
| CANADR | 0 | X | 1 | 0 | 0 | 1 | 0 | 0 |
| SP | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| STE | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| TCON | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TH0, TH1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TMH2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TL0, TL1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TML2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TMOD | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TM2CON | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TM2IR | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| T3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **CAN part** | | | | | | | | |
| CR | 0 | X | 1 | X | X | X | X | 1 |
| CMR | 1 | 1 | X | 0 | X | X | X | X |
| SR | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| IR | X | X | X | 0 | 0 | 0 | 0 | 0 |
| ACR | X | X | X | X | X | X | X | X |
| AMR | X | X | X | X | X | X | X | X |
| BTR0 | X | X | X | X | X | X | X | X |
| BTR1 | X | X | X | X | X | X | X | X |
| OCR | X | X | X | X | X | X | X | X |
| TR | X | X | X | X | X | X | X | X |
| TXB 10 to 19 | X | X | X | X | X | X | X | X |
| RXB 20 to 29 | X | X | X | X | X | X | X | X |

### 17.1 Power-on Reset

If the RST pin is connected to $V_{DD}$ via a 2.2 µF capacitor, as shown in Fig.26, an automatic reset can be obtained by switching on $V_{DD}$ (provided its rise time is <10 ms). The decrease of the RST pin voltage depends on the capacitor and the internal resistor $R_{RST}$. That voltage must remain above the lower threshold for at minimum the oscillator start-up time plus 2 machine cycles.

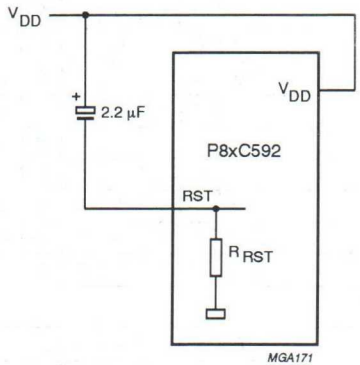



Fig.26 Power-on-reset.

## 18 INSTRUCTION SET

The P8xC592 uses the powerful instruction set of the P80C51. It consists of 49 single-byte, 45 two-byte and 17 three-byte instructions. Using a 16 MHz quartz, 64 of the instructions are executed in 0.75 µs, 45 in 1.5 µs and the multiply, divide instructions in 3 µs. A summary of the instruction set is given in Tables 84, 85, 86, 87 and 88.

### 18.1 Addressing Modes

Most instructions have a 'destination/source' field that specifies the data type, addressing modes and operands involved. For all these instructions, except from MOVs, the destination operand is also a source operand (e.g. ADD A, R7).

Five types of addressing modes are used:

- Register Addressing,
  - R0 to R7 (4 banks)
  - A,B,C (bit), AB (2 bytes), DPTR (double byte).
- Direct Addressing,
  - lower 128 bytes of internal MAIN RAM (including the 4 R0 to R7 register banks)
  - Special Function Registers (SFRs)
  - 128 bits in a subset of the internal MAIN RAM (see Fig.5)
  - 128 bits in a subset of the Special Function Registers (see Figs 6 and 7).
- Register-Indirect Addressing,
  - internal RAM (@R0, @R1, @SP [PUSH/POP])
  - internal AUXILIARY RAM (@R0, @R1, @DPTR)
  - external Data Memory (@DPTR).
- Immediate Addressing,
  - Program Memory (in-code 8 bit or 16 bit constant).
- Base-Register-plus Index-Register-Indirect Addressing,
  - Program Memory look-up table (@DPTR+A, @PC+A).

The first three addressing modes are usable for destination operands.

### 18.2 Instruction Set

For the description of the **Data Addressing Modes** and **Hexadecimal opcode cross-reference** see Table 88.

**Table 84** Instruction set description: Arithmetic operations

| MNEMONIC | | DESCRIPTION | BYTES | CYCLES | OPCODE (HEX) |
|---|---|---|---|---|---|
| **Arithmetic operations** | | | | | |
| ADD | A,Rr | Add register to A | 1 | 1 | 2* |
| ADD | A,direct | Add direct byte to A | 2 | 1 | 25 |
| ADD | A,@Ri | Add indirect RAM to A | 1 | 1 | 26, 27 |
| ADD | A,#data | Add immediate data to A | 2 | 1 | 24 |
| ADDC | A,Rr | Add register to A with carry flag | 1 | 1 | 3* |
| ADDC | A,direct | Add direct byte to A with carry flag | 2 | 1 | 35 |
| ADDC | A,@Ri | Add indirect RAM to A with carry flag | 1 | 1 | 36, 37 |
| ADDC | A,#data | Add immediate data to A with carry flag | 2 | 1 | 34 |
| SUBB | A,Rr | Subtract register from A with borrow | 1 | 1 | 9* |
| SUBB | A,direct | Subtract direct byte from A with borrow | 2 | 1 | 95 |
| SUBB | A,@Ri | Subtract indirect RAM from A with borrow | 1 | 1 | 96, 97 |
| SUBB | A,#data | Subtract immediate data from A with borrow | 2 | 1 | 94 |
| INC | A | Increment A | 1 | 1 | 04 |
| INC | Rr | Increment register | 1 | 1 | 0* |
| INC | direct | Increment direct byte | 2 | 1 | 05 |
| INC | @Ri | Increment indirect RAM | 1 | 1 | 06, 07 |
| DEC | A | Decrement A | 1 | 1 | 14 |
| DEC | Rr | Decrement register | 1 | 1 | 1* |
| DEC | direct | Decrement direct byte | 2 | 1 | 15 |
| DEC | @Ri | Decrement indirect RAM | 1 | 1 | 16, 17 |
| INC | DPTR | Increment data pointer | 1 | 2 | A3 |
| MUL | AB | Multiply A and B | 1 | 4 | A4 |
| DIV | AB | Divide A by B | 1 | 4 | 84 |
| DA | A | Decimal adjust A | 1 | 1 | D4 |

**Table 85** Instruction set description: Logic operations

| MNEMONIC | | DESCRIPTION | BYTES | CYCLES | OPCODE (HEX) |
|---|---|---|---|---|---|
| **Logic operations** | | | | | |
| ANL | A,Rr | AND register to A | 1 | 1 | 5* |
| ANL | A,direct | AND direct byte to A | 2 | 1 | 55 |
| ANL | A,@Ri | AND indirect RAM to A | 1 | 1 | 56, 57 |
| ANL | A,#data | AND immediate data to A | 2 | 1 | 54 |
| ANL | direct,A | AND A to direct byte | 2 | 1 | 52 |
| ANL | direct,#data | AND immediate data to direct byte | 3 | 2 | 53 |
| ORL | A,Rr | OR register to A | 1 | 1 | 4* |
| ORL | A,direct | OR direct byte to A | 2 | 1 | 45 |
| ORL | A,@Ri | OR indirect RAM to A | 1 | 1 | 46, 47 |
| ORL | A,#data | OR immediate data to A | 2 | 1 | 44 |
| ORL | direct,A | OR A to direct byte | 2 | 1 | 42 |
| ORL | direct,#data | OR immediate data to direct byte | 3 | 2 | 43 |
| XRL | A,Rr | Exclusive-OR register to A | 1 | 1 | 6* |
| XRL | A,direct | Exclusive-OR direct byte to A | 2 | 1 | 65 |
| XRL | A,@Ri | Exclusive-OR indirect RAM to A | 1 | 1 | 66, 67 |
| XRL | A,#data | Exclusive-OR immediate data to A | 2 | 1 | 64 |
| XRL | direct,A | Exclusive-OR A to direct byte | 2 | 1 | 62 |
| XRL | direct,#data | Exclusive-OR immediate data to direct byte | 3 | 2 | 63 |
| CLR | A | Clear A | 1 | 1 | E4 |
| CPL | A | Complement A | 1 | 1 | F4 |
| RL | A | Rotate A left | 1 | 1 | 23 |
| RLC | A | Rotate A left through the carry flag | 1 | 1 | 33 |
| RR | A | Rotate A right | 1 | 1 | 03 |
| RRC | A | Rotate A right through the carry flag | 1 | 1 | 13 |
| SWAP | A | Swap nibbles within A | 1 | 1 | C4 |

**Table 86** Instruction set description: Data transfer

| MNEMONIC | | DESCRIPTION | BYTES | CYCLES | OPCODE (HEX) |
|---|---|---|---|---|---|
| **Data transfer** | | | | | |
| MOV | A,Rr | Move register to A | 1 | 1 | E* |
| MOV | A,direct (note 1) | Move direct byte to A | 2 | 1 | E5 |
| MOV | A,@Ri | Move indirect RAM to A | 1 | 1 | E6, E7 |
| MOV | A,#data | Move immediate data to A | 2 | 1 | 74 |
| MOV | Rr,A | Move A to register | 1 | 1 | F* |
| MOV | Rr,direct | Move direct byte to register | 2 | 2 | A* |
| MOV | Rr,#data | Move immediate data to register | 2 | 1 | 7* |
| MOV | direct,A | Move A to direct byte | 2 | 1 | F5 |
| MOV | direct,Rr | Move register to direct byte | 2 | 2 | 8* |
| MOV | direct,direct | Move direct byte to direct | 3 | 2 | 85 |
| MOV | direct,@Ri | Move indirect RAM to direct byte | 2 | 2 | 86, 87 |
| MOV | direct,#data | Move immediate data to direct byte | 3 | 2 | 75 |
| MOV | @Ri,A | Move A to indirect RAM | 1 | 1 | F6, F7 |
| MOV | @Ri,direct | Move direct byte to indirect RAM | 2 | 2 | A6, A7 |
| MOV | @Ri,#data | Move immediate data to indirect RAM | 2 | 1 | 76, 77 |
| MOV | DPTR,#data16 | Load data pointer with a 16-bit constant | 3 | 2 | 90 |
| MOVC | A,@A+DPTR | Move code byte relative to DPTR to A | 1 | 2 | 93 |
| MOVC | A,@A+PC | Move code byte relative to PC to A | 1 | 2 | 83 |
| MOVX | A,@Ri | Move external RAM (8-bit address) to A | 1 | 2 | E2, E3 |
| MOVX | A,@DPTR | Move external RAM (16-bit address) to A | 1 | 2 | E0 |
| MOVX | @Ri,A | Move A to external RAM (8-bit address) | 1 | 2 | F2, F3 |
| MOVX | @DPTR,A | Move A to external RAM (16-bit address) | 1 | 2 | F0 |
| PUSH | direct | Push direct byte onto stack | 2 | 2 | C0 |
| POP | direct | Pop direct byte from stack | 2 | 2 | D0 |
| XCH | A,Rr | Exchange register with A | 1 | 1 | C* |
| XCH | A,direct | Exchange direct byte with A | 2 | 1 | C5 |
| XCH | A,@Ri | Exchange indirect RAM with A | 1 | 1 | C6, C7 |
| XCHD | A,@Ri | Exchange LOW-order digit indirect RAM with A | 1 | 1 | D6, D7 |

**Note**

1. MOV A,ACC is not permitted.

**Table 87** Instruction set description: Boolean variable manipulation, Program and machine control

| MNEMONIC | | DESCRIPTION | BYTES | CYCLES | OPCODE (HEX) |
|---|---|---|---|---|---|
| **Boolean variable manipulation** | | | | | |
| CLR | C | Clear carry flag | 1 | 1 | C3 |
| CLR | bit | Clear direct bit | 2 | 1 | C2 |
| SETB | C | Set carry flag | 1 | 1 | D3 |
| SETB | bit | Set direct bit | 2 | 1 | D2 |
| CPL | C | Complement carry flag | 1 | 1 | B3 |
| CPL | bit | Complement direct bit | 2 | 1 | B2 |
| ANL | C,bit | AND direct bit to carry flag | 2 | 2 | 82 |
| ANL | C,/bit | AND complement of direct bit to carry flag | 2 | 2 | B0 |
| ORL | C,bit | OR direct bit to carry flag | 2 | 2 | 72 |
| ORL | C,/bit | OR complement of direct bit to carry flag | 2 | 2 | A0 |
| MOV | C,bit | Move direct bit to carry flag | 2 | 1 | A2 |
| MOV | bit,C | Move carry flag to direct bit | 2 | 2 | 92 |
| **Program and machine control** | | | | | |
| ACALL | addr11 | Absolute subroutine call | 2 | 2 | •1 |
| LCALL | addr16 | Long subroutine call | 3 | 2 | 12 |
| RET | | Return from subroutine | 1 | 2 | 22 |
| RETI | | Return from interrupt | 1 | 2 | 32 |
| AJMP | addr11 | Absolute jump | 2 | 2 | ♦1 |
| LJMP | addr16 | Long jump | 3 | 2 | 02 |
| SJMP | rel | Short jump (relative address) | 2 | 2 | 80 |
| JMP | @A+DPTR | Jump indirect relative to the DPTR | 1 | 2 | 73 |
| JZ | rel | Jump if A is zero | 2 | 2 | 60 |
| JNZ | rel | Jump if A is not zero | 2 | 2 | 70 |
| JC | rel | Jump if carry flag is set | 2 | 2 | 40 |
| JNC | rel | Jump if carry flag is not set | 2 | 2 | 50 |
| JB | bit,rel | Jump if direct bit is set | 3 | 2 | 20 |
| JNB | bit,rel | Jump if direct bit is not set | 3 | 2 | 30 |
| JBC | bit,rel | Jump if direct bit is set and clear bit | 3 | 2 | 10 |
| CJNE | A,direct,rel | Compare direct to A and jump if not equal | 3 | 2 | B5 |
| CJNE | A,#data,rel | Compare immediate to A and jump if not equal | 3 | 2 | B4 |
| CJNE | Rr,#data,rel | Compare immediate to register and jump if not equal | 3 | 2 | B* |
| CJNE | @Ri,#data,rel | Compare immediate to indirect and jump if not equal | 3 | 2 | B6, B7 |
| DJNZ | Rr,rel | Decrement register and jump if not zero | 2 | 2 | D* |
| DJNZ | direct,rel | Decrement direct and jump if not zero | 3 | 2 | D5 |
| NOP | | No operation | 1 | 1 | 00 |

**Table 88** Description of the mnemonics in the Instruction set

| MNEMONIC | DESCRIPTION |
|---|---|
| **Data addressing modes** | |
| Rr | Working register R0-R7. |
| direct | 128 internal RAM locations and any special function register (SFR). |
| @Ri | Indirect internal RAM location addressed by register R0 or R1 of the actual register bank. |
| #data | 8-bit constant included in instruction. |
| #data 16 | 16-bit constant included as bytes 2 and 3 of instruction. |
| bit | Direct addressed bit in internal RAM or SFR. |
| addr16 | 16-bit destination address. Used by LCALL and LJMP.<br>The branch will be anywhere within the 64 kbytes Program Memory address space. |
| addr11 | 11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 kbytes page of Program Memory as the first byte of the following instruction. |
| rel | Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps.<br>Range is –128 to +127 bytes relative to first byte of the following instruction. |
| **Hexadecimal opcode cross-reference** | |
| * | 8, 9, A, B, C, D, E, F. |
| • | 1, 3, 5, 7, 9, B, D, F. |
| ♦ | 0, 2, 4, 6, 8, A, C, E. |

**Table 89** Instruction map

| First hexadecimal character of opcode ↓ | ← Second hexadecimal character of opcode → 0 | 1 | 2 | 3 | 4 | 5 | 6 7 | 8 9 A B C D E F |
|---|---|---|---|---|---|---|---|---|
| 0 | NOP | AJMP addr11 | LJMP addr16 | RR A | INC A | INC direct | INC @Ri: 0 1 | INC Rr: 0 1 2 3 4 5 6 7 |
| 1 | JBC bit,rel | ACALL addr11 | LCALL addr16 | RRC A | DEC A | DEC direct | DEC @Ri: 0 1 | DEC Rr: 0 1 2 3 4 5 6 7 |
| 2 | JB bit,rel | AJMP addr11 | RET | RL A | ADD A,#data | ADD A,direct | ADD A,@Ri: 0 1 | ADD A,Rr: 0 1 2 3 4 5 6 7 |
| 3 | JNB bit,rel | ACALL addr11 | RETI | RLC A | ADDC A,#data | ADDC A,direct | ADDC A,@Ri: 0 1 | ADDC A,Rr: 0 1 2 3 4 5 6 7 |
| 4 | JC rel | AJMP addr11 | ORL direct,A | ORL direct,#data | ORL A,#data | ORL A,direct | ORL A,@Ri: 0 1 | ORL A,Rr: 0 1 2 3 4 5 6 7 |
| 5 | JNC rel | ACALL addr11 | ANL direct,A | ANL direct,#data | ANL A,#data | ANL A,direct | ANL A,@Ri: 0 1 | ANL A,Rr: 0 1 2 3 4 5 6 7 |
| 6 | JZ rel | AJMP addr11 | XRL direct,A | XRL direct,#data | XRL A,#data | XRL A,direct | XRL A,@Ri: 0 1 | XRL A,Rr: 0 1 2 3 4 5 6 7 |
| 7 | JNZ rel | ACALL addr11 | ORL C,bit | JMP @A+DPTR | MOV A,#data | MOV direct,#data | MOV @Ri,#data: 0 1 | MOV Rr,#data: 0 1 2 3 4 5 6 7 |
| 8 | SJMP rel | AJMP addr11 | ANL C,bit | MOVC A,@A+PC | DIV AB | MOV direct,direct | MOV direct,@Ri: 0 1 | MOV direct,Rr: 0 1 2 3 4 5 6 7 |
| 9 | MOV DTPR,#data16 | ACALL addr11 | MOV bit,C | MOVC A,@A+DPTR | SUBB A,#data | SUBB A,direct | SUBB A,@Ri: 0 1 | SUB A,Rr: 0 1 2 3 4 5 6 7 |
| A | ORL C,/bit | AJMP addr11 | MOV bit,C | INC DPTR | MUL AB | | MOV @Ri,direct: 0 1 | MOV Rr,direct: 0 1 2 3 4 5 6 7 |
| B | ANL C,/bit | ACALL addr11 | CPL bit | CPL C | CJNE A,#data,rel | CJNE A,direct,rel | CJNE @Ri,#data,rel: 0 1 | CJNE Rr,#data,rel: 0 1 2 3 4 5 6 7 |
| C | PUSH direct | AJMP addr11 | CLR bit | CLR C | SWAP A | XCH A,direct | XCH A,@Ri: 0 1 | XCH A,Rr: 0 1 2 3 4 5 6 7 |
| D | POP direct | ACALL addr11 | SETB bit | SETB C | DA A | DJNZ direct,rel | XCHD A,@Ri: 0 1 | DJNZ Rr,rel: 0 1 2 3 4 5 6 7 |
| E | MOVX A,@DTPR | AJMP addr11 | MOVX A,@Ri: 0 1 | | CLR A | MOV A,direct (1) | MOV A,@Ri: 0 1 | MOV A,Rr: 0 1 2 3 4 5 6 7 |
| F | MOVX @DTPR,A | ACALL addr11 | MOVX @Ri,A: 0 1 | | CPL A | MOV direct,A | MOV @Ri,A: 0 1 | MOV Rr,A: 0 1 2 3 4 5 6 7 |

**Note**

1. MOV A, ACC is not a valid instruction.

## 19 ABSOLUTE MAXIMUM RATINGS (note 1)

In accordance with the Absolute Maximum Rating System (IEC 134).

| SYMBOL | PARAMETER | MIN. | MAX. | UNIT |
|---|---|---|---|---|
| $V_{DD}$ | voltage on $V_{DD}$ pin | –0.5 | +6.5 | V |
| $V_{I1}$ | input voltage on any pin (except CTX0, CTX1, CRX0, CRX1 and $\overline{EA}/V_{PP}$) | –0.5 | $V_{DD}$ + 0.5 | V |
| $V_{I2}$ | input voltage on $\overline{EA}/V_{PP}$ to $V_{SS}$ | –0.5 | +13 | V |
| $I_I$, $I_O$ | input/output current on any single I/O pin (except from CTX0 and CTX1) | – | ±10 | mA |
| $I_{OT}$ | sink current of CTX0, CTX1 together | – | 30 | mA |
| | source current of CTX0, CTX1 together | – | –20 | mA |
| $P_{tot}$ | total power dissipation (note 2) | – | 1.0 | W |
| $T_{stg}$ | storage temperature range | –65 | +150 | °C |
| $T_{amb}$ | operating ambient temperature range: | | | |
| | P8xC592 FFA | –40 | +85 | °C |
| | P8xC592 FHA | –40 | +125 | °C |

**Notes**

1. The following applies to the Absolute Maximum Ratings:
   a) Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the Chapters 20 "DC characteristics" and 21 "AC characteristics" of this specification is not implied.
   b) This product includes circuitry specifically designed for the protection of its internal devices from the damaging effect of excessive static charge. However, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
   c) Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.
2. This value is based on the maximum allowable die temperature and the thermal resistance of the package, not on device power consumption.

## 20 DC CHARACTERISTICS

$V_{DD}$ = 5 V ± 10%; $V_{SS}$ = 0 V; all voltages with respect to $V_{SS}$ unless otherwise specified.
$T_{amb}$ = −40 to +125 °C for the **P8xC592FHA**; $T_{amb}$ = −40 to +85 °C for the **P8xC592FFA**.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| **Supply (digital part)** | | | | | |
| $V_{DD}$ | supply voltage | | 4.5 | 5.5 | V |
| $I_{DD}$ | operating supply current | $f_{CLK}$ = 16 MHz; note 1 | – | 50 | mA |
| $I_{DD(ID)}$ | supply current Idle mode | $f_{CLK}$ = 16 MHz; note 2 | – | 15 | mA |
| $I_{DD(IS)}$ | supply current Idle & Sleep mode | $f_{CLK}$ = 16 MHz; note 3 | – | 10 | mA |
| $I_{DD(PD)}$ | supply current Power-down mode:<br>P8xC592 FHA<br>P8xC592 xFx | note 4 | <br>–<br>– | <br>150<br>50 | <br>µA<br>µA |
| **Inputs** | | | | | |
| $V_{IL}$ | LOW level input voltage (except $\overline{EA}$, CRX0 and CRX1) | | −0.5 | $0.2V_{DD}$ − 0.1 | V |
| $V_{IL1}$ | LOW level input voltage $\overline{EA}$ | | −0.5 | $0.2V_{DD}$ − 0.3 | V |
| $V_{IH}$ | HIGH level input voltage (except RST, XTAL1, CRX0,CRX1) | | $0.2V_{DD}$ + 0.9 | $V_{DD}$ + 0.5 | V |
| $V_{IH1}$ | HIGH level input voltage (RST and XTAL1) | | $0.7V_{DD}$ | $V_{DD}$ + 0.5 | V |
| $I_{IL}$ | LOW level input current Ports 1, 2, 3 and 4 | $V_I$ = 0.45 V | – | −50 | µA |
| $I_{TL}$ | input current HIGH-to-LOW transition Ports 1, 2, 3 and 4 (except P1.6 and P1.7) | $V_I$ = 2.0 to 0.45 V | – | −650 | µA |
| $I_{LI1}$ | input leakage current Port 0, $\overline{EA}$, STADC, $\overline{EW}$, P1.6, P1.7 | 0.45 V < $V_I$ < $V_{DD}$ | – | ±10 | µA |
| $I_{LI2}$ | input leakage current Port 5 | 0.45 V < $V_I$ < $V_{DD}$ | – | ±1 | µA |
| **Outputs** | | | | | |
| $V_{OL}$ | LOW level output voltage Ports 1, 2, 3 and 4 (except P1.6 and P1.7) | $I_{OL}$ = 1.6 mA; note 5 | – | 0.45 | V |
| $V_{OL1}$ | LOW level output voltage Port 0, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$, P1.6, P1.7 | $I_{OL}$ = 3.2 mA; note 5 | – | 0.45 | V |
| $V_{OH}$ | HIGH level output voltage Ports 1, 2, 3 and 4 (except P1.6 and P1.7) | $I_{OH}$ = −60 µA<br>$I_{OH}$ = −25 µA<br>$I_{OH}$ = −10 µA | 2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | –<br>–<br>– | V<br>V<br>V |
| $V_{OH1}$ | HIGH level output voltage Port 0 in external bus mode, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$ | $I_{OH}$ = −400 µA<br>$I_{OH}$ = −150 µA<br>$I_{OH}$ = −40 µA; note 6 | 2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | –<br>–<br>– | V<br>V<br>V |

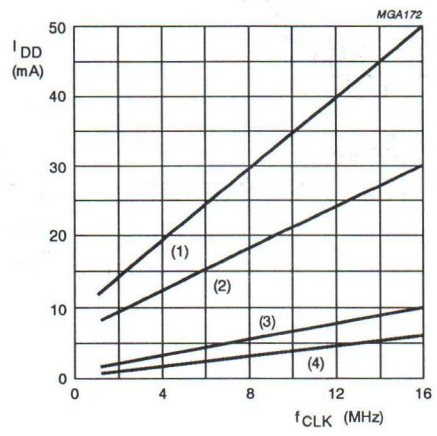
| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| $V_{OH2}$ | HIGH level output voltage RST | $I_{OH}$ = –400 µA | 2.4 | – | V |
| | | $I_{OH}$ = –120 µA | $0.8V_{DD}$ | – | V |
| $R_{RST}$ | RST pull-down resistor | | 50 | 150 | kΩ |
| $C_{I/O}$ | I/O pin capacitance | test frequency = 1 MHz; $T_{amb}$ = 25 °C | – | 10 | pF |
| **Supply (analog part)** | | | | | |
| $AV_{DD}$ | supply voltage | $AV_{DD} = V_{DD} \pm 0.2$ V | 4.5 | 5.5 | V |
| $AI_{DD}$ | operating supply current | Port 5 = $AV_{DD}$; note 1 | – | 2.5 | mA |
| $AI_{DD(ID)}$ | supply current Idle mode | note 2 | – | 2.5 | mA |
| $AI_{DD(IS)}$ | supply current Idle and Sleep mode: | note 3 | | | |
| | P83C592 FHA | | – | 400 | µA |
| | P8xC592 xFx | | – | 350 | µA |
| $AI_{DD(PD)}$ | supply current Power-down mode: | note 4 | | | |
| | P83C592 FHA | | – | 400 | µA |
| | P8xC592 xFx | | – | 350 | µA |
| **Analog inputs** | | | | | |
| $AV_{IN}$ | analog input voltage | | $AV_{SS}$ – 0.2 | $AV_{DD}$ + 0.2 | V |
| $AV_{REF-}$ | reference voltage | | $AV_{SS}$ – 0.2 | – | V |
| $AV_{REF+}$ | | | – | $AV_{DD}$ + 0.2 | V |
| $R_{REF}$ | resistance between $AV_{REF+}$ and $AV_{REF-}$ | | 10 | 50 | kΩ |
| $C_{IA}$ | analog input capacitance | | – | 15 | pF |
| $t_{ADS}$ | sampling time | note 7 | – | $8t_{CY}$ | µs |
| $t_{ADC}$ | conversion time (including sample time) | note 7 | – | $50t_{CY}$ | µs |
| $DL_e$ | differential non-linearity | notes 8, 9 and 10 | – | ±1 | LSB |
| $IL_e$ | integral non-linearity | notes 8 and 11 | – | ±2 | LSB |
| $OS_e$ | offset error | notes 8 and 12 | – | ±2 | LSB |
| $G_e$ | gain error | notes 8 and 13 | – | ±0.4 | % |
| $A_e$ | absolute voltage error | notes 8 and 14 | – | ±3 | LSB |
| $M_{ctc}$ | channel to channel matching | | – | ±1 | LSB |
| $C_t$ | crosstalk between P5 inputs | 0 to 100 kHz | – | –60 | dB |
| **CAN input comparator (CRX0, CRX1)** | | | | | |
| $V_{DIF}$ | differential input voltage (note 15) | $AV_{DD}$ = 5 V ± 5%; 1.4 V < $V_I$ < $AV_{DD}$–1.4 V | ±32 | – | mV |
| $V_{HYST}$ | hysteresis voltage (note 15) | | 8 | 30 | mV |
| $I_I$ | input current | | – | ±400 | nA |

| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| **CAN output driver ($V_{DD}$ = 5 V ± 5%)** | | | | | |
| $V_{OLT}$ | LOW level output voltage (CTX0 and CTX1) | $I_o$ = 1.2 mA; note 15 | – | 0.1 | V |
| | | $I_o$ = 10 mA | – | 0.6 | V |
| $V_{OHT}$ | High level output voltage (CTX0 and CTX1) | $I_o$ = –1.2 mA; note 15 | $V_{DD}$ – 0.1 | – | V |
| | | $I_o$ = –10 mA; note 16 | $V_{DD}$ – 0.6 | – | V |
| **Reference ($AV_{DD}$ = 5 V ± 5%)** | | | | | |
| $V_{REFOUT}$ | REF output voltage | –0.1 mA < $I_L$ < 0.1 mA; $C_L$ = 10 nF; note 15; bit Reference Active = HIGH | $\frac{1}{2}AV_{DD}$–0.1 | $\frac{1}{2}AV_{DD}$+0.1 | V |
| $I_{REFIN}$ | REF input current | 1.5 V < $V_{REFIN}$ < $AV_{DD}$–1.5 V; bit Reference Active = LOW | – | ±10 | µA |

**Notes to the DC characteristics**

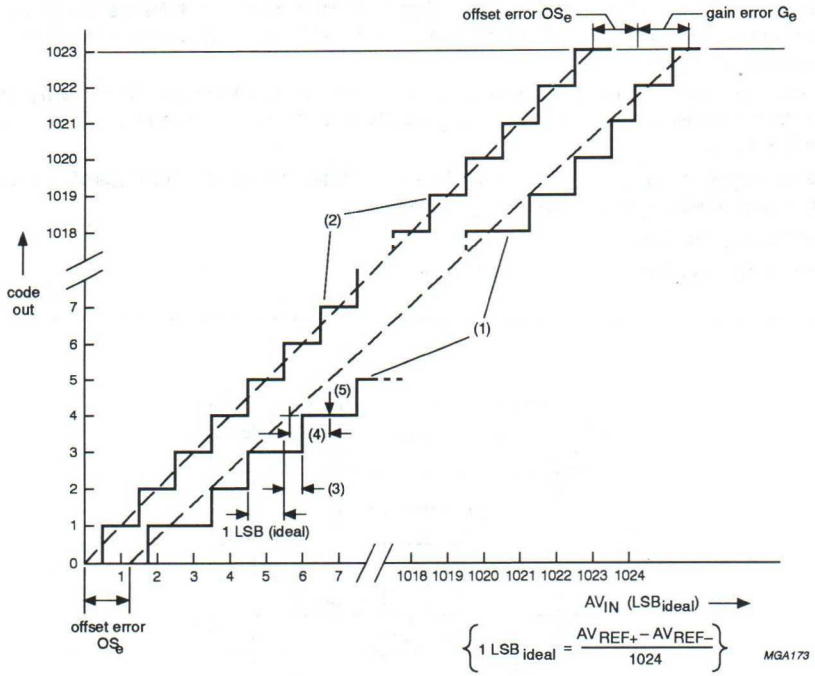
1. Conditions for:
   a) The **digital** operating current measurement: all output pins disconnected; XTAL1 is driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0.5 V; $V_{IH} = V_{DD}$ – 0.5 V; $\overline{EA}$ = RST = Port 0 = P1.6 = P1.7 = $\overline{EW}$ = $V_{DD}$; STADC = $V_{SS}$; CRX0 = 2.7 V; CRX1 = 2.3 V.
   b) The **analog** operating current measurement: Port 5 = $AV_{DD}$; CAN: register 6: = 00H; load current reference voltage source 100 µA.
2. Conditions for:
   a) The **digital** Idle mode supply current measurement: all output pins disconnected; XTAL1 is driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0.5 V; $V_{IH} = V_{DD}$ – 0.5 V; Port 0 = P1.6 = P1.7 = $\overline{EW}$ = $V_{DD}$; $\overline{EA}$ = RST = STADC = $V_{SS}$; CRX0 = 2.7 V; CRX1 = 2.3 V.
   b) The **analog** Idle mode current measurement: Port 5 = $AV_{DD}$; CAN: register 6: = 00H; load current reference voltage source 100 µA.
3. Conditions for:
   a) The **digital** Idle and Sleep mode supply current measurement: all output pins disconnected; XTAL1 is driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0.5 V; $V_{IH} = V_{DD}$ – 0.5 V; Port 0 = P1.6 = P1.7 = $\overline{EW}$ = CRX0 = $V_{DD}$; $\overline{EA}$ = RST = STADC = CRX1 = $V_{SS}$; CAN: register 6: = 00H, register 7: = 12H, register 8: = 02H, register 0: = 20H, wait 15$t_{CY}$, register 1: = 10H, wait for bit Sleep = 1.
   b) The **analog** Idle and Sleep mode current measurement: Port 5 = $AV_{DD}$; load current reference voltage source 100 µA.
4. Window devices have to be covered. Conditions for:
   a) The **digital** Power-down mode supply current measurement: all output pins and Port 5 disconnected; Port 0 = P1.6 = P1.7 = $\overline{EW}$ = CRX0 = $V_{DD}$; $\overline{EA}$ = RST = STADC = CRX1 = XTAL1 = $AV_{REF+}$ = $AV_{REF-}$ = $CV_{SS}$ = $V_{SS}$; $AV_{DD}$ = $V_{DD}$, but current into $AV_{DD}$ pin is not comprised in digital Power-down current.
   b) The **analog** Power-down mode supply current measurement: Port 5 = $AV_{DD}$.
5. Capacitive loads on Port 0 and Port 2 may degrade the LOW level output voltage of ALE, Port 1 and Port 3. During a HIGH-to-LOW transition on the Port 0 and Port 2 pins and a capacitive load >100 pF, the ALE LOW level may exceed 0.8 V. In the case that it is necessary to connect ALE to a Schmitt trigger input respectively use an address latch with a Schmitt trigger STROBE input.

6. Capacitive loads on Port 0 and Port 2 may cause a HIGH level voltage degradation of ALE and $\overline{PSEN}$ below $0.9V_{DD}$ during the address bits are stabilizing.
7. $t_{CY} = 12\ t_{CLK}$ is the machine cycle time.
8. $AV_{REF+} = 5.12$ V; $AV_{REF-} = 0$ V; $AV_{DD} = 5.0$ V.
9. The differential non-linearity ($DL_e$) is the difference between the actual step width and the ideal step width.
10. The ADC is monotonic, there are no missing codes.
11. The integral non-linearity ($IL_e$) is the peak difference between the centre of the steps of the actual and the ideal transfer curve after appropriate adjustment of gain and offset error.
12. The offset error ($OS_e$) is the absolute difference between the straight line which fits the actual transfer curve after removing gain error, and a straight line which fits the ideal transfer curve. The offset error is constant at every point of the actual transfer curve.
13. The gain error ($G_e$) is relative difference in percent between the straight line fitting the actual transfer curve after removing offset error and the straight line which fits the ideal transfer curve. The gain error is constant at every point on the transfer curve.
14. The absolute voltage error ($A_e$) is the maximum difference between the centre of the steps of the actual transfer curve of the not calibrated ADC and the ideal transfer curve.
15. Not tested during production.
16. Source current for the CTX0, CTX1 outputs together.




(1) Maximum Operating mode ($I_{DD}$); $V_{DD} = 5.5$ V
(2) Maximum Operating mode ($I_{DD}$); $V_{DD} = 4.5$ V
(3) Maximum Idle and Sleep mode ($I_{DD(IS)}$); $V_{DD} = 5.5$ V
(4) Maximum Idle and Sleep mode ($I_{DD(IS)}$); $V_{DD} = 4.5$ V

Fig.27 Supply current ($I_{DD}$) as a function of frequency at XTAL1 ($f_{CLK}$).




(1) Example of an actual transfer curve.
(2) The ideal transfer curve.
(3) Differential non-linearity ($DL_e$).
(4) Integral non-linearity ($IL_e$).
(5) Centre of a step of the actual transfer curve.

Fig.28 ADC conversion characteristic.

## 21 AC CHARACTERISTICS

See notes 1 and 2; $C_L$ = 100 pF for Port 0, ALE and $\overline{PSEN}$; $C_L$ = 80 pF for all other outputs unless otherwise specified.

| SYMBOL | PARAMETER | $f_{CLK}$ = 16 MHz | | $f_{CLK}$ = 12 MHz | | VARIABLE CLOCK 1.2 to 16 MHz | | UNIT |
|---|---|---|---|---|---|---|---|---|
| | | MIN. | MAX. | MIN. | MAX. | MIN. | MAX. | |
| **External Program Memory** | | | | | | | | |
| $t_{LHLL}$ | ALE pulse width | 85 | – | 127 | – | $2t_{CLK} - 40$ | – | ns |
| $t_{AVLL}$ | address valid to ALE LOW | 23 | – | 43 | – | $t_{CLK} - 40$ | – | ns |
| $t_{LLAX}$ | address hold after ALE LOW | 33 | – | 53 | – | $t_{CLK} - 30$ | – | ns |
| $t_{LLIV}$ | ALE LOW to valid instruction in | – | 150 | – | 233 | – | $4t_{CLK} - 100$ | ns |
| $t_{LLPL}$ | ALE LOW to $\overline{PSEN}$ LOW | 33 | – | 53 | – | $t_{CLK} - 30$ | – | ns |
| $t_{PLPH}$ | $\overline{PSEN}$ pulse width | 143 | – | 205 | – | $3t_{CLK} - 45$ | – | ns |
| $t_{PLIV}$ | $\overline{PSEN}$ LOW to valid instruction in | – | 83 | – | 145 | – | $3t_{CLK} - 105$ | ns |
| $t_{PXIX}$ | input instruction hold after $\overline{PSEN}$ | 0 | – | 0 | – | 0 | – | ns |
| $t_{PXIZ}$ | input instruction float after $\overline{PSEN}$ | – | 38 | – | 59 | – | $t_{CLK} - 25$ | ns |
| $t_{AVIV}$ | address to valid instruction in | – | 208 | – | 312 | – | $5t_{CLK} - 105$ | ns |
| $t_{PLAZ}$ | $\overline{PSEN}$ LOW to address float | – | 10 | – | 10 | – | 10 | ns |
| **External data memory** | | | | | | | | |
| $t_{RLRH}$ | $\overline{RD}$ pulse width | 275 | – | 400 | – | $6t_{CLK} - 100$ | – | ns |
| $t_{WLWH}$ | $\overline{WR}$ pulse width | 275 | – | 400 | – | $6t_{CLK} - 100$ | – | ns |
| $t_{AVLL}$ | address valid to ALE LOW | 8 | – | 28 | – | $t_{CLK} - 55$ | – | ns |
| $t_{LLAX}$ | address hold after ALE LOW | 33 | – | 53 | – | $t_{CLK} - 30$ | – | ns |
| $t_{RLDV}$ | $\overline{RD}$ LOW to valid data in | – | 148 | – | 252 | – | $5t_{CLK} - 165$ | ns |
| $t_{RHDX}$ | data hold after $\overline{RD}$ | 0 | – | 0 | – | 0 | – | ns |
| $t_{RHDZ}$ | data float after $\overline{RD}$ | – | 55 | – | 97 | – | $2t_{CLK} - 70$ | ns |
| $t_{LLDV}$ | ALE LOW to valid data in | – | 350 | – | 517 | – | $8t_{CLK} - 150$ | ns |
| $t_{AVDV}$ | address to valid data in | – | 398 | – | 585 | – | $9t_{CLK} - 165$ | ns |
| $t_{LLWL}$ | ALE LOW to $\overline{RD}$ or $\overline{WR}$ LOW | 138 | 238 | 200 | 300 | $3t_{CLK} - 50$ | $3t_{CLK} + 50$ | ns |
| $t_{AVWL}$ | address valid to $\overline{RD}$ or $\overline{WR}$ LOW | 120 | – | 203 | – | $4t_{CLK} - 130$ | – | ns |
| $t_{WHLH}$ | $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH | 23 | 103 | 43 | 123 | $t_{CLK} - 40$ | $t_{CLK} + 40$ | ns |
| $t_{QVWX}$ | data valid to $\overline{WR}$ transition | 13 | – | 33 | – | $t_{CLK} - 50$ | – | ns |
| $t_{QVWH}$ | data valid time $\overline{WR}$ HIGH | 288 | – | 433 | – | $7t_{CLK} - 150$ | – | ns |
| $t_{WHQX}$ | data hold after $\overline{WR}$ | 13 | – | 33 | – | $t_{CLK} - 50$ | – | ns |
| $t_{RLAZ}$ | $\overline{RD}$ LOW to address float | – | 0 | – | 0 | – | 0 | ns |

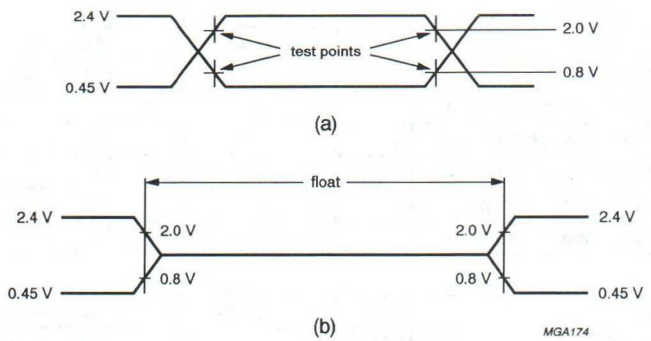
**Notes**

1. For the AC Characteristics the following conditions are valid: **P8xC592 FFA (FHA)**: $V_{DD}$ = 5 V ± 10%; $T_{amb}$ = –40 to +85 °C (125 °C); $f_{CLK}$ = 1.2 to 16 MHz.

2. $t_{CLK} = \frac{1}{f_{CLK}}$ = one oscillator clock period ; $t_{CLK}$ = 62.5 ns at $f_{CLK}$ = 16 MHz.

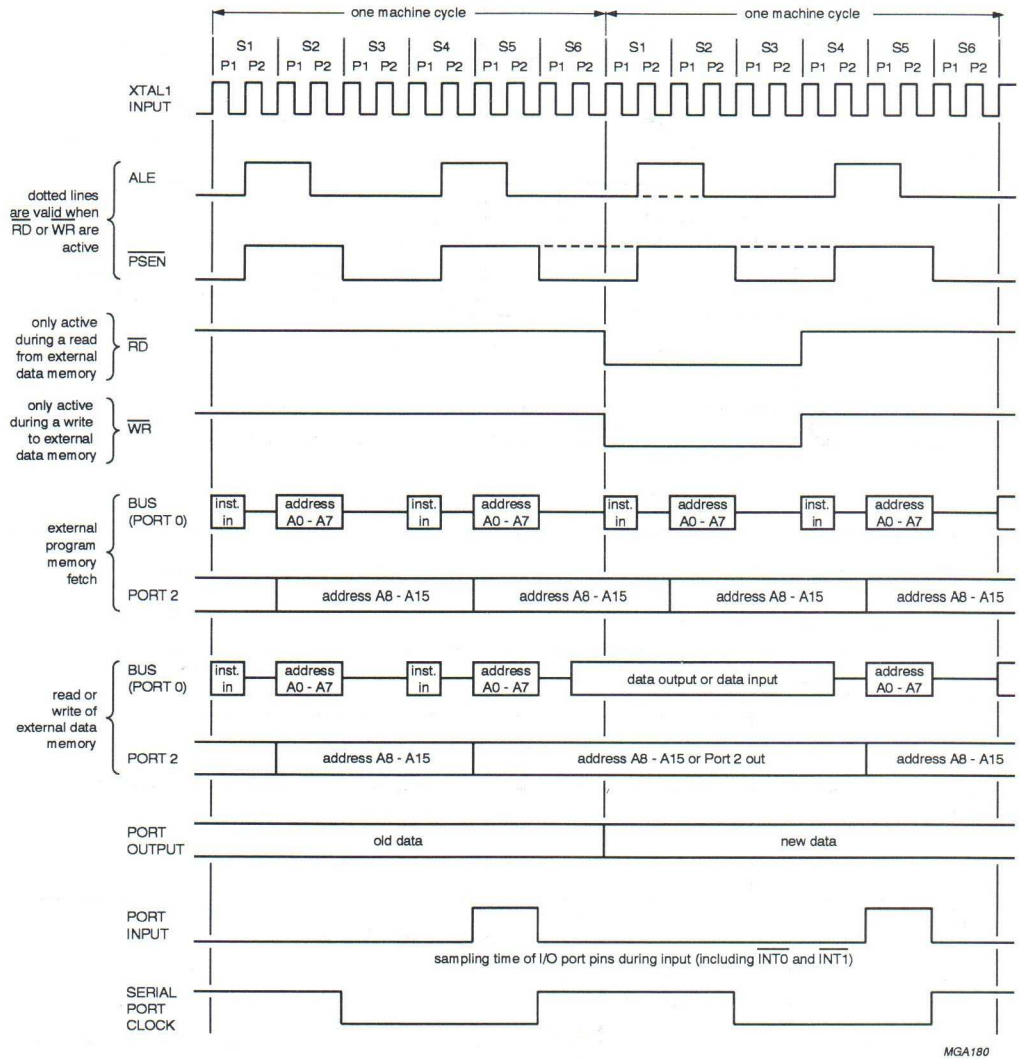
**Table 90** CAN characteristics

| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| **CAN input comparator/output driver** | | | | | |
| $t_{sd}$ | sum of input and output delay | $AV_{DD}$ = 5 V ± 5%; $V_{DIF}$ = ± 32 mV; 1.4 V < $V_I$ < $AV_{DD}$ − 1.4 V | – | 60 | ns |




AC testing inputs are driven at 2.4 V for a HIGH and 0.45 V for a LOW.
Timing measurements are taken at 2.0 V for a HIGH and 0.8 V for a LOW, see Fig.29 (a).
The float state is defined as the point at which a Port 0 pin sinks 3.2 mA or sources 400 μA at the voltage test levels, see Fig.29 (b).

Fig.29 AC testing input, output waveform (a) and float waveform (b).




The Port 5 input buffers have a maximum propagation delay of 300 ns. As a result Port 5 sample time begins 300 ns before state S5 and ends when S5 has finished.

Fig.30 Instruction cycle timing.

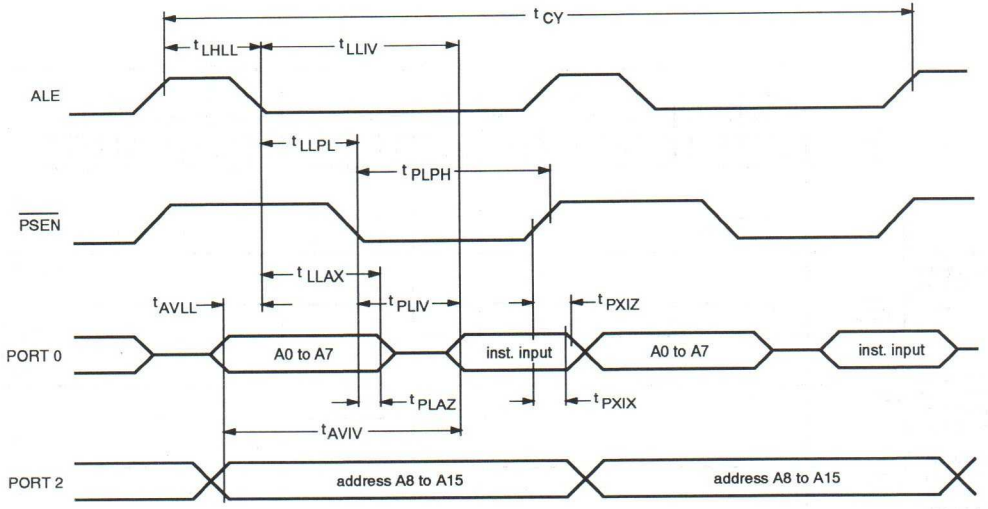



Fig.31 Read from external Program Memory.

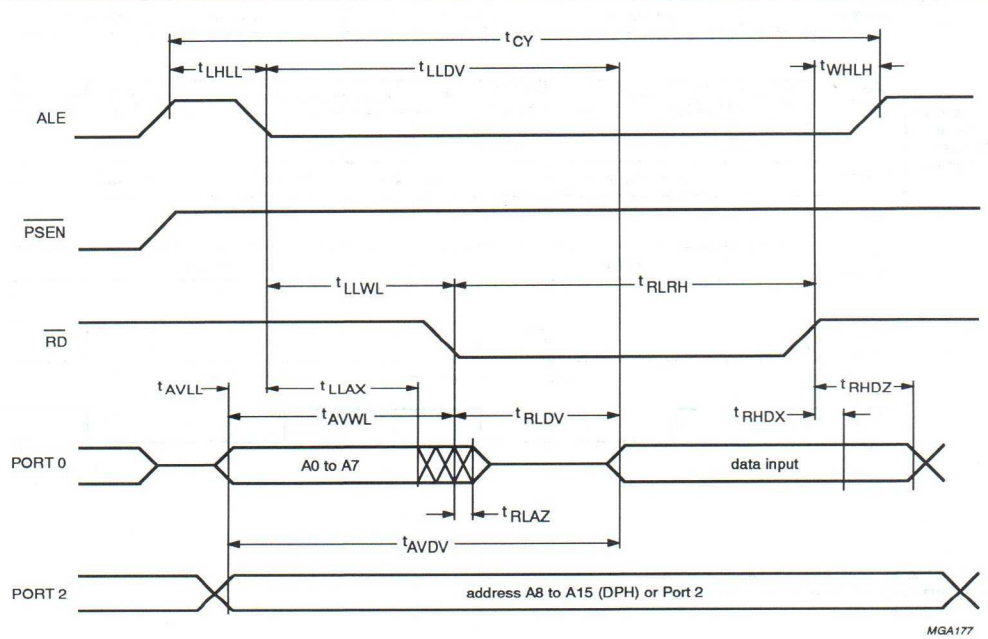



Fig.32 Read from external Data Memory.

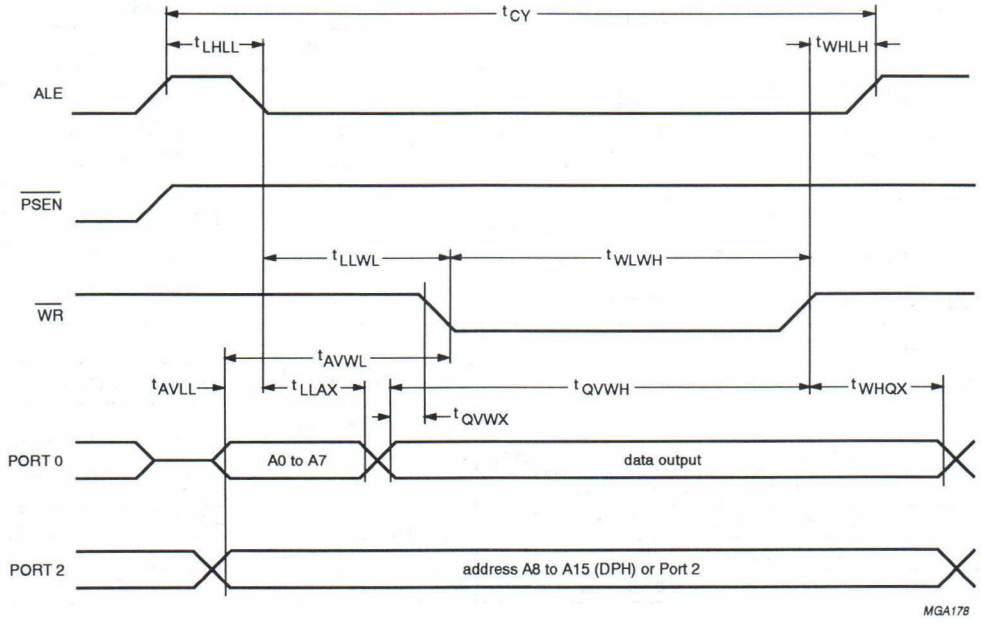



Fig.33 Write to external Data Memory.

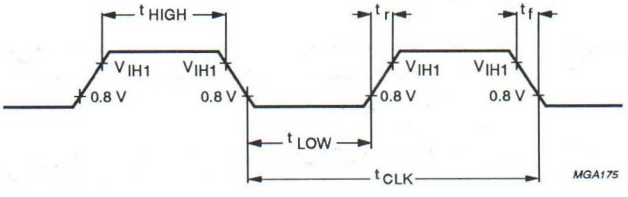



Fig.34 External clock drive XTAL1 (see Table 91).

**Table 91** External clock drive XTAL1

| SYMBOL | PARAMETER | VARIABLE CLOCK ($f_{CLK}$ = 1.2 to 16 MHz) | | UNIT |
|---|---|---|---|---|
| | | MIN. | MAX. | |
| $t_{CLK}$ | oscillator clock period (P83C592) | 62.5 | 833.3 | ns |
| $t_{HIGH}$ | HIGH time | 20 | $t_{CLK} - t_{LOW}$ | ns |
| $t_{LOW}$ | LOW time | 20 | $t_{CLK} - t_{HIGH}$ | ns |
| $t_r$ | rise time | – | 20 | ns |
| $t_f$ | fall time | – | 20 | ns |
| $t_{CY}$ | cycle time ($12 \times t_{CLK}$) | 0.75 | 10 | µs |

**Table 92** UART Timing in Shift Register Mode

| SYMBOL | PARAMETER | $f_{CLK}$ | | | | | | UNIT |
|---|---|---|---|---|---|---|---|---|
| | | 16 MHz | | 12 MHz | | VARIABLE CLOCK | | |
| | | MIN. | MAX. | MIN. | MAX. | MIN. | MAX. | |
| $t_{XLXL}$ | Serial Port clock cycle timing | 0.75 | – | 1.0 | – | $12t_{CLK}$ | – | ms |
| $t_{QVXH}$ | output data setup to clock rising edge | 492 | – | 700 | – | $10t_{CLK} - 133$ | – | ns |
| $t_{XHQX}$ | output data hold after clock rising edge | 8.0 | – | 50 | – | $2t_{CLK} - 117$ | – | ns |
| $t_{XHDX}$ | input data hold after clock rising edge | 0 | – | 0 | – | 0 | – | ns |
| $t_{XHDV}$ | clock rising edge to input data valid | – | 492 | – | 700 | – | $10t_{CLK} - 133$ | ns |

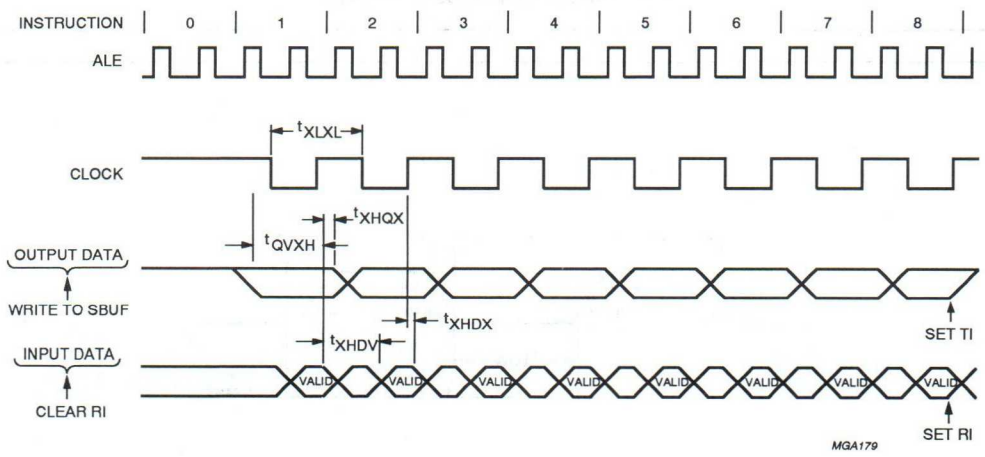



Fig.35 UART waveforms in Shift Register Mode.

## 22 CAN APPLICATION INFORMATION

### 22.1 Latency time requirements

Real-time applications require the ability to process and transfer information in a limited and predetermined period of time. If knowing this total time and the time required to process the information, the (maximum allowed) transfer delay time is given.

It is measured from the initiation of the transfer up to the signalling of reception.
For instance, this is the period of time between programming the CAN Command Register bit 0 (Transmission Request) to HIGH and the time getting an interrupt at a receiving CAN-device (due to the reception of the respective message).

#### 22.1.1 MAXIMUM ALLOWED BIT-TIME CALCULATION

The maximum allowed bit-time ($t_{BIT}$) due to latency time requirements can be calculated as:

$$t_{BIT} \leq \frac{t_{MAX\ TRANSFER\ TIME}}{(n_{BIT,\ MAX\ LATENCY} + n_{BIT,\ MESSAGE})} \quad (1)$$

Where:

- $t_{MAX\ TRANSFER\ TIME}$:
  the maximum allowed transfer delay time (application-specific).
- $n_{BIT,\ MAX\ LATENCY}$:
  the maximum latency time (in terms of number of bits), which depends on the actual state of the CAN network (e.g. another message already on the network);
- $n_{BIT,\ MESSAGE}$:
  the number of bits of a message; it varies with the number of transferred data bytes $n_{DATA\ BYTES}$ (0..8) and Stuffbits like:

$$44 + 8.n_{DATA\ BYTES} \leq n_{BIT,\ MESSAGE} \leq 52 + 10.n_{DATABYTES} \quad (2)$$

Example:

For the calculation of $n_{BIT,\ MAX\ LATENCY}$ the following is assumed (the term 'our message' refers to that one the latency time is calculated for):

- since at maximum one-bit-time ago another CAN-controller is transmitting.
- a single error occurs during the transmission of that message preceding ours, leading to the additional transfer of one Error Frame
- 'our message' has the highest priority,

giving:

$$n_{BIT,\ MAX\ LATENCY} \geq 44 + 8.n_{DATA\ BYTES,\ WORST\ CASE} + 18 \quad (3)$$

$$n_{BIT,\ MAX\ LATENCY} \leq 52 + 10.n_{DATA\ BYTES,\ WORST\ CASE} + 18 \quad (4)$$

Where:

- The additional 18 bits are due to the Error Frame and the Intermission Field preceding 'our message'.
- $n_{DATA\ BYTES,\ WORST\ CASE}$ denotes the number of data bytes contained by the longest message being used in a given CAN network.

22.1.2 CALCULATING THE MAXIMUM BIT-TIME

**Table 93** Example for calculating the maximum bit-time

| STATEMENT | COMMENTS |
|---|---|
| $t_{MAX\ TRANSFER\ TIME} = 10$ ms | assumption |
| $n_{DATA\ BYTES,\ WORST\ CASE} = 6$ | longest message in that network; assumption |
| $n_{DATA\ BYTES} = 4$ | 'our message'; assumption |
| $n_{BIT\ MAX\ LATENCY} \leq 130$ | using Equation (3) and (4) |
| $n_{MESSAGE} \leq 92$ | using Equation (2) |
| $t_{BIT} \leq \frac{10\ ms}{(130 + 92)} = 0.045\ ms = 45\ \mu s$ | using Equation (1) |

## 22.2 Connecting a P8xC592 to a bus line (physical layer)

### 22.2.1 ON-CHIP TRANSCEIVER

The P8xC592 features an on-chip differential transceiver including output driver and input comparator both being configurable (see Fig.36). Therefore it supports many types of common transmission media such as:

- Single-wire bus line
- Two-wire bus line (differential)
- Optical cable bus line.

The P8xC592 can directly drive a differential bus line. An example is given in Fig.37 for a bus line having a characteristic impedance of 120 Ω. Direct interfacing to the bus line is well suited for applications with limited requirements concerning electromagnetic susceptibility, wiring failure tolerance and protection against transients.

### 22.2.2 TRANSCEIVER FOR IN-VEHICLE COMMUNICATION

Fig.38 shows a versatile transceiver implementation designed for automotive applications. It features a bit rate of up to 1 Mbit/s and dissipates low power during standby (1.4 mA). Thus it is suitable also for applications requiring a Sleep mode function with system activation via the bus line. The transceiver provides and extended common mode range for high electromagnetic susceptibility performance.

Two external driver transistors amplify the output current to 35 mA typically and provide protection against overvoltage conditions on the bus line (e.g. due to an accidental short-circuit between a bus wire and battery voltage). The serial diodes prevent in combination with the transistors the bus from being blocked in case of a bus not powered. More than 32 nodes may be connected to the bus line.

### 22.2.3 DETECTION AND HANDLING OF BUS WIRING FAILURES

Using the P8xC592 a superior wiring failure tolerance and detection performance can be achieved. This requires both bus lines to be mutually decoupled as shown in Fig.39. Each bus wire is based separately to a reference voltage of $\frac{1}{2}AV_{DD}$.

The diodes suppress reverse current in case of a termination circuit being not properly powered or a bus line being short i.e. to a voltage higher than 5 V. Applying this bus termination circuit the following wiring failures on the bus are detectable and can be handled:

- Interruption of one bus wire at any location.
- Short-circuit of one bus wire to ground or battery voltage.
- Short-circuit between the bus wires.

A bus failure can be detected e.g. by a drop out of a status message, regularly being transmitted on the bus. If a bus wire is corrupted the following actions have to be taken:

- Switch the corresponding comparator input over to a reference voltage of $\frac{1}{2}AV_{DD}$.
- Disable the corresponding output driver stage.

As a consequence communication will continue on that bus wire not being corrupted. The required reference voltage and the switches for the comparator inputs are provided on-chip. An output driver stage can be disabled by reconfiguration of the on-chip output driver (reprogramming of the Output Control Register of the P8xC592; see Section 13.5.11, Table 51). To find out which of the bus wires is corrupted a heuristic method is applied.

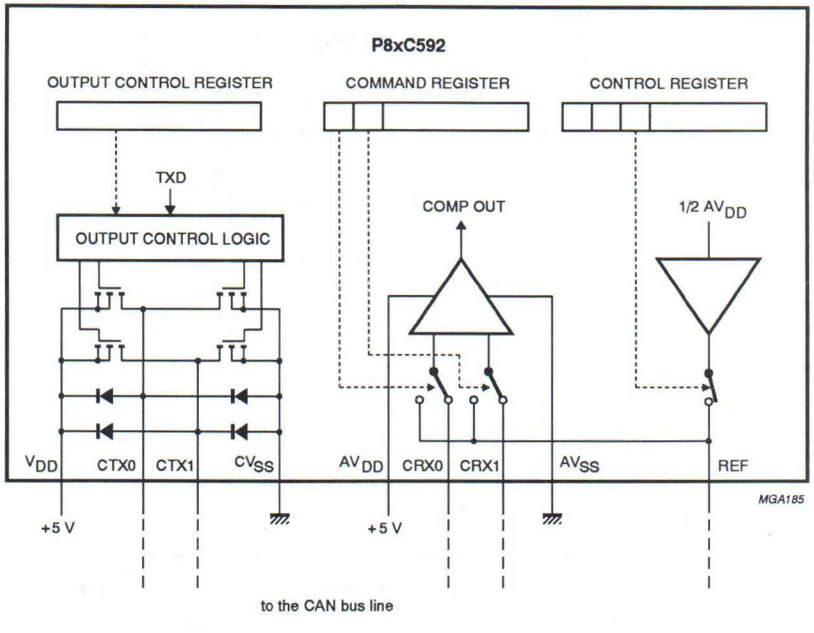



Fig.36 Structure of on-chip CAN-Transceiver.

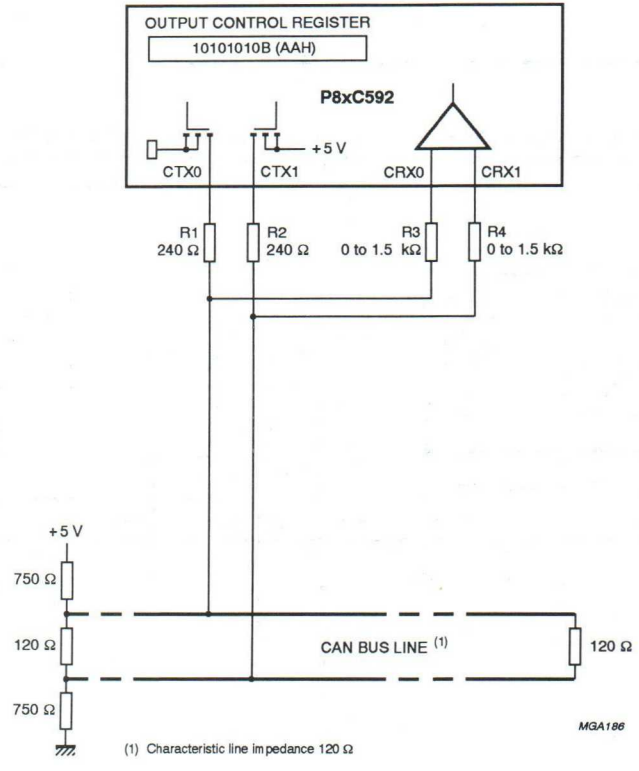



Fig.37 Direct interface to a two-wire differential bus.

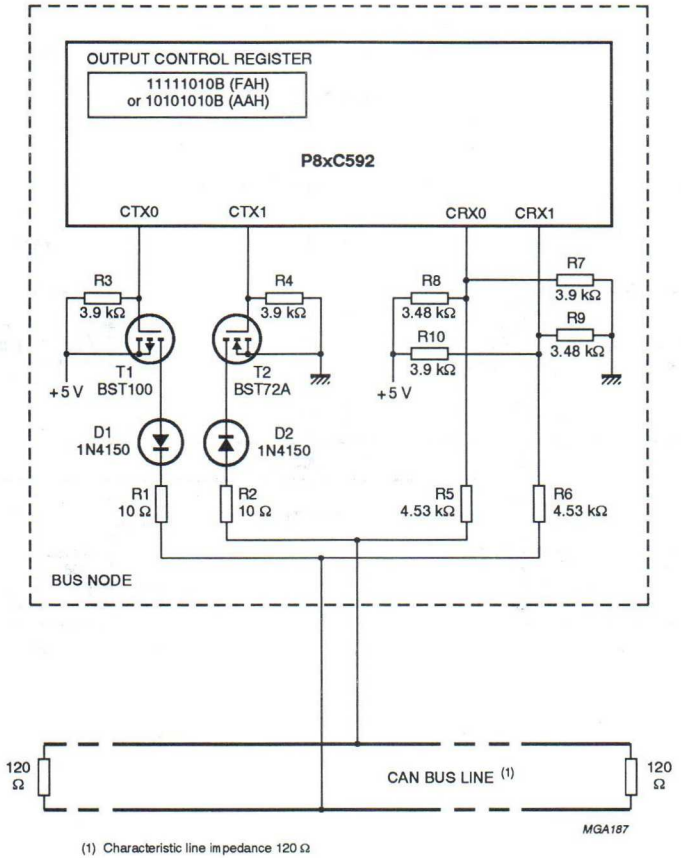



Fig.38 In-vehicle Transceiver.

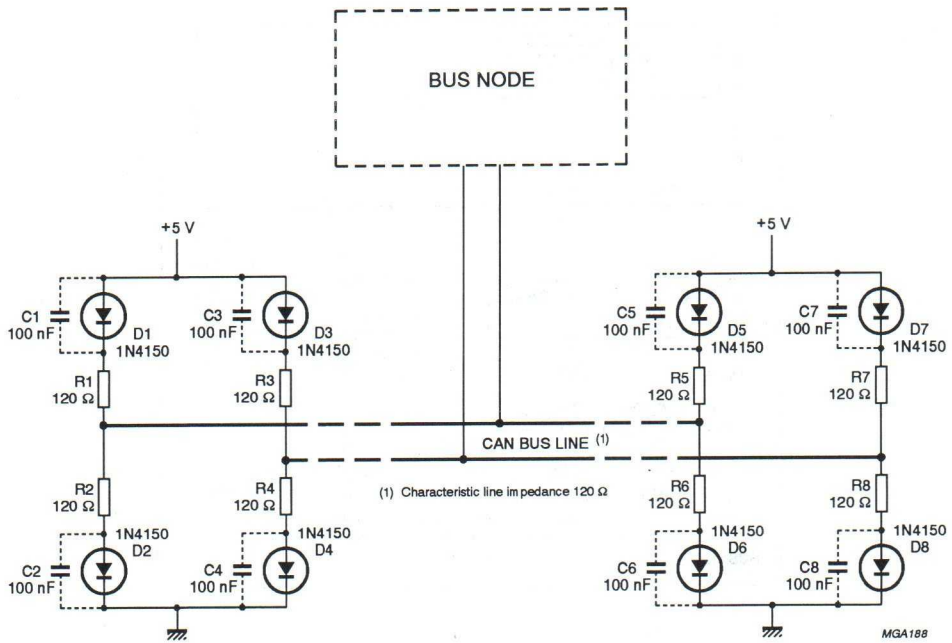



Fig.39 Bus termination with decoupled wires.

### 22.2.4 Connection to an Optical Bus Line

Using an optical medium provides the following advantages:

- Bus nodes are galvanically decoupled.
- Optical cable features very high noise immunity.
- No noise emission by the bus cable.

An example for an interface to an optical connector is given in Fig.40. In most cases a transistor is required to amplify the TX-output current.

Thus more optical power is provided to compensate for losses in the optical connectors and the optical star. The P8xC592 features an on-chip $\frac{1}{2}AV_{DD}$ reference voltage output so only a capacitor is required for the receiver part. Two optical fibres are used to connect the bus nodes. The TX-fibre transfers the output signal of the CAN-controller to the optical star. The optical star transfers the TX-fibre input signal over to all the RX-fibres. The RX-fibres transfer the resulting optical signal over to the receivers of all the bus nodes.

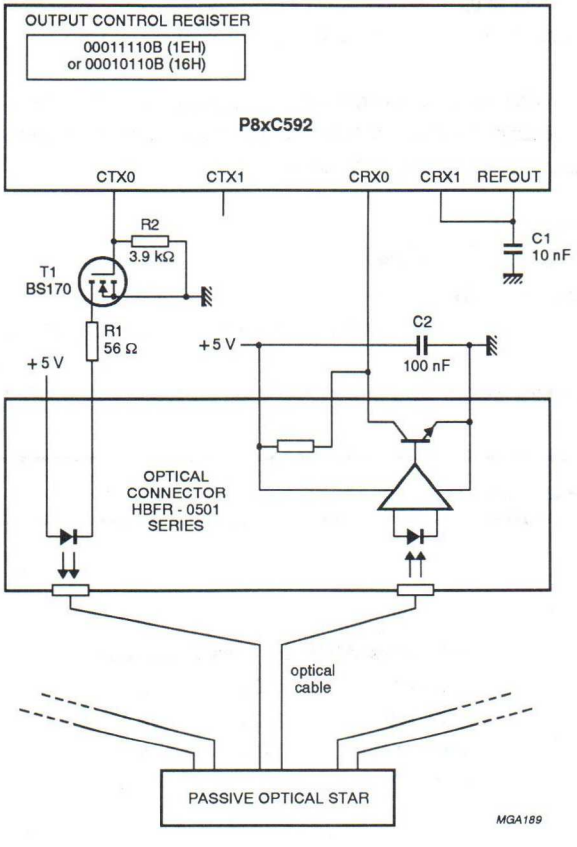



Fig.40 Optical Transceiver.

22.2.5 P8xC592 CAN INTERRUPT HANDLER SOFTWARE EXAMPLE (INCLUDING FAST DMA TRANSFER).

**MCS-51 MACRO ASSEMBLER P8xC592 CAN interrupt-handler**

```
LOC  OBJ    LINE  SOURCE

             1    $TITLE (8xC592 CAN interrupt-handler)
00A0         2    $NOSYMBOLS NOPAGING
00A1         3
             4    ;*******************************************************************************
             5    ;
             6    ;Very fast receive-routine for the 8xC592. It:
             7    • is embedded in the interrupt-handler for the CAN-controller,
             8    • uses the DMA-logic and
             9    • handles up to eight different messages
00A2        10    ;(if these have the same leading 8 identifier-bits).
            11    ;
            12    ;To allow for faster receive-routine, it is assumed that all other routines
            13    ;accessing the CAN-controller, disable the interrupt of the CAN-controller
            14    ;(IEN0.5) during their execution.
00A5        15    ;
00A7        16    ;Version:    1.0
            17    ;Date:       12-April-90
            18    ;Author:     Bernhard Reckels
            19    ;at:         Philips Components Application Lab., Hamburg (PCALH)
00A9        20
00AB        21    ;*******************************************************************************
00AD        22
            23    ;*******************************************************************************
            24    ;initial stuff
            25    ;*******************************************************************************
            26
            27    ;equatas
            28
            29              ;addresses of Special Function Registers
00AE        30              CANADR      EQU     0DBH
00AF        31              CANDAT      EQU     0DAH
            32              CANCON      EQU     0D9H
00B0        33              CANSTA      EQU     0D8H
            34
```

```
LOC   OBJ      LINE  SOURCE
                 35          ;commands for the CAN-controller / DMA logic
                 36          CAN_REF_REL     EQU     00000100B   ;Release Receive Buffer
00A0             37          CAN_RX_DMA      EQU     80H + 22    ;Rx DMA-transfer
00A1             38
                 39          ; addresses of CAN-controller internal registers
                 40          CAN_REF         EQU     20          ;1st address of Rx-buffer
                 41
                 42          ; masks
                 43          INT_FLAG_MASK   EQU     00011111B   ;all CAN's interrupt-flags
                 44          ID2_0_MASK      EQU     11100000B   ;only ID.2 ... ID.0 bits
00A2             45    ; jump-address for a CAN-controller interrupt
                 46
                 47
                 48    CSEG at 2BH
         020080  49          LJMP            CAN_INT_HANDLER     ; CAN's interrupt-vector
00A5             50
00A7             51    ; data storage
                 52
                 53    DSEG at 20H
                 54          CAN_INT_IMAGE:  DS      1
00A9             55
00AB             56    BSEG at 00H
00AD             57          CAN_INT_RX:     DBIT    1           ; = CAN_INT_IMAGE.0
                 58          CAN_INT_TX:     DBIT    1           ; = CAN_INT_IMAGE.1
                 59          CAN_INT_KR:     DBIT    1           ; = CAN_INT_IMAGE.2
                 60          CAN_INT_OV:     DBIT    1           ; = CAN_INT_IMAGE.3
                 61          CAN_INT_WK:     DBIT    1           ; = CAN_INT_IMAGE.4
                 62
                 63    ;*******************************************************************************************
                 64    ;CAN-controller interrupt-handler
00AE             65    ;
00AF             66    ;Only the receive-interrupt is coded.
                 67    ;
00B0             68    ;*******************************************************************************************
                 69
                 70    CSEG at 080H
                 71
```

```
LOC   OBJ      LINE  SOURCE
00A0           72    CAN_INT_HANDLER:
00A1           73
               74            ; first save used registers
      C0D0     75            PUSH        PSW
      C0E0     76            PUSH        ACC
               77
               78            ; store the CAN-controller's Interrupt Register contents
               79            ; (here: at a bit-addressable location).
00A2           80            ; This is necessary because after reading the Interrupt Register
               81            ; its contents is cleared, but – on the other hand – several flags
               82            ; may be set in coincidence.
      E5D9     83            MOV         A, CANON
      541F     84            ANL         A, #INT_FLAG_MASK      ; only interrupt-flags
00A5  F520     85            MOV         CAN_INT_IMAGE, A
00A7           86
               87
               88    ;dispatcher----------------------------------------------------------------------------------
               89    INT_TEST0:
00A9  100000   90            JBC     CAN_INT_RX,CAN_RX_SERV     ;receive-interrupt?
00AB           91
00AD           92    INT_TEST1:
               93            ; here the dispatcher has to be completed according
               94            ; to the application-specific requirements
               95            ; ...
               96            ; ...
               97    ; end of dispatcher--------------------------------------------------------------------------
               98
               99    ;Rx-serve------------------------------------------------------------------------------------
00AE           100   ; copy message (Data-Field only) from CAN- to CPU memory
00AF           101
               102   CAN_RX_SERVE
00B0           103           ; read 2nd Descriptor-Byte from the Rx-Buffer (address 21)
      75DB15   104           MOV         CANADR, #CAN_REF + 1
      E5DA     105           MOV         A, CANDAT
               106
```

```
LOC   OBJ      LINE  SOURCE
00A0           107       ; determine the destination address in data-memory for the
00A1           108       ; message's Data-Field
               109   54E0  ANL   A, #ID2_0_MASK   ; use ID.2 ... ID.0 only
               110   C4    SWAP  A
               111   03    RR    A               ; A = 4*ID.2 + 2*ID.1 + ID.0
               112       ; this value is used as an index for an array of 8 bytes
               113       ; containing the destination-addresses for the 8 different
               114       ; messages. Note, that the #RX_ARRAY_OFFSET is due to the
00A2           115       ; program counter-relative access to the array.
               116   2415  ADD   A, #RX_ARRAY_START - RX_ARRAY_OFFSET
               117   83    MOVC  A, @A + PC
               118       RX_ARRAY_OFFSET:
               119
00A5           120       ; if a message passes the acceptance-filter of the CAN
00A7           121       ; Controller, but the CPU doesn't need it, the array
               122       ; entry's value may be set to zero indicating this.
               123       ; The following <jz> instruction cares for this.
               124   6007  JZ    CAN_RX_READY
00A9           125
00AB           126       ; now copy the Data-Field (only) from CAN- to CPU memory
00AD           127       ; with the aid of the DMA-logic. Note, that a TX-DMA is
               128       ; performed when writing 8AH (DMA + address 10) into CANADR
               129       ; and a RX-DMA is performed when writing 94H (DMA + address 20)
               130       ; ... 9DH (DMA + address 29) into CANADR. Here address 22 is
               131       ; used to copy just the Data-Field.
               132   F5D8  MOV   CANSTA, A         ; data-memory address
               133   75DB96 MOV  CANADR, #CAN_RX_DMA ; starts RX-DMA at address 22
               134
00AE           135       ; the DMA-transfer is done in at maximum 2 instruction cycles.
00AF           136       ; During the transfer, neither the data-memory (RAM) nor one
               137       ; of the SFRs CANADR, CANDAT, CANCON and
00B0           138       ; CANSTA may be accessed by the CPU.
               139       ; For simplicity, two NOPs are used here.
               140   00    NOP
               141   00    NOP
00A0           142
```

```
LOC  OBJ     LINE  SOURCE
00A1         143           ; after reading the Rx-Buffer it must be released back to
             144           ; the CAN-controller. In coincidence, the Clear Overrun bit
             145           ; (CANCON.3) may be set, regardless of an existing or
             146           ; non-existing data overrun.
             147     CAN_RX_READY:
     75D904  148           MOV       CANCON, #CAN_RBF_REL
             149
00A2         150           ; if no other interrupt-flag is set, the interrupt-handler
             151           ; for the CAN-controller can be left. Otherwise further
             152           ; services are required.
     E520    153           MOV       A, CAN_INT_IMAGE
     70E4    154           JNZ       INT_TEST1
00A5         155
00A7         156           ; no other service is required, so the interrupt-handler
             157           ; is left.
     D0E0    158           POP       ACC
     D0D0    159           POP       PSW
00A9 32      160           RETI
00AB         161     ; end of Rx-serve----------------------------------------------------------------------------
00AD         162
             163           ; here the array follows containing 8 destination-addresses
             164           ; for up to 8 different messages to be received. The values
             165           ; are fully application-specific (the values below show an
             166           ; example only).
             167     RX_ARRAY_START:
     E0      168           DB        0E0H      ; Rx-message #0
     00      169           DB        000H      ; this message is not used
00AE         170           ; ...
00AF FA      171           DB        0FAH      ; RX-message #7, containing 6 data bytes
             172
00B0         173           END

REGISTER BANK(S) USED: 0

ASSEMBLY COMPLETE, NO ERRORS FOUND
```

# 8-bit microcontroller with on-chip CAN P8xCE598

## 1 FEATURES

- 80C51 central processing unit (CPU)
- 32 kbytes on-chip ROM, externally expandible to 64 kbytes
- 2 x 256 bytes on-chip RAM, externally expandible to 64 kbytes
- Two standard 16-bit timers/counters
- One additional 16-bit timer/counter coupled to four capture and three compare registers
- 10-bit ADC with 8 multiplexed analog inputs
- Two 8-bit resolution Pulse Width Modulated outputs
- 15 interrupt sources with 2 priority levels (2 to 6 external interrupt sources possible)
- Five 8-bit I/O ports, plus one 8-bit input port shared with analog inputs
- CAN-controller (CAN = Controller Area Network) with DMA data transfer facility to internal RAM
- 1 Mbit/s CAN-controller with bus failure management facility
- $\frac{1}{2}AV_{DD}$ reference voltage
- Full-duplex UART compatible with the standard 80C51
- On-chip Watchdog Timer (WDT)
- 1.2 to 16 MHz clock frequency
- Improved Electromagnetic Compatibility (EMC).

## 2 GENERAL DESCRIPTION

The P8xCE598 is a single-chip 8-bit high-performance microcontroller with on-chip CAN-controller, derived from the 80C51 microcontroller family.
It uses the powerful 80C51 instruction set.
Figure 1 shows a block diagram of the P8xCE598.

The P8xCE598 is manufactured in an advanced CMOS process, and is designed for use in automotive and general industrial applications. In addition to the 80C51 standard features, the device provides a number of dedicated hardware functions for these applications.

Two versions of the P8xCE598 will be offered:

- P80CE598 (without ROM)
- P83CE598 (with ROM)

Hereafter these versions will be referred to as P8xCE598.

The temperature range includes (max. $f_{CLK}$ = 16 MHz):

- –40 to +85 °C version, for general applications
- –40 to +125 °C version for automotive applications.

The P8xCE598 combines the functions of P8XC552 (microcontroller) and the PCA82C200 (Philips CAN-controller) with the following enhanced features:

- 32 kbytes Program Memory
- 2 x 256 bytes Data Memory
- DMA between CAN Transmit/Receive Buffer and internal RAM.

The main differences to the P8xC552 microcontroller are:

- 32 kbytes programmable ROM (P8xC552 has 8 kbytes)
- Additional 256 bytes RAM
- A CAN-controller instead of the $I^2C$-serial interface.

### 2.1 Electromagnetic Compatibility (EMC)

Primary attention is paid to the reduction of electromagnetic emission of the microcontroller P8xCE598. The following features reduce the electromagnetic emission and additionally improve the electromagnetic susceptibility:

- One analog part power supply pin ($AV_{DD}$) and one analog part ground pin ($AV_{SS}$), placed as a pair of pins on one side of the package (see Fig.3), providing power supply (+5V) and ground for ADC, CAN receiver and reference voltage.
- Four digital part supply voltage pins ($V_{DD1}$ to $V_{DD4}$) and four digital part ground pins ($V_{SS1}$ to $V_{SS4}$) are provided on the package. These pins, one $V_{DD}$ and one $V_{SS}$ as a pair of pins are placed on each of the four sides of the package to provide:
  - $V_{DD1}/V_{SS1}$ for internal logic (CPU, Timers/counters, Memory, CAN, UART, ADC)
  - $V_{DD2}/V_{SS2}$ for Port 1, Port 3 and Port 4, and $\overline{PWM0}$ and $\overline{PWM1}$ outputs
  - $V_{DD3}/V_{SS3}$ for the on-chip oscillator
  - $V_{DD4}/V_{SS4}$ for the Port 0, Port 2, ALE output and $\overline{PSEN}$ output.
- External capacitors should be connected across associated $V_{DDx}$ and $V_{SSx}$ pins (i.e. $V_{DD1}$ and $V_{SS1}$). Lead length should be as short as possible. Ceramic chip capacitors are recommended (100 nF).
- One CAN supply voltage pin ($CV_{DD}$) and one CAN ground pin ($CV_{SS}$) as a pair of pins placed on one side of the package providing (digital part) power supply (+5V) and ground for the CAN transmitter outputs.
- Internal decoupling capacitance improves the EMC radiation behaviour and the EMC immunity.

### 2.2 Recommendation on ALE

For application that require no external memory or temporarily no external memory: the ALE output signal (pulses at a frequency of $\frac{1}{6} f_{OSC}$) can be disabled under software control (bit 5 in PCON SFR: **'RFI'**); if disabled, no ALE pulse will occur. ALE pin will be pulled down internally, switching an external address latch to a quiet state. The MOVX instruction will still toggle ALE as a normal MOVX.

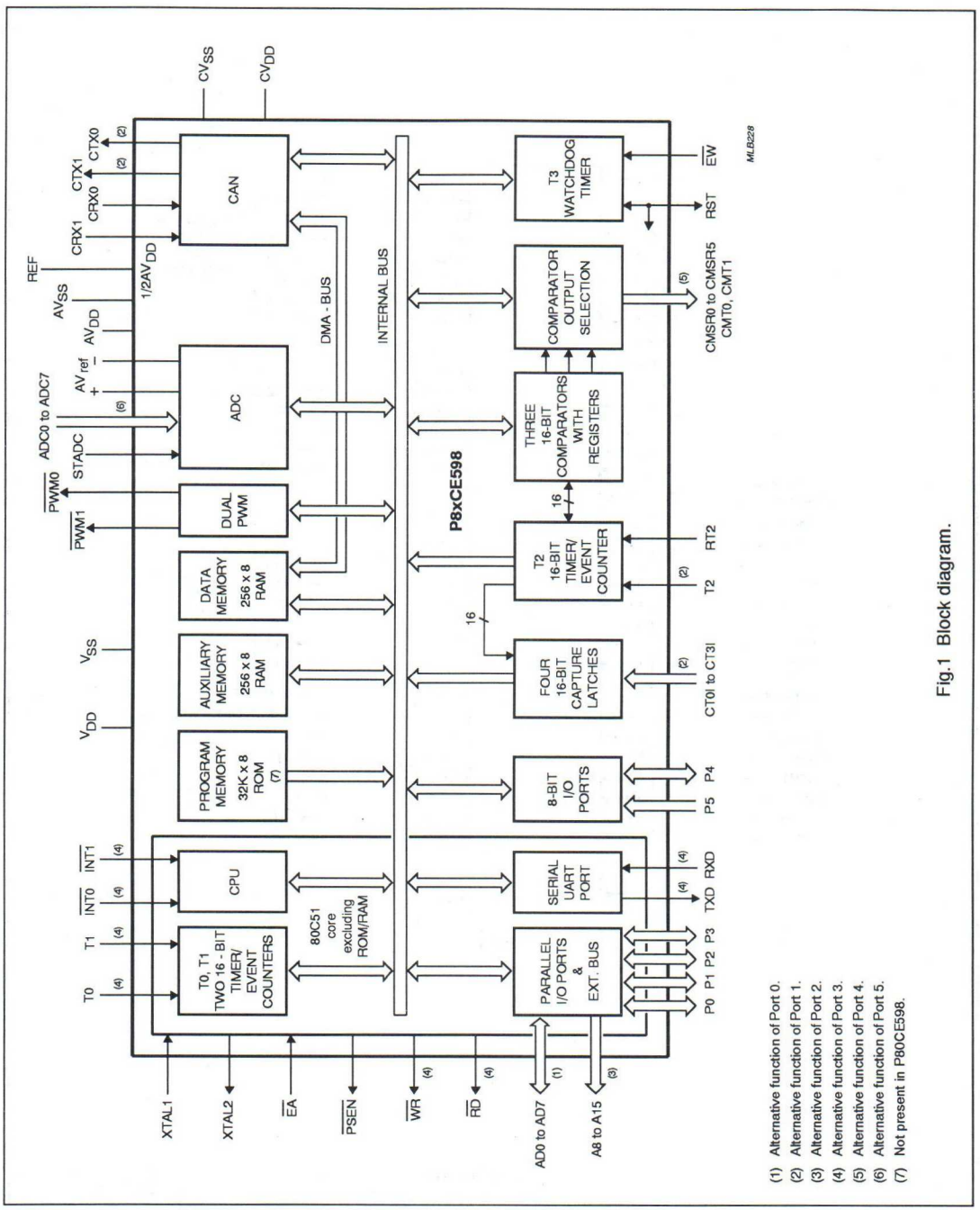
ALE will retain its normal HIGH value during Idle mode and a LOW value during Power-down mode while in the **'RFI** reduction mode'.

Additionally during internal access ($\overline{EA}$ = 1) ALE will toggle normally when the address exceeds the internal Program Memory size. During external access ($\overline{EA}$ = 0) ALE will always toggle normally, whether the flag **'RFI'** is set or not.

## 3 ORDERING INFORMATION

| TYPE NUMBER | PACKAGE | | | TEMPERATURE RANGE (°C) | FREQ. (MHz) |
|---|---|---|---|---|---|
| | NAME | DESCRIPTION | VERSION | | |
| **Without ROM** | | | | | |
| P80CE598FFB | QFP80 | plastic quad flat package; 80 leads (lead length 1.95 mm); body 14 × 20 × 2.7 mm; high stand-off height | SOT318-1 | –40 to +85 | 1.2 to 16 |
| P80CE598FHB | | | | –40 to +125 | |
| **With ROM** | | | | | |
| P83CE598FFB | QFP80 | plastic quad flat package; 80 leads (lead length 1.95 mm); body 14 × 20 × 2.7 mm; high stand-off height | SOT318-1 | –40 to +85 | 1.2 to 16 |
| P83CE598FHB | | | | –40 to +125 | |

## 4 BLOCK DIAGRAM




(1) Alternative function of Port 0.
(2) Alternative function of Port 1.
(3) Alternative function of Port 2.
(4) Alternative function of Port 3.
(5) Alternative function of Port 4.
(6) Alternative function of Port 5.
(7) Not present in P80CE598.

Fig.1 Block diagram.

## 5 PINNING

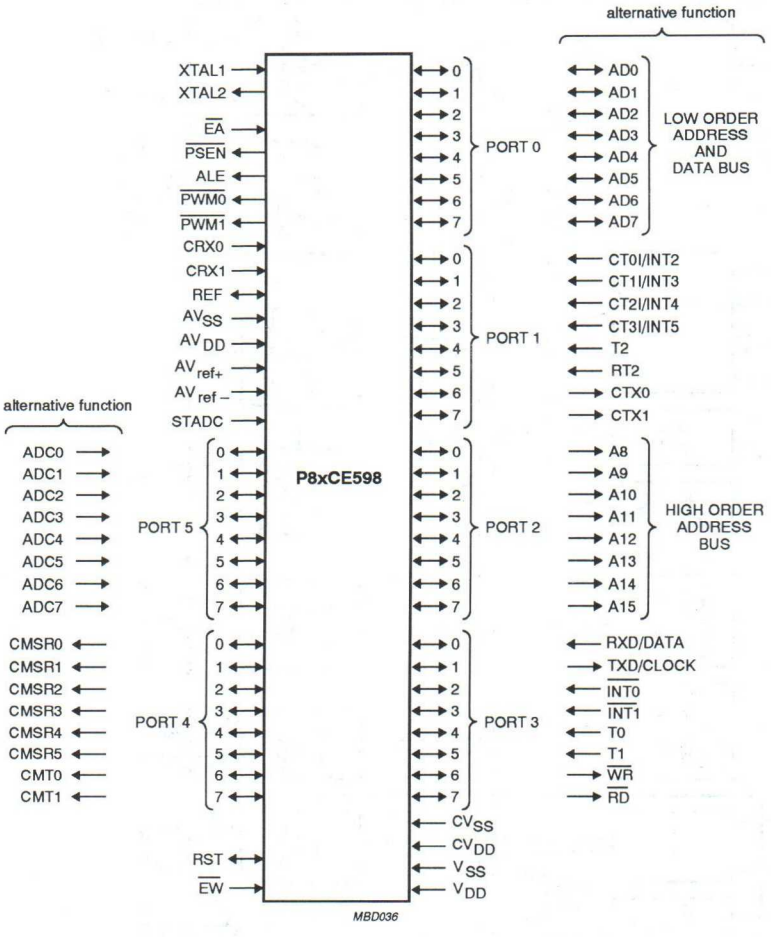



Fig.2 Pin functions.

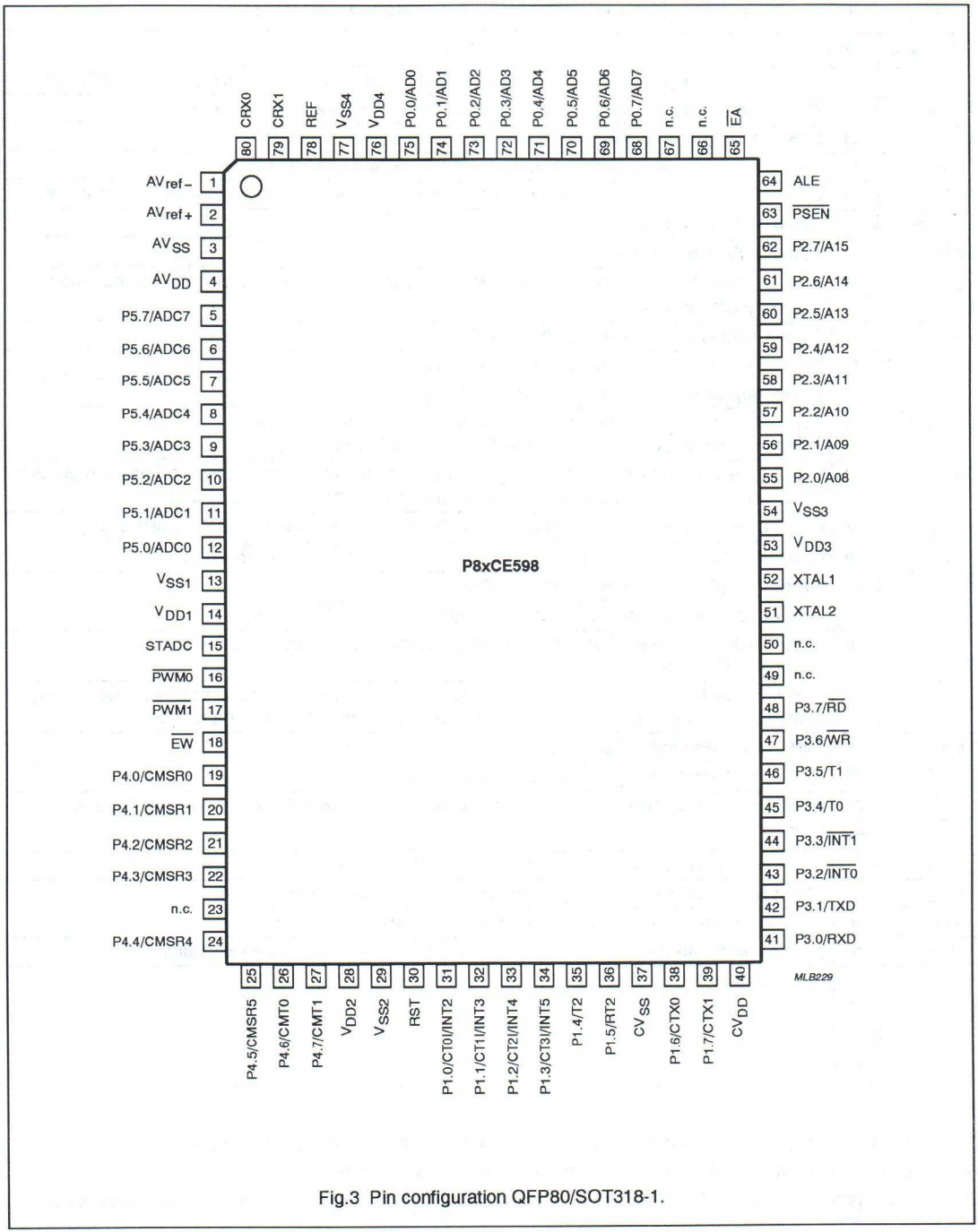



Fig.3 Pin configuration QFP80/SOT318-1.

**Table 1** Pin description for **single function** pins (SOT318-1 and SOT351-1; see note 1)

| SYMBOL | PIN | DESCRIPTION |
|---|---|---|
| $V_{DD1}$ | 14 | **Power supply, digital part:** for internal logic (CPU, Timers/counters, Memory, CAN, UART, ADC). |
| $V_{DD2}$ | 28 | **Power supply, digital part:** for Port 1, Port 3, Port 4, $\overline{PWM0}$ and $\overline{PWM1}$ outputs. |
| $V_{DD3}$ | 53 | **Power supply, digital part:** for the on-chip oscillator. |
| $V_{DD4}$ | 76 | **Power supply, digital part:** for Port 0, Port 2, ALE output and $\overline{PSEN}$ output. |
| STADC | 15 | **Start ADC operation.** Input starting analog-to-digital conversion (note 2). This pin must not float. |
| $\overline{PWM0}$ | 16 | **Pulse width modulation** output 0. |
| $\overline{PMW1}$ | 17 | **Pulse width modulation** output 1. |
| $\overline{EW}$ | 18 | **Enable Watchdog Timer (WDT):** enable for T3 Watchdog Timer and disable Power-down mode. This pin must not float. |
| RST | 30 | **Reset:** input to reset the P8xCE598 (note 3). |
| $CV_{SS}$ | 37 | **Ground** potential for the CAN transmitter outputs. |
| $CV_{DD}$ | 40 | **Power supply** (+5V) for the CAN transmitter outputs. |
| XTAL2 | 51 | **Crystal pin 2:** output of the inverting amplifier that forms the oscillator. When an external clock oscillator is used this pin is left open-circuit. |
| XTAL1 | 52 | **Crystal pin 1:** input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external clock oscillator signal, when an external oscillator is used. |
| $V_{SS1}$ | 13 | **Ground, digital part:** for internal logic (CPU, Timers/Counters, Memory, CAN, UART, ADC). |
| $V_{SS2}$ | 29 | **Ground, digital part:** for Port 1, Port 3 and Port 4, and $\overline{PWM0}$ and $\overline{PWM1}$ outputs. |
| $V_{SS3}$ | 54 | **Ground, digital part:** for the on-chip oscillator. |
| $V_{SS4}$ | 77 | **Ground, digital part:** for the Port 0, Port 2, ALE output and $\overline{PSEN}$ output. |
| $\overline{PSEN}$ | 63 | **Program Store Enable:** Read strobe to external Program Memory (active LOW). Drive: 8 × LSTTL inputs. |
| ALE | 64 | **Address Latch Enable:** latches the Low-byte of the address during accesses to external memory (note 4). Drive: 8 × LSTTL inputs; handles CMOS inputs without an external pull-up. |
| $\overline{EA}$ | 65 | **External Access input.** See note 5. |
| REF | 78 | **$\frac{1}{2}AV_{DD}$ reference voltage** output respectively input (note 6). |
| CRX1 | 79 | **Inputs from the CAN-bus line** to the differential input comparator of the on-chip CAN-controller (note 7). |
| CRX0 | 80 | |
| $AV_{REF-}$ | 1 | **Low-end of ADC** (analog-to-digital conversion) reference resistor. |
| $AV_{REF+}$ | 2 | **High-end of ADC** (analog-to-digital conversion) reference resistor (note 8). |
| $AV_{SS}$ | 3 | **Ground, analog part.** For ADC, CAN receiver and reference voltage. |
| $AV_{DD}$ | 4 | **Power supply, analog part** (+5 V). For ADC, CAN receiver and reference voltage. |
| n.c. | 23, 49, 50, 66, 67 | **No connection.** |

**Notes**

1. To avoid a 'latch up' effect at power-on: $V_{SS} - 0.5\ V <$ 'voltage on any pin at any time' $< V_{DD} + 0.5\ V$.
2. Triggered by a rising edge. ADC operation can also be started by software.
3. RST also provides a reset pulse as output when timer T3 overflows or after a CAN wake-up from Power-down.

4. ALE is activated every six oscillator periods. During an external data memory access one ALE pulse is skipped.
5. See Section 7.1, Table 3 for $\overline{EA}$ operation. For P83CE598 microcontrollers specified with the option 'ROM-code protection', the $\overline{EA}$ pin is latched during reset and is 'don't care' after reset, regardless of whether the ROM-code protection is selected or not.
6. Pin 78, REF:
   a) Selection of input respectively output dependent of CAN Control Register bit 5 (CR.5; see Section 13.5.3 Table 32).
   b) If the internal reference is used, then REF should be connected to $AV_{SS}$ via a capacitor with a value of ≥ 10 nF.
   c) After an external reset (RST = HIGH) the internal $\frac{1}{2}AV_{DD}$ source is activated and, REF is a reference output.
   d) If the CAN-controller is in the reset state, e.g. after an external reset, then the $\frac{1}{2}AV_{DD}$ source is switched off during Power-down mode.
7. CAN Bus line:
   a) CRX0 level > CRX1 level is interpreted as a logic 1 (recessive).
   b) CRX0 level < CRX1 level is interpreted as a logic 0 (dominant).
8. The level of $AV_{REF+}$ must be higher than that of $AV_{REF-}$.

**Table 2** Pin description for pins with **alternative functions** (SOT318-2 and SOT351-1; see note 1)

| SYMBOL | | PIN | DESCRIPTION |
|---|---|---|---|
| **DEFAULT** | **ALTERNATIVE** | | |
| **Port 4** | | | |
| P4.0 to P4.7 | | 19 to 22, 24 to 27 | **8-bit quasi-bidirectional I/O port.** |
| | CMSR0 | 19 | **Compare and Set/Reset outputs** for Timer T2. |
| | CMSR1 | 20 | |
| | CMSR2 | 21 | |
| | CMSR3 | 22 | |
| | CMSR4 | 24 | |
| | CMSR5 | 25 | |
| | CMT0 | 26 | **Compare and toggle outputs** for Timer T2. |
| | CMT1 | 27 | |
| **Port 1** | | | |
| P1.0 to P1.7 | | 31 to 36, 38 to 39 | **8-bit quasi-bidirectional I/O port.** |
| | CT0I/INT2 | 31 | **Capture timer inputs** for Timer T2, or **External interrupt inputs** 2 to 5. |
| | CT1I/INT3 | 32 | |
| | CT2I/INT4 | 33 | |
| | CT3I/INT5 | 34 | |
| | T2 | 35 | **T2 event input** (rising edge triggered). |
| | RT2 | 36 | **T2 timer reset input** (rising edge triggered). |
| | CTX0 | 38 | **CAN transmitter** output 0 (note 2). |
| | CTX1 | 39 | **CAN transmitter** output 1 (note 2). |

| SYMBOL | | PIN | DESCRIPTION |
|---|---|---|---|
| **DEFAULT** | **ALTERNATIVE** | | |
| **Port 3** | | | |
| P3.0 to P3.7 | | 41 to 48 | **8-bit quasi-bidirectional I/O port.** |
| | RXD | 41 | **Serial Input Port.** |
| | TXD | 42 | **Serial Output Port.** |
| | $\overline{\text{INT0}}$ | 43 | **External interrupt input 0.** |
| | $\overline{\text{INT1}}$ | 44 | **External interrupt input 1.** |
| | T0 | 45 | **Timer 0 external input.** |
| | T1 | 46 | **Timer 1 external input.** |
| | $\overline{\text{WR}}$ | 47 | **External Data Memory Write strobe.** |
| | $\overline{\text{RD}}$ | 48 | **External Data Memory Read strobe.** |
| **Port 2 (Sink/source: 1 × TTL = 4 × LSTTL inputs)** | | | |
| P2.0 to P2.7 | | 55 to 62 | **8-bit quasi-bidirectional I/O port.** |
| | A08 to A15 | | **High-order address byte for external memory.** |
| **Port 0 (Sink/source: 8 × LSTTL inputs)** | | | |
| P0.7 to P0.0 | | 68 to 75 | **8-bit open drain bidirectional I/O port.** |
| | AD7 to AD0 | | **Multiplexed Low-order address and Data bus for external memory.** |
| **Port 5** | | | |
| P5.7 to P5.0 | | 5 to 12 | **8-bit input port.** |
| | ADC7 to ADC0 | | **8 input channels to ADC.** |

**Notes**

1. To avoid a 'latch up' effect at power-on: $V_{SS} - 0.5\ V <$ 'voltage on any pin at any time' $< V_{DD} + 0.5\ V$.
2. If the CAN-controller is in the reset state (e.g. after a power-up reset; CAN Control Register bit CR.0; see Section 13.5.3 Table 32, the CAN transmitter outputs are floating and the pins P1.6 and P1.7 can be used as open-drain port pins. After a power-up reset the port data is HIGH, leaving the pins P1.6 and P1.7 floating.

## 6 FUNCTIONAL DESCRIPTION

The P8xCE598 functions will be described as shown in the following overview:

- Memory organization
- I/O Port structure
- Pulse Width Modulated outputs
- Analog-to-Digital Converter
- Timers/Counters
- Serial I/O Ports
- Interrupt system
- Power reduction modes
- Oscillator circuitry
- Reset circuitry
- Instruction Set
- EMC (see Section 2.1).

## 7 MEMORY ORGANIZATION

The Central Processing Unit (CPU) manipulates operands in three memory spaces (see Fig.4) as follows:

- 32 kbytes internal, resp. 64 kbytes external Program Memory
- 512 bytes internal Data Memory MAIN- and AUXILIARY RAM.
- Up to 64 kbytes external Data Memory (with 256 bytes residing in the internal AUXILIARY RAM).

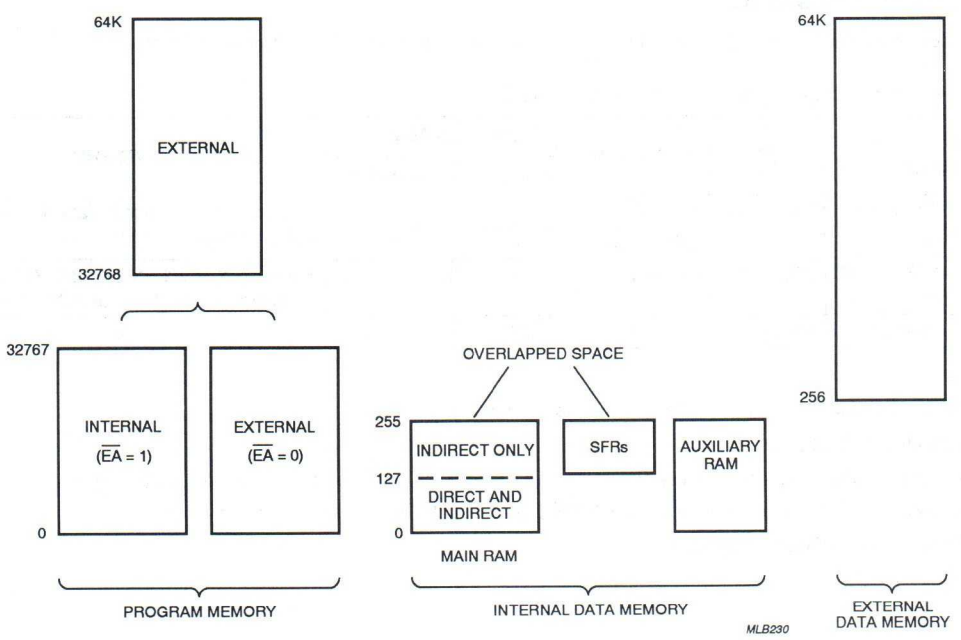



Fig.4 Memory map.

### 7.1 Program Memory

The Program Memory of the P8xCE598 consists of 32 kbytes ROM on-chip, externally expandible up to 64 kbytes.

**Table 3** Instruction fetch controlled by $\overline{EA}$

| PIN $\overline{EA}$ (note 1) | | INSTRUCTIONS FETCHED FROM: | ADDRESS LOCATION |
|---|---|---|---|
| DURING RESET LATCHED TO: | AFTER RESET | | |
| H | – | internal Program Memory (note 2) | 0000H → 7FFFH |
| H | – | external Program Memory | 8000H → FFFFH |
| L | – | | 0000H → FFFFH |
| – | 'don't care' | – | – |

**Notes**

1. This implementation prevents reading of the internal program code by switching from external Program Memory during a MOVC instruction.
2. By setting a security bit the internal Program Memory content is protected, which means it cannot be read out. If the security bit has been set to LOW there are no restrictions for the MOVC instruction.

### 7.2 Internal Data Memory

The internal Data Memory is physically built-up and accessible as shown in Table 4 (see Fig.5).

**Table 4** Internal Data Memory size and address mode

| INTERNAL DATA MEMORY | SIZE | LOCATION | ADDRESS MODE | | POINTERS |
|---|---|---|---|---|---|
| | | | DIRECT | INDIRECT | |
| MAIN RAM (note 1) | 256 bytes | 0 to 127 | X | X | Address pointers are R0 and R1 of the selected register bank. |
| | | 128 to 255 | – | X | |
| AUXILIARY RAM (note 2) | 256 bytes | 0 to 255 | – | X | Address pointers are R0 and R1 of the selected register bank and the DPTR. |
| SFRs (note 3) | 128 bytes | 128 to 255 | X | – | – |

**Notes**

1. MAIN RAM can be addressed directly and indirectly as in the 80C51.
2. AUXILIARY RAM (0 to 255):
   a) Is indirectly addressable in the same way as the external Data Memory with MOVX instructions.
   b) Access will not affect the ports P0, P2, P3.6 and P3.7 during internal program execution.
3. SFRs = Special Function Registers.

**NOTE:** For chapters 7.3 through 18.2, please refer to the 8xC592 datasheet.

## 19 ABSOLUTE MAXIMUM RATINGS

In accordance with the Absolute Maximum Rating System (IEC 134); note 1

| SYMBOL | PARAMETER | MIN. | MAX. | UNIT |
|---|---|---|---|---|
| $V_{DD}$ | voltage on $V_{DD}$ pins | –0.5 | +6.5 | V |
| $V_I$ (note 1) | input voltage on any pin (except CTX0, CTX1, CRX0, CRX1 and $\overline{EA}/V_{PP}$) | –0.5 | $V_{DD}$ + 0.5 | V |
| $V_I$ (note 2) | input voltage on $\overline{EA}/V_{PP}$ to $V_{SS}$ | –0.5 | +13 | V |
| $I_I$, $I_O$ | input/output current on any single I/O pin (except from CTX0 and CTX1) | – | ±10 | mA |
| $I_{OT}$ | sink current of CTX0, CTX1 together | – | 30 | mA |
| | source current of CTX0, CTX1 together | – | –20 | mA |
| $P_{tot}$ | total power dissipation (note 2) | – | 1.0 | W |
| $T_{stg}$ | storage temperature range | –65 | +150 | °C |
| $T_{amb}$ | operating ambient temperature range: | | | |
| | P83CE598 FFB/P80CE598 FFB | –40 | +85 | °C |
| | P83CE598 FHB/P80CE598 FHB | –40 | +125 | °C |

**Notes**

1. The following applies to the Absolute Maximum Ratings:
   a) Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the Chapters "AC characteristics" and "DC characteristics" of this specification is not implied.
   b) This product includes circuitry specifically designed for the protection of its internal devices from the damaging effect of excessive static charge. However, its suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
   c) Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.
2. This value is based on the maximum allowable die temperature and the thermal resistance of the package, not on device power consumption.

## 20 DC CHARACTERISTICS

$V_{DD}$ = 5 V ± 10%; $V_{SS}$ = 0 V; all voltages with respect to $V_{SS}$ unless otherwise specified.
$T_{amb}$ = −40 to +125 °C for the **P8xCE598FFB**; $T_{amb}$ = −40 to +85 °C for the **P8xCE598FHB**.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| **Supply (digital part)** | | | | | |
| $V_{DD}$ | supply voltage | | 4.5 | 5.5 | V |
| $I_{DD}$ | operating supply current | $f_{CLK}$ = 16 MHz; note 1 | – | 50 | mA |
| $I_{DD(ID)}$ | supply current Idle mode | $f_{CLK}$ = 16 MHz; note 2 | – | 15 | mA |
| $I_{DD(IS)}$ | supply current Idle & Sleep mode | $f_{CLK}$ = 16 MHz; note 3 | – | 10 | mA |
| $I_{DD(PD)}$ | supply current Power-down mode:<br>P8xCE598 FHB<br>P8xCE598 FFB | note 4 | <br>–<br>– | <br>150<br>50 | <br>µA<br>µA |
| **Inputs** | | | | | |
| $V_{IL}$ | LOW level input voltage (except $\overline{EA}$, CRX0 and CRX1) | | −0.5 | $0.2V_{DD}$ − 0.1 | V |
| $V_{IL1}$ | LOW level input voltage $\overline{EA}$ | | −0.5 | $0.2V_{DD}$ − 0.3 | V |
| $V_{IH}$ | HIGH level input voltage (except RST, XTAL1, CRX0,CRX1) | | $0.2V_{DD}$ + 0.9 | $V_{DD}$ + 0.5 | V |
| $V_{IH1}$ | HIGH level input voltage (RST and XTAL1) | | $0.7V_{DD}$ | $V_{DD}$ + 0.5 | V |
| $I_{IL}$ | LOW level input current Ports 1, 2, 3 and 4 | $V_I$ = 0.45 V | – | −50 | µA |
| $I_{TL}$ | input current HIGH-to-LOW transition Ports 1, 2, 3 and 4 (except P1.6 and P1.7) | $V_I$ = 2.0 V to 0.45 V | – | −650 | µA |
| $I_{LI1}$ | input leakage current Port 0, $\overline{EA}$, STADC, $\overline{EW}$, P1.6, P1.7 | 0.45 V < $V_I$ < $V_{DD}$ | – | ±10 | µA |
| $I_{LI2}$ | input leakage current Port 5 | 0.45 V < $V_I$ < $V_{DD}$ | – | ±1 | µA |
| **Outputs** | | | | | |
| $V_{OL}$ | LOW level output voltage Ports 1, 2, 3 and 4 (except P1.6 and P1.7) | $I_{OL}$ = 1.6 mA; note 5 | – | 0.45 | V |
| $V_{OL1}$ | LOW level output voltage Port 0, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$, P1.6, P1.7 | $I_{OL}$ = 3.2 mA; note 5 | – | 0.45 | V |
| $V_{OH}$ | HIGH level output voltage Ports 1, 2, 3 and 4 (except P1.6 and P1.7) | $I_{OH}$ = −60 µA<br>$I_{OH}$ = −25 µA<br>$I_{OH}$ = −10 µA | 2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | –<br>–<br>– | V<br>V<br>V |
| $V_{OH1}$ | HIGH level output voltage Port 0 in external bus mode, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$ | $I_{OH}$ = −400 µA<br>$I_{OH}$ = −150 µA<br>$I_{OH}$ = −40 µA; note 6 | 2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | –<br>–<br>– | V<br>V<br>V |

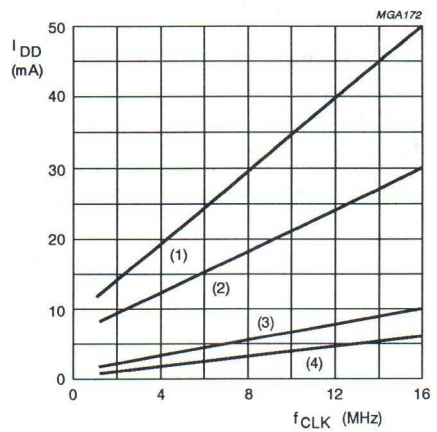
| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| $V_{OH2}$ | HIGH level output voltage RST | $I_{OH}$ = −400 μA | 2.4 | – | V |
| | | $I_{OH}$ = −120 μA | $0.8V_{DD}$ | – | V |
| $R_{RST}$ | RST pull-down resistor | | 50 | 150 | kΩ |
| $C_{I/O}$ | I/O pin capacitance | test frequency = 1 MHz; $T_{amb}$ = 25 °C | – | 10 | pF |
| **Supply (analog part)** | | | | | |
| $AV_{DD}$ | supply voltage | $AV_{DD} = V_{DD} \pm 0.2$ V | 4.5 | 5.5 | V |
| $AI_{DD}$ | supply current operating | Port 5 = $AV_{DD}$; note 1 | – | 2.5 | mA |
| $AI_{DD(ID)}$ | supply current Idle mode | note 2 | – | 2.5 | mA |
| $AI_{DD(IS)}$ | supply current Idle and Sleep mode: | note 3 | | | |
| | P8xCE598 FHB | | – | 400 | μA |
| | P8xCE598 xFx | | – | 350 | μA |
| $AI_{DD(PD)}$ | supply current Power-down mode: | note 4 | | | |
| | P8xCE598 FHB | | – | 400 | μA |
| | P8xCE598 xFx | | – | 350 | μA |
| **Analog inputs** | | | | | |
| $AV_{IN}$ | analog input voltage | | $AV_{SS}$ − 0.2 | $AV_{DD}$ + 0.2 | V |
| $AV_{REF-}$ | reference voltage | | $AV_{SS}$ − 0.2 | – | V |
| $AV_{REF+}$ | | | – | $AV_{DD}$ + 0.2 | V |
| $R_{REF}$ | resistance between $AV_{REF+}$ and $AV_{REF-}$ | | 10 | 50 | kΩ |
| $C_{IA}$ | analog input capacitance | | – | 15 | pF |
| $t_{ADS}$ | sampling time | note 7 | – | $8t_{CY}$ | μs |
| $t_{ADC}$ | conversion time (including sample time) | note 7 | – | $50t_{CY}$ | μs |
| $DL_e$ | differential non-linearity | notes 8, 9 and 10 | – | ±1 | LSB |
| $IL_e$ | integral non-linearity | notes 8 and 11 | – | ±2 | LSB |
| $OS_e$ | offset error | notes 8 and 12 | – | ±2 | LSB |
| $G_e$ | gain error | notes 8 and 13 | – | ±0.4 | % |
| $A_e$ | absolute voltage error | notes 8 and 14 | – | ±3 | LSB |
| $M_{ctc}$ | channel-to-channel matching | | – | ±1 | LSB |
| $C_t$ | crosstalk between P5 inputs | 0 to 100 kHz | – | −60 | dB |
| **CAN input comparator (CRX0, CRX1)** | | | | | |
| $V_{DIF}$ | differential input voltage (note 15) | $AV_{DD}$ = 5 V ± 5%; 1.4 V < $V_I$ < $AV_{DD}$ − 1.4 V | ±32 | – | mV |
| $V_{HYST}$ | hysteresis voltage (note 15) | | 8 | 30 | mV |
| $I_I$ | input current | | – | ±400 | nA |

| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| **CAN output driver ($V_{DD}$ = 5 V ± 5%)** | | | | | |
| $V_{OLT}$ | output voltage LOW (CTX0 and CTX1) | $I_o$ = 1.2 mA; note 15<br>$I_o$ = 10 mA | –<br>– | 0.1<br>0.6 | V<br>V |
| $V_{OHT}$ | output voltage HIGH (CTX0 and CTX1) | $I_o$ = –1.2 mA; note 15<br>$I_o$ = –10 mA; note 16 | $V_{DD}$–0.1<br>$V_{DD}$–0.6 | –<br>– | V<br>V |
| **Reference ($AV_{DD}$ = 5 V ± 5%)** | | | | | |
| $V_{REFOUT}$ | REF output voltage | –0.1 mA < $I_L$ < 0.1 mA; $C_L$ = 10 nF; note 15; bit Reference Active = HIGH | $\frac{1}{2}AV_{DD}$–0.1 | $\frac{1}{2}AV_{DD}$+0.1 | V |
| $I_{REFIN}$ | REF input current | 1.5 V < $V_{REFIN}$ < $AV_{DD}$–1.5 V; bit Reference Active = LOW | – | ±10 | µA |

**Notes to the DC characteristics**

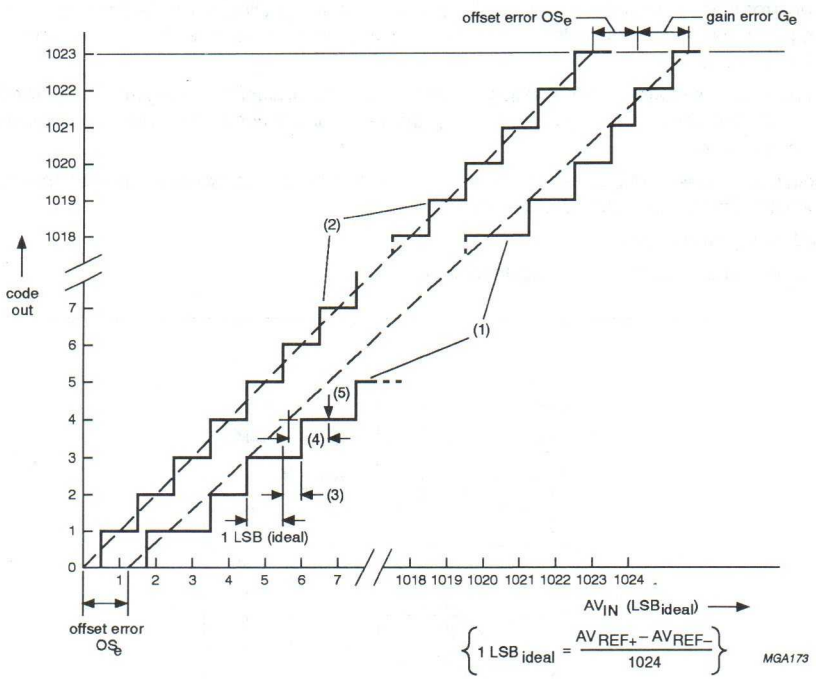
1. Conditions for:
   a) The **digital** operating current measurement: all output pins disconnected; XTAL1 is driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0.5 V; $V_{IH} = V_{DD}$ – 0.5 V; $\overline{EA}$ = RST = Port 0 = P1.6 = P1.7 = $\overline{EW}$ = $V_{DD}$; STADC = $V_{SS}$; CRX0 = 2.7 V; CRX1 = 2.3 V.
   b) The **analog** operating current measurement: Port 5 = $AV_{DD}$; CAN: register 6: = 00H; load current reference voltage source 100 µA.
2. Conditions for:
   a) The **digital** Idle mode supply current measurement: all output pins disconnected; XTAL1 is driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0.5 V; $V_{IH} = V_{DD}$ – 0.5 V; Port 0 = P1.6 = P1.7 = $\overline{EW}$ = $V_{DD}$; $\overline{EA}$ = RST = STADC = $V_{SS}$; CRX0 = 2.7 V; CRX1 = 2.3 V.
   b) The **analog** Idle mode current measurement: Port 5 = $AV_{DD}$; CAN: register 6: = 00H; load current reference voltage source 100 µA.
3. Conditions for:
   a) The **digital** Idle and Sleep mode supply current measurement: all output pins disconnected; XTAL1 is driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0.5 V; $V_{IH} = V_{DD}$ – 0.5 V; Port 0 = P1.6 = P1.7 = $\overline{EW}$ = CRX0 = $V_{DD}$; $\overline{EA}$ = RST = STADC = CRX1 = $V_{SS}$; CAN: register 6: = 00H, register 7: = 12H, register 8: = 02H, register 0: = 20H, wait 15$t_{CY}$, register 1: = 10H, wait for bit Sleep = 1.
   b) The **analog** Idle and Sleep mode current measurement: Port 5 = $AV_{DD}$; load current reference voltage source 100 µA.
4. Window devices have to be covered. Conditions for:
   a) The **digital** Power-down mode supply current measurement: all output pins and Port 5 disconnected; Port 0 = P1.6 = P1.7 = $\overline{EW}$ = CRX0 = $V_{DD}$; $\overline{EA}$ = RST = STADC = CRX1 = XTAL1 = $AV_{REF+}$ = $AV_{REF-}$ = $CV_{SS}$ = $V_{SS}$; $AV_{DD}$ = $V_{DD}$, but current into AVDD pin is not comprised in digital Power-down current.
   b) The **analog** Power-down mode supply current measurement: Port 5 = $AV_{DD}$.
5. Capacitive loads on Port 0 and Port 2 may degrade the LOW level output voltage of ALE, Port 1 and Port 3. During a HIGH-to-LOW transition on the Port 0 and Port 2 pins and a capacitive load > 100 pF, the ALE LOW level may exceed 0.8 V. In the case that it is necessary to connect ALE to a Schmitt trigger input respectively use an address latch with a Schmitt trigger STROBE input.

6. Capacitive loads on Port 0 and Port 2 may cause a HIGH level voltage degradation of ALE and $\overline{PSEN}$ below $0.9V_{DD}$ during the address bits are stabilizing.
7. $t_{CY} = 12\ t_{CLK}$ is the machine cycle time.
8. $AV_{REF+} = 5.12$ V; $AV_{REF-} = 0$ V; $AV_{DD} = 5.0$ V.
9. The differential non-linearity ($DL_e$) is the difference between the actual step width and the ideal step width.
10. The ADC is monotonic, there are no missing codes.
11. The integral non-linearity ($IL_e$) is the peak difference between the centre of the steps of the actual and the ideal transfer curve after appropriate adjustment of gain and offset error.
12. The offset error ($OS_e$) is the absolute difference between the straight line which fits the actual transfer curve after removing gain error, and a straight line which fits the ideal transfer curve. The offset error is constant at every point of the actual transfer curve.
13. The gain error ($G_e$) is relative difference in percent between the straight line fitting the actual transfer curve after removing offset error and the straight line which fits the ideal transfer curve. The gain error is constant at every point on the transfer curve.
14. The absolute voltage error ($A_e$) is the maximum difference between the centre of the steps of the actual transfer curve of the not calibrated ADC and the ideal transfer curve.
15. Not tested during production.
16. Source current for the CTX0, CTX1 outputs together.




(1) Maximum Operating mode ($I_{DD}$); $V_{DD} = 5.5$ V.
(2) Maximum Operating mode ($I_{DD}$); $V_{DD} = 4.5$ V.
(3) Maximum Idle and Sleep mode ($I_{DD(IS)}$); $V_{DD} = 5.5$ V.
(4) Maximum Idle and Sleep mode ($I_{DD(IS)}$); $V_{DD} = 4.5$ V.

Fig.27 Supply current ($I_{DD}$) as a function of frequency at XTAL1 ($f_{CLK}$).




(1) Example of an actual transfer curve.
(2) The ideal transfer curve.
(3) Differential non-linearity ($DL_e$).
(4) Integral non-linearity ($IL_e$).
(5) Centre of a step of the actual transfer curve.

Fig.28 ADC conversion characteristic.

## 21 AC CHARACTERISTICS

See notes 1 and 2; $C_L$ = 100 pF for Port 0, ALE and $\overline{PSEN}$; $C_L$ = 80 pF for all other outputs unless otherwise specified.

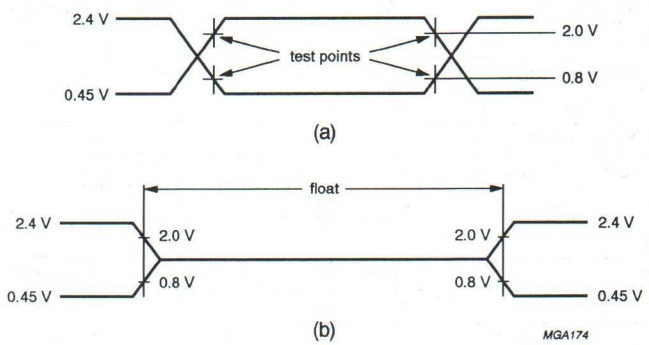
| SYMBOL | PARAMETER | $f_{CLK}$ = 16 MHz | | $f_{CLK}$ = 12 MHz | | VARIABLE CLOCK 1.2 to 16 MHz | | UNIT |
|---|---|---|---|---|---|---|---|---|
| | | MIN. | MAX. | MIN. | MAX. | MIN. | MAX. | |
| **External Program Memory** | | | | | | | | |
| $t_{LHLL}$ | ALE pulse width | 85 | – | 127 | – | $2t_{CLK} - 40$ | – | ns |
| $t_{AVLL}$ | address valid to ALE LOW | 8 | – | 28 | – | $t_{CLK} - 55$ | – | ns |
| $t_{LLAX}$ | address hold after ALE LOW | 28 | – | 48 | – | $t_{CLK} - 35$ | – | ns |
| $t_{LLIV}$ | ALE LOW to valid instruction in | – | 150 | – | 233 | – | $4t_{CLK} - 100$ | ns |
| $t_{LLPL}$ | ALE LOW to $\overline{PSEN}$ LOW | 23 | – | 43 | – | $t_{CLK} - 40$ | – | ns |
| $t_{PLPH}$ | $\overline{PSEN}$ pulse width | 143 | – | 205 | – | $3t_{CLK} - 45$ | – | ns |
| $t_{PLIV}$ | $\overline{PSEN}$ LOW to valid instruction in | – | 83 | – | 145 | – | $3t_{CLK} - 105$ | ns |
| $t_{PXIX}$ | input instruction hold after $\overline{PSEN}$ | 0 | – | 0 | – | 0 | – | ns |
| $t_{PXIZ}$ | input instruction float after $\overline{PSEN}$ | – | 38 | – | 59 | – | $t_{CLK} - 25$ | ns |
| $t_{AVIV}$ | address to valid instruction in | – | 208 | – | 312 | – | $5t_{CLK} - 105$ | ns |
| $t_{PLAZ}$ | $\overline{PSEN}$ LOW to address float | – | 10 | – | 10 | – | 10 | ns |
| **External data memory** | | | | | | | | |
| $t_{RLRH}$ | $\overline{RD}$ pulse width | 275 | – | 400 | – | $6t_{CLK} - 100$ | – | ns |
| $t_{WLWH}$ | $\overline{WR}$ pulse width | 275 | – | 400 | – | $6t_{CLK} - 100$ | – | ns |
| $t_{AVLL}$ | address valid to ALE LOW | 8 | – | 28 | – | $t_{CLK} - 55$ | – | ns |
| $t_{LLAX}$ | address hold after ALE LOW | 33 | – | 53 | – | $t_{CLK} - 30$ | – | ns |
| $t_{RLDV}$ | $\overline{RD}$ LOW to valid data in | – | 148 | – | 252 | – | $5t_{CLK} - 165$ | ns |
| $t_{RHDX}$ | data hold after $\overline{RD}$ | 0 | – | 0 | – | 0 | – | ns |
| $t_{RHDZ}$ | data float after $\overline{RD}$ | – | 55 | – | 97 | – | $2t_{CLK} - 70$ | ns |
| $t_{LLDV}$ | ALE LOW to valid data in | – | 350 | – | 517 | – | $8t_{CLK} - 150$ | ns |
| $t_{AVDV}$ | address to valid data in | – | 398 | – | 585 | – | $9t_{CLK} - 165$ | ns |
| $t_{LLWL}$ | ALE LOW to $\overline{RD}$ or $\overline{WR}$ LOW | 138 | 238 | 200 | 300 | $3t_{CLK} - 50$ | $3t_{CLK} + 50$ | ns |
| $t_{AVWL}$ | address valid to $\overline{RD}$ or $\overline{WR}$ LOW | 120 | – | 203 | – | $4t_{CLK} - 130$ | – | ns |
| $t_{WHLH}$ | $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH | 23 | 103 | 43 | 123 | $t_{CLK} - 40$ | $t_{CLK} + 40$ | ns |
| $t_{QVWX}$ | data valid to $\overline{WR}$ transition | 3 | – | 23 | – | $t_{CLK} - 60$ | – | ns |
| $t_{QVWH}$ | data valid time $\overline{WR}$ HIGH | 288 | – | 433 | – | $7t_{CLK} - 150$ | – | ns |
| $t_{WHQX}$ | data hold after $\overline{WR}$ | 13 | – | 33 | – | $t_{CLK} - 50$ | – | ns |
| $t_{RLAZ}$ | $\overline{RD}$ LOW to address float | – | 0 | – | 0 | – | 0 | ns |

**Notes**

1. For the AC Characteristics the following conditions are valid:**P8xCE598 FxB**: $V_{DD}$ = 5 V ± 10%; $T_{amb}$ = –40 to +85 °C (125 °C); $f_{CLK}$ = 1.2 to 16 MHz.
2. $t_{CLK} = \frac{1}{f_{CLK}}$ = one oscillator clock period; $t_{CLK}$ = 62.5 ns at $f_{CLK}$ = 16 MHz.

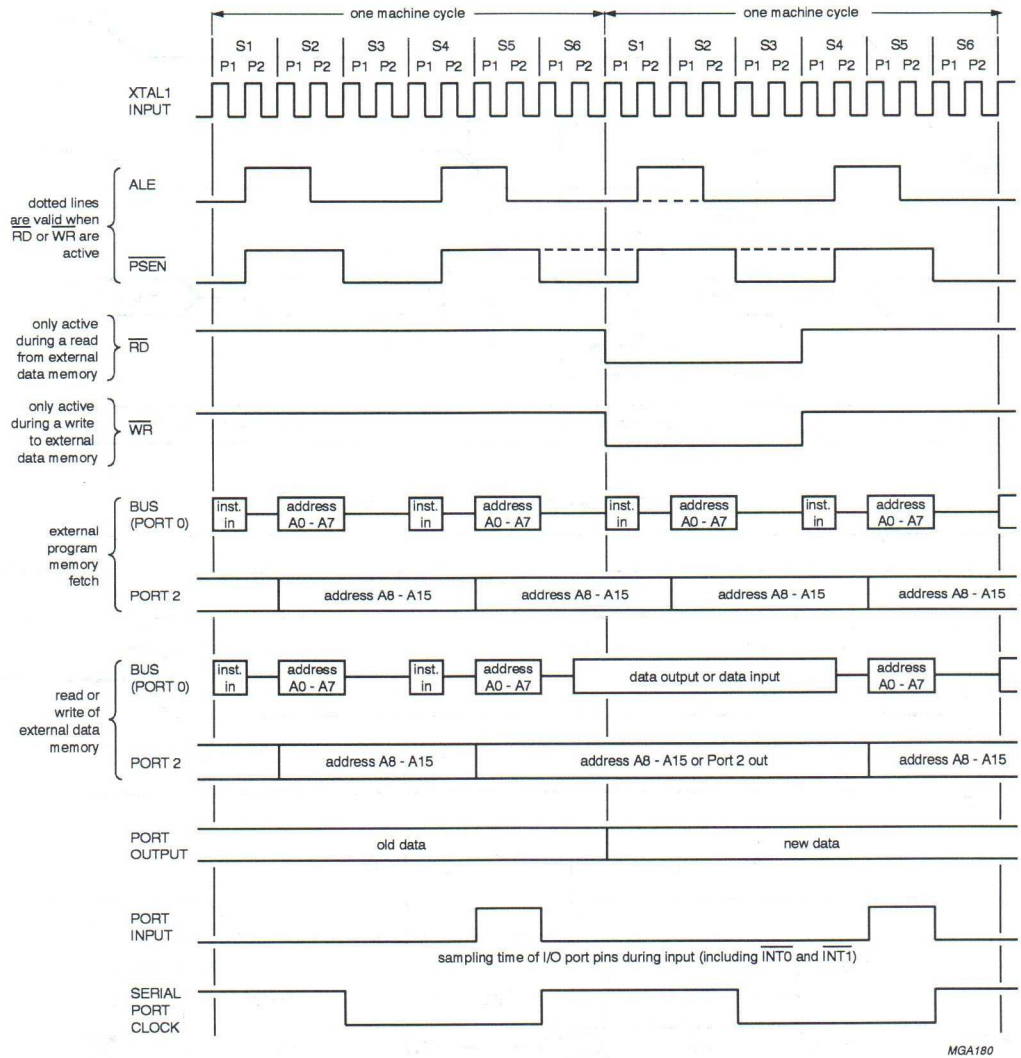
**Table 90** CAN characteristics.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| **CAN input comparator/output driver** | | | | | |
| $t_{sd}$ | sum of input and output delay | $AV_{DD}$ = 5 V ± 5%; $V_{DIF}$ = ± 32 mV; 1.4 V < $V_I$ < $AV_{DD}$ − 1.4 V | | | |
| | P8xCE598 xFx | | – | 60 | ns |
| | P8xCE598 FHB | | – | 70 | ns |




AC testing inputs are driven at 2.4 V for a HIGH and 0.45 V for a LOW.
Timing measurements are taken at 2.0 V for a HIGH and 0.8 V for a LOW, see Fig.29 (a).
The float state is defined as the point at which a Port 0 pin sinks 3.2 mA or sources 400 µA at the voltage test levels, see Fig.29 (b).

Fig.29 AC testing input, output waveform (a) and float waveform (b).




The Port 5 input buffers have a maximum propagation delay of 300 ns.
As a result Port 5 sample time begins 300 ns before state S5 and ends when S5 has finished.

Fig.30 Instruction cycle timing.

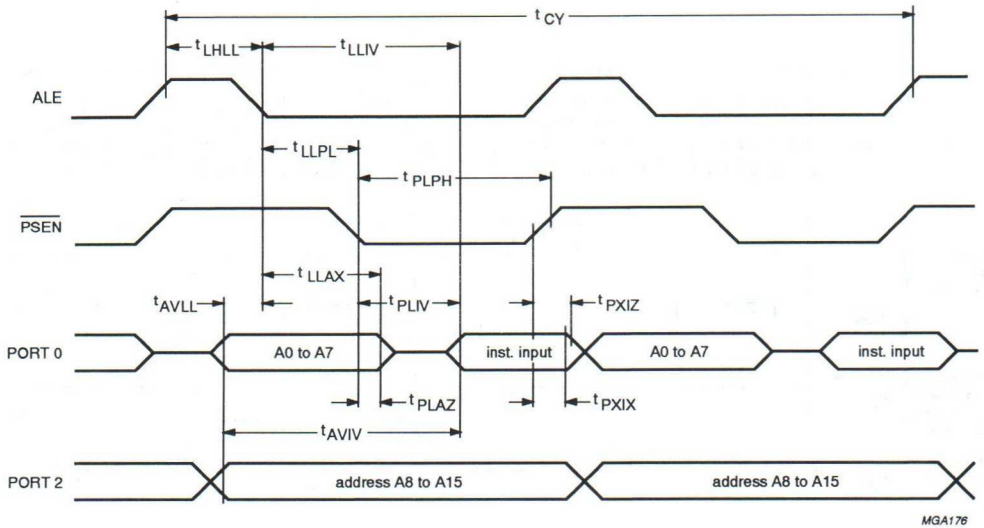



Fig.31 Read from external Program Memory.

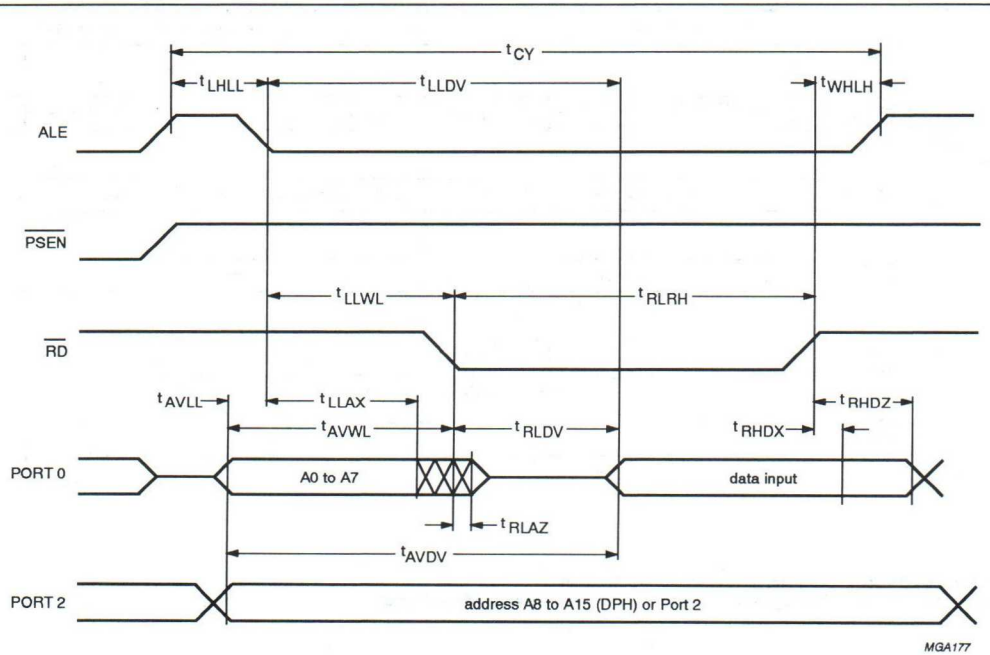



Fig.32 Read from external Data Memory.

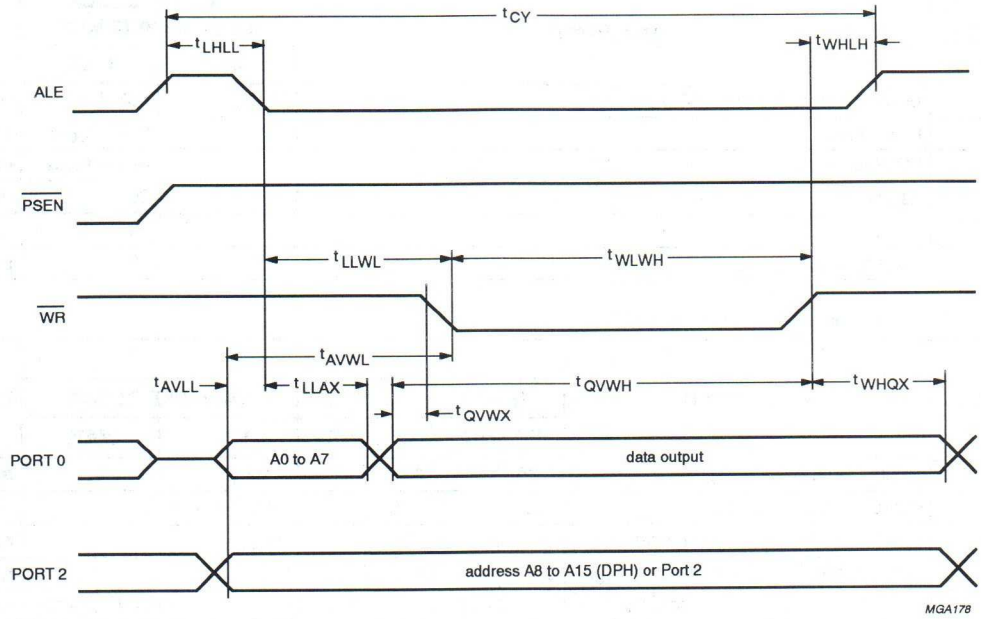



Fig.33 Write to external Data Memory.

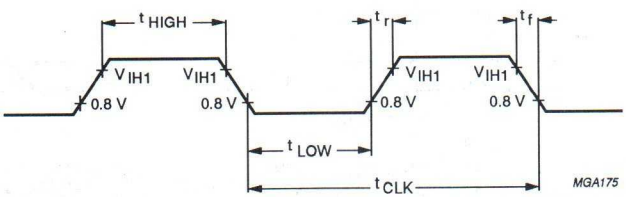



Fig.34 External clock drive XTAL1 (see Table 91).

**Table 91** External clock drive XTAL1

| SYMBOL | PARAMETER | VARIABLE CLOCK ($f_{CLK}$ = 1.2 to 16 MHz) | | UNIT |
|---|---|---|---|---|
| | | MIN. | MAX. | |
| $t_{CLK}$ | oscillator clock period (P83CE598) | 62.5 | 833.3 | ns |
| $t_{HIGH}$ | HIGH time | 20 | $t_{CLK} - t_{LOW}$ | ns |
| $t_{LOW}$ | LOW time | 20 | $t_{CLK} - t_{HIGH}$ | ns |
| $t_r$ | rise time | – | 20 | ns |
| $t_f$ | fall time | – | 20 | ns |
| $t_{CY}$ | cycle time ($12 \times t_{CLK}$) | 0.75 | 10 | µs |

**Table 92** UART Timing in Shift Register Mode

| SYMBOL | PARAMETER | $f_{CLK}$ | | | | | | UNIT |
|---|---|---|---|---|---|---|---|---|
| | | 16 MHz | | 12 MHz | | VARIABLE CLOCK | | |
| | | MIN. | MAX. | MIN. | MAX. | MIN. | MAX. | |
| $t_{XLXL}$ | Serial Port clock cycle timing | 0.75 | – | 1.0 | – | 12 $t_{CLK}$ | – | ms |
| $t_{QVXH}$ | output data setup to clock rising edge | 492 | – | 700 | – | 10 $t_{CLK}$ – 133 | – | ns |
| $t_{XHQX}$ | output data hold after clock rising edge | 8.0 | – | 50 | – | 2 $t_{CLK}$ – 117 | – | ns |
| $t_{XHDX}$ | input data hold after clock rising edge | 0 | – | 0 | – | 0 | – | ns |
| $t_{XHDV}$ | clock rising edge to input data valid | – | 492 | – | 700 | – | 10 $t_{CLK}$ – 133 | ns |

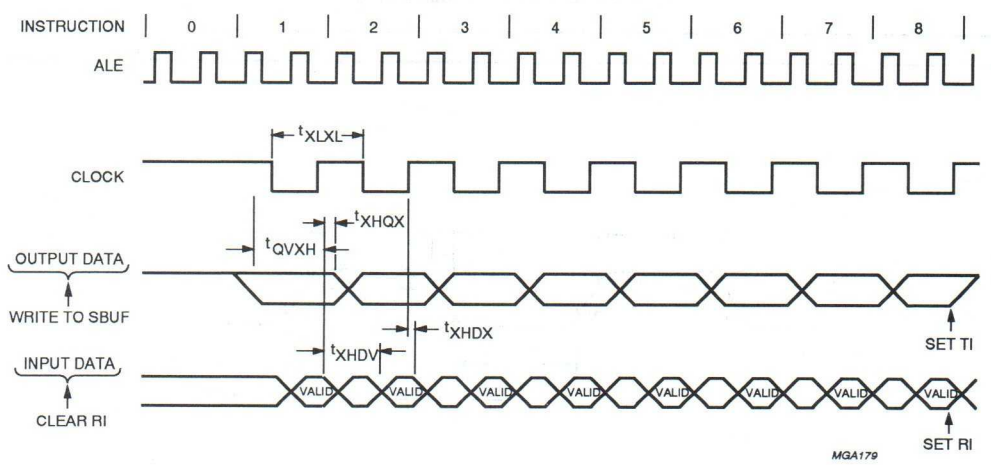



Fig.35 UART waveforms in Shift Register Mode.

## 22 CAN APPLICATION INFORMATION

**NOTE:** For chapter 22, please refer to the 8xC592 datasheet.

# CMOS single-chip 8-bit microcontrollers — 80C652/83C652

## DESCRIPTION

The P80C652/83C652 Single-Chip 8-Bit Microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 80C652/83C652 has the same instruction set as the 80C51. Three versions of the derivative exist:

83C652 — 8k bytes mask programmable ROM

80C652 — ROMless version

87C652 — EPROM version (described in a separate chapter)

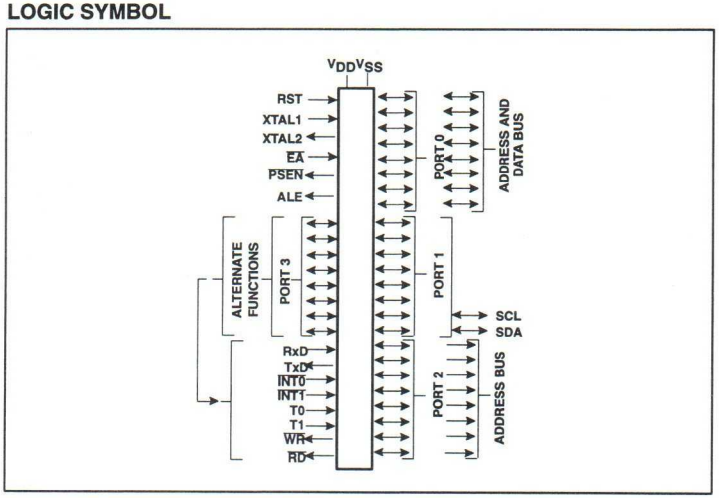
This device provides architectural enhancements that make it applicable in a variety of applications for general control systems. The 8XC652 contains a non-volatile 8k × 8 read-only program memory, a volatile 256 × 8 read/write data memory, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a multi-source, two-priority-level, nested interrupt structure, an $I^2C$ interface, UART and on-chip oscillator and timing circuits. For systems that require extra capability, the 8XC652 can be expanded using standard TTL compatible memories and logic.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte and 17 three-byte. With a 16(24)MHz crystal, 58% of the instructions are executed in 0.75(0.5)µs and 40% in 1.5(1)µs. Multiply and divide instructions require 3(2)µs.

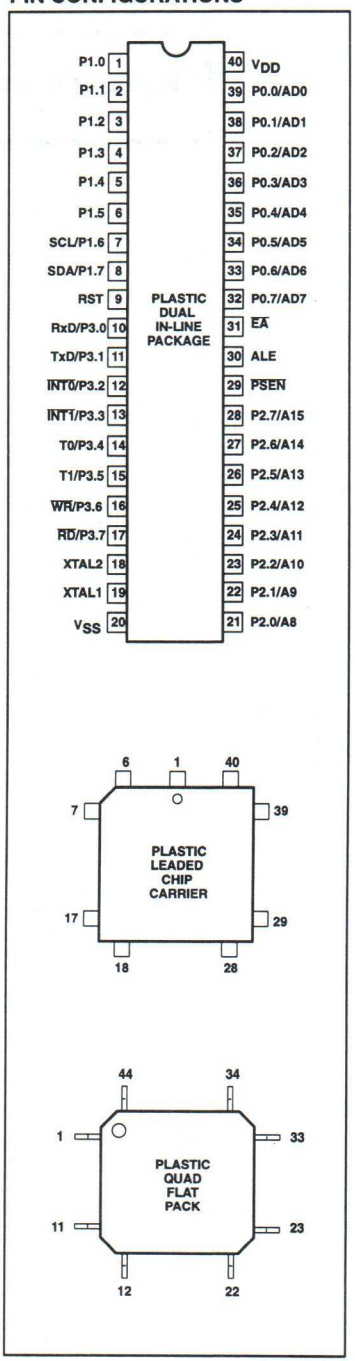
## FEATURES

- 80C51 central processing unit
- 8k × 8 ROM expandable externally to 64k bytes
- 256 × 8 RAM, expandable externally to 64k bytes
- Two standard 16-bit timer/counters
- Four 8-bit I/O ports
- $I^2C$-bus serial I/O port with byte oriented master and slave functions
- Full-duplex UART facilities
- Power control modes
  - Idle mode
  - Power-down mode
- ROM code protection
- Extended frequency range: 1.2 to 24 MHz
- Three operating ambient temperature ranges:
  - 0 to +70°C
  - –40 to +85°C
  - –40 to +125°C

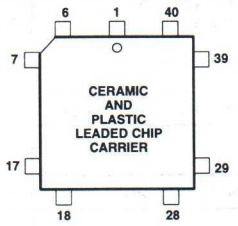
## LOGIC SYMBOL




## PIN CONFIGURATIONS



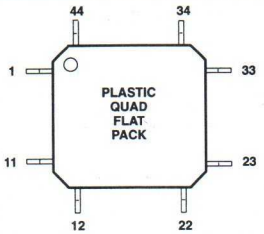
## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC* | 23 | NC* |
| 2 | P1.0 | 24 | P2.0/A8 |
| 3 | P1.1 | 25 | P2.1/A9 |
| 4 | P1.2 | 26 | P2.2/A10 |
| 5 | P1.3 | 27 | P2.3/A11 |
| 6 | P1.4 | 28 | P2.4/A12 |
| 7 | P1.5 | 29 | P2.5/A13 |
| 8 | P1.6/SCL | 30 | P2.6/A14 |
| 9 | P1.7/SDA | 31 | P2.7/A15 |
| 10 | RST | 32 | $\overline{PSEN}$ |
| 11 | P3.0/RxD | 33 | ALE |
| 12 | NC* | 34 | NC* |
| 13 | P3.1/TxD | 35 | $\overline{EA}$ |
| 14 | P3.2/$\overline{INT0}$ | 36 | P0.7/AD7 |
| 15 | P3.3/$\overline{INT1}$ | 37 | P0.6/AD6 |
| 16 | P3.4/T0 | 38 | P0.5/AD5 |
| 17 | P3.5/T1 | 39 | P0.4/AD4 |
| 18 | P3.6/$\overline{WR}$ | 40 | P0.3/AD3 |
| 19 | P3.7/$\overline{RD}$ | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | $V_{SS}$ | 44 | $V_{DD}$ |

*DO NOT CONNECT

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5 | 23 | P2.5/A13 |
| 2 | P1.6/SCL | 24 | P2.6/A14 |
| 3 | P1.7/SDA | 25 | P2.7/A15 |
| 4 | RST | 26 | $\overline{PSEN}$ |
| 5 | P3.0/RxD | 27 | ALE |
| 6 | $V_{SS4}$ | 28 | $V_{SS2}$ |
| 7 | P3.1/TxD | 29 | $\overline{EA}$/$V_{PP}$ |
| 8 | P3.2/$\overline{INT0}$ | 30 | P0.7/AD7 |
| 9 | P3.3/$\overline{INT1}$ | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/$\overline{WR}$ | 34 | P0.3/AD3 |
| 13 | P3.7/$\overline{RD}$ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | $V_{SS1}$ | 38 | $V_{DD}$ |
| 17 | NC* | 39 | $V_{SS3}$ |
| 18 | P2.0/A8 | 40 | P1.0 |
| 19 | P2.1/A9 | 41 | P1.1 |
| 20 | P2.2/A10 | 42 | P1.2 |
| 21 | P2.3/A11 | 43 | P1.3 |
| 22 | P2.4/A12 | 44 | P1.4 |

*DO NOT CONNECT

**NOTES TO QFP ONLY:**

1. Due to EMC improvements, all $V_{SS}$ pins (6, 16, 28, 39) must be connected to $V_{SS}$ on the 80C652/83C652.

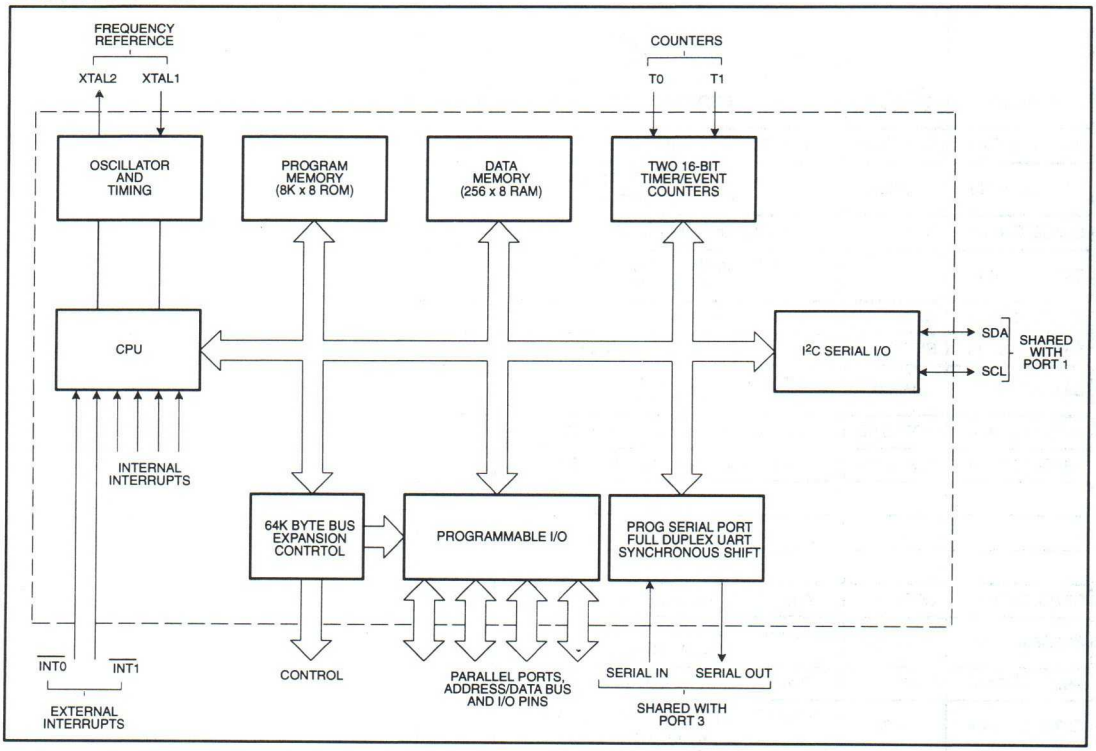
## ORDER INFORMATION

| PHILIPS PART ORDER NUMBER PART MARKING | | | PHILIPS NORTH AMERICA PART ORDER NUMBER | | TEMPERATURE RANGE (°C) AND PACKAGE | FREQ MHz[1,2] |
|---|---|---|---|---|---|---|
| ROMless | ROM[3] | Drawing Number | ROMless | ROM | | |
| P80C652FBP | P83C652FBP/xxx | SOT129-1 | P80C652FBPN | P83C652FBPN | 0 to +70, Plastic Dual In-line Package | 16 |
| | | | | | | |
| P80C652FBA | P83C652FBA/xxx | SOT187-2 | P80C652FBAA | P83C652FBAA | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| | | | | | | |
| P80C652FBB | P83C652FBB/xxx | SOT307-2 [4] | P80C652FBBB | P83C652FBBB | 0 to +70, Plastic Quad Flat Pack | 16 |
| P80C652FFP | P83C652FFP/xxx | SOT129-1 | P80C652FFPN | P83C652FFPN | –40 to +85, Plastic Dual In-line Package | 16 |
| | | | | | | |
| P80C652FFA | P83C652FFA/xxx | SOT187-2 | P80C652FFAA | P83C652FFAA | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| P80C652FFB | P83C652FFB/xxx | SOT307-2 [4] | P80C652FFBB | P83C652FFBB | –40 to +85, Plastic Quad Flat Pack | 16 |
| P80C652FHP | P83C652FHP/xxx | SOT129-1 | P80C652FHPN | P83C652FHPN | –40 to +125, Plastic Dual In-line Package | 16 |
| P80C652FHA | P83C652FHA/xxx | SOT187-2 | P80C652FHAA | P83C652FHAA | –40 to +125, Plastic Leaded Chip Carrier | 16 |
| P80C652FHB | P83C652FHB/xxx | SOT307-2 [4] | P80C652FHBB | P83C652FHBB | –40 to +125, Plastic Quad Flat Pack | 16 |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| P80C652IBP | P83C652IBP/xxx | SOT129-1 | P80C652IBPN | P83C652IBPN | 0 to +70, Plastic Dual In-line Package | 24 |
| P80C652IBA | P83C652IBA/xxx | SOT187-2 | P80C652IBAA | P83C652IBAA | 0 to +70, Plastic Leaded Chip Carrier | 24 |
| P80C652IBB | P83C652IBB/xxx | SOT307-2 [4] | P80C652IBBB | P83C652IBBB | 0 to +70, Plastic Quad Flat Pack | 24 |
| P80C652IFP | P83C652IFP/xxx | SOT129-1 | P80C652IFPN | P83C652IFPN | –40 to +85, Plastic Dual In-line Package | 24 |
| P80C652IFA | P83C652IFA/xxx | SOT187-2 | P80C652IFAA | P83C652IFAA | –40 to +85, Plastic Leaded Chip Carrier | 24 |
| P80C652IFB | P83C652IFB/xxx | SOT307-2 [4] | P80C652IFBB | P83C652IFBB | –40 to +85, Plastic Quad Flat Pack | 24 |

**NOTES:**
1. 80C652 and 83C652 frequency range is 1.2MHz–16MHz or 1.2 to 24MHz.
2. For specification of the EPROM version, see the 87C652 data sheet.
3. xxx denotes the ROM code number.
4. SOT311 replaced by SOT307-2.

| EPROM[2] | Drawing Number | TEMPERATURE RANGE (°C) AND PACKAGE | FREQ MHz[1,2] |
|---|---|---|---|
| S87C652-4N40 | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 16 |
| S87C652-4F40 | 0590B | 0 to +70, Ceramic Dual In-line Package w/Window | 16 |
| S87C652-4A44 | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| S87C652-4K44 | 1472A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 16 |
| S87C652-4B44 | SOT307-2 | 0 to +70, Plastic Quad Flat Pack | 16 |
| S87C652-5N40 | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 16 |
| S87C652-5F40 | 0590B | –40 to +85, Ceramic Dual In-line Package w/Window | 16 |
| S87C652-5A44 | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| S87C652-5B44 | SOT307-2 | –40 to +85, Plastic Quad Flat Pack | 16 |
| | | | |
| | | | |
| | | | |
| S87C652-7N40 | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 20 |
| S87C652-7F40 | 0590B | 0 to +70, Ceramic Dual In-line Package w/Window | 20 |
| S87C652-7A44 | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 20 |
| S87C652-7K44 | 1472A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 20 |
| S87C652-8N40 | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 20 |
| S87C652-8F40 | 0590B | –40 to +85, Ceramic Dual In-line Package w/Window | 20 |
| S87C652-8A44 | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 20 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

## BLOCK DIAGRAM



## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER DIP | PLCC | QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 6, 16, 28, 39 | I | **Ground:** 0V reference. With the QFP package all $V_{SS}$ pins ($V_{SS1}$ to $V_{SS4}$) must be connected. |
| $V_{DD}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups, except P1.6 and P1.7 which are open drain. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Alternate functions include: |
| P1.6 | 7 | 8 | 2 | I/O | **SCL:** $I^2C$-bus serial port clock line. |
| P1.7 | 8 | 9 | 3 | I/O | **SDA:** $I^2C$-bus serial port data line. |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{DD}$. |
| ALE | 30 | 33 | 27 | I/O | **Address Latch Enable:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency. Note that one ALE pulse is skipped during each access to external data memory. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** Read strobe to external program memory via Port 0 and Port 2. It is activated twice each machine cycle during fetches from the external program memory. When executing out of external program memory two activations of $\overline{PSEN}$ are skipped during each access to external data memory. $\overline{PSEN}$ is not activated (remains HIGH) during no fetches from external program memory. $\overline{PSEN}$ can sink/source 8 LSTTL inputs and can drive CMOS inputs without external pull–ups. |
| $\overline{EA}$ | 31 | 35 | 29 | I | **External Access:** If during a RESET, $\overline{EA}$ is held at TTL, level HIGH, the CPU executes out of the internal program memory ROM provided the Program Counter is less than 8192. If during a RESET, $\overline{EA}$ is held a TTL LOW level, the CPU executes out of external program memory. $\overline{EA}$ is not allowed to float. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTE:**
To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher than $V_{DD}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

**Table 1. 8XC652/654 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes) | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 | 0x000000B |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP*# | Interrupt priority | B8H | – | | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 | xx000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1*# | Port 1 | 90H | SDA | SCL | | | | | | | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TXD | RXD | FFH |
| PCON | Power control | 87H | SMOD | – | – | – | GF1 | GF0 | PD | IDL | 0xxx0000B |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| S0CON*# | Serial 0 port control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| S0BUF# | Serial 0 data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | F1 | P | 00H |
| S1DAT# | Serial 1 data | DAH | | | | | | | | | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| S1ADR# | Serial 1 address | DBH | SLAVE ADDRESS | | | | | | | GC | 00H |
| | | | | | | | | | | | |
| S1STA# | Serial 1 status | D9H | SC4 | SC3 | SC2 | SC1 | SC0 | 0 | 0 | 0 | F8H |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| S1CON*# | Serial 1 control | D8H | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 | 00000000B |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| TH1 | Timer high 1 | 8DH | | | | | | | | | 00H |
| TH0 | Timer high 0 | 8CH | | | | | | | | | 00H |
| TL1 | Timer low 1 | 8BH | | | | | | | | | 00H |
| TL0 | Timer low 0 | 8AH | | | | | | | | | 00H |
| TMOD | Timer mode | 89H | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

## ROM CODE PROTECTION (83C652)

The 8XC652 has an additional security feature. ROM code protection may be selected by setting a mask–programmable security bit (i.e., user dependent). This feature may be requested during ROM code submission. When selected, the ROM code is protected and cannot be read out at any time by any test mode or by any instruction in the external program memory space.

The MOVC instructions are the only instructions that have access to program code in the internal or external program memory. The $\overline{EA}$ input is latched during RESET and is "don't care" after RESET (also if the security bit is not set). This implementation prevents reading internal program code by switching from external program memory to internal program memory during a MOVC instruction or any other instruction that uses immediate data.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the Logic Symbol, page 3-992.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## Reset

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{DD}$ and RST must come up at the same time for a proper start-up.

## Idle Mode

In the idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## Power-Down Mode

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. The control bits for the reduced power modes are in the special function register PCON. Table 2 shows the state of the I/O ports during low current operating modes.

## $I^2C$ Serial Communication—SIO1

The $I^2C$ serial port is identical to the $I^2C$ serial port on the 8XC552. The operation of this subsystem is described in detail in the 8XC552 section of this manual.

Note that in both the 8XC652/4 and the 8XC552 the $I^2C$ pins are alternate functions to port pins P1.6 and P1.7. Because of this, P1.6 and P1.7 on these parts do not have a pull-up structure as found on the 80C51. Therefore P1.6 and P1.7 have open drain outputs on the 8XC652/4.

**Table 2. External Pin Status During Idle and Power-Down Mode**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

**Serial Control Register (S1CON) – See Table 3**

| S1CON (D8H) | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |
|---|---|---|---|---|---|---|---|---|

**Bits CR0, CR1 and CR2 determine the serial clock frequency that is generated in the master mode of operation.**

**Table 3. Serial Clock Rates**

| CR2 | CR1 | CR0 | BIT FREQUENCY (kHz) AT $f_{OSC}$ 6MHz | 12MHz | 16MHz | 24MHz | $f_{OSC}$ DIVIDED BY |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 23 | 47 | 62.5 | 94 | 256 |
| 0 | 0 | 1 | 27 | 54 | 71 | 107[1] | 224 |
| 0 | 1 | 0 | 31.25 | 62.5 | 83.3 | 125[1] | 192 |
| 0 | 1 | 1 | 37 | 75 | 100 | 150[1] | 160 |
| 1 | 0 | 0 | 6.25 | 12.5 | 17 | 25 | 960 |
| 1 | 0 | 1 | 50 | 100 | 133[1] | 200[1] | 120 |
| 1 | 1 | 0 | 100 | 200[1] | 267[1] | 400[1] | 60 |
| 1 | 1 | 1 | 0.24 < 62.5<br>0 to 255 | 0.49 < 62.5<br>0 to 254 | 0.65 < 55.6<br>0 to 253 | 0.98 < 50.0<br>0 to 251 | 96 × (256 – (reload value Timer 1))<br>reload value range Timer 1 (in mode 2) |

**NOTES:**

1. These frequencies exceed the upper limit of 100kHz of the $I^2C$-bus specification and cannot be used in an $I^2C$-bus application.

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage on any other pin to $V_{SS}$ | –0.5 to + 6.5 | V |
| Input, output current on any single pin | ±5 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DEVICE SPECIFICATIONS

| TYPE | SUPPLY VOLTAGE (V) | | FREQUENCY (MHz) | | TEMPERATURE RANGE (°C) |
|---|---|---|---|---|---|
| | MIN. | MAX. | MIN. | MAX. | |
| P8XC652FBx | 4.0 | 6.0 | 1.2 | 16 | 0 to +70 |
| P8XC652FFx | 4.0 | 6.0 | 1.2 | 16 | –40 to +85 |
| P8XC652FHx | 4.5 | 5.5 | 1.2 | 16 | –40 to +125 |
| P8XC652IBx | 4.5 | 5.5 | 1.2 | 24 | 0 to +70 |
| P83X652IFx | 4.5 | 5.5 | 1.2 | 24 | –40 to +85 |

## DC ELECTRICAL CHARACTERISTICS

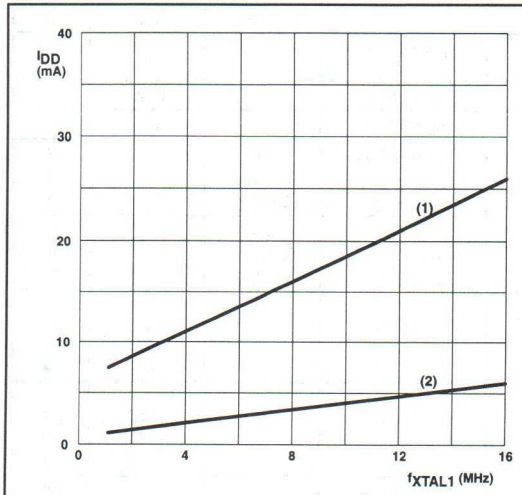
$V_{SS}$ = 0V

| SYMBOL | PARAMETER | PART TYPE | TEST CONDITIONS | LIMITS MIN. | LIMITS MAX. | UNIT |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input low voltage, except $\overline{EA}$, P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C<br>–40 to +125°C | | –0.5<br>–0.5<br>–0.5 | $0.2V_{DD}$–0.1<br>$0.2V_{DD}$–0.15<br>$0.2V_{DD}$–0.25 | V<br>V<br>V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | 0 to +70°C<br>–40 to +85°C<br>–40 to +125°C | | –0.5<br>–0.5<br>–0.5 | $0.2V_{DD}$–0.3<br>$0.2V_{DD}$–0.35<br>$0.2V_{DD}$–0.45 | V<br>V<br>V |
| $V_{IL2}$ | Input low voltage to P1.6/SCL, P1.7/SDA[6] | | | –0.5 | $0.3V_{DD}$ | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST, P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C<br>–40 to +125°C | | $0.2V_{DD}$+0.9<br>$0.2V_{DD}$+1.0<br>$0.2V_{DD}$+1.0 | $V_{DD}$+0.5<br>$V_{DD}$+0.5<br>$V_{DD}$+0.5 | V<br>V<br>V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | 0 to +70°C<br>–40 to +85°C<br>–40 to +125°C | | $0.7V_{DD}$<br>$0.7V_{DD}$+0.1<br>$0.7V_{DD}$+0.1 | $V_{DD}$+0.5<br>$V_{DD}$+0.5<br>$V_{DD}$+0.5 | V<br>V<br>V |
| $V_{IH2}$ | Input high voltage, P1.6/SCL, P1.7/SDA[6] | | | $0.7V_{DD}$ | 6.0 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3, except P1.6/SCL, P1.7/SDA | | $I_{OL}$ = 1.6mA[8, 9] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$ | | $I_{OL}$ = 3.2mA[8, 9] | | 0.45 | V |
| $V_{OL2}$ | Output low voltage, P1.6/SCL, P1.7/SDA | | $I_{OL}$ = 3.0mA | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3, ALE, $\overline{PSEN}$[10] | | $I_{OH}$ = –60µA<br>$I_{OH}$ = –25µA<br>$I_{OH}$ = –10µA | 2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | V<br>V<br>V |
| $V_{OH1}$ | Output high voltage; port 0 in external bus mode | | $I_{OH}$ = –800µA<br>$I_{OH}$ = –300µA<br>$I_{OH}$ = –80µA | 2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | V<br>V<br>V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3, except P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C<br>–40 to +125°C | $V_{IN}$ = 0.45V | | –50<br>–75<br>–75 | µA<br>µA<br>µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3, except P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C<br>–40 to +125°C | See note 7 | | –650<br>–750<br>–750 | µA<br>µA<br>µA |
| $I_{L1}$ | Input leakage current, port 0, $\overline{EA}$ | | 0.45V < $V_I$ < $V_{DD}$ | | ±10 | µA |
| $I_{L2}$ | Input leakage current, P1.6/SCL, P1.7/SDA | | 0V < $V_I$ < 6.0V<br>0V < $V_{DD}$ < 6.0V | | ±10 | µA<br>µA |
| $I_{DD}$ | Power supply current:<br>Active mode @ 16MHz[2, 11]<br>Active mode @ 24MHz[2, 11]<br>Idle mode @ 16MHz[3, 11]<br>Idle mode @ 24MHz[3, 11]<br>Power down mode[4, 5]<br>Power down mode[4, 5] | <br><br><br><br><br><br>–40 to +125°C | See note 1<br>$V_{DD}$=6.0V<br>$V_{DD}$=5.5V | | <br>26.5<br>33.8<br>6<br>7<br>50<br>100 | <br>mA<br>mA<br>mA<br>mA<br>µA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | | 50 | 150 | kΩ |
| $C_{IO}$ | Pin capacitance | | Freq.=1MHz | | 10 | pF |

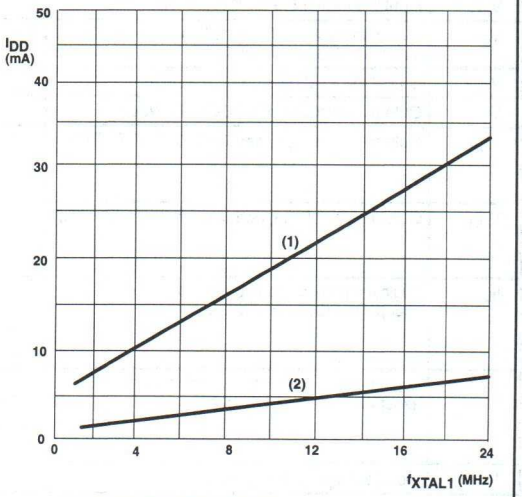
**NOTES ON NEXT PAGE.**

**NOTES FOR DC ELECTRICAL CHARACTERISTICS:**

1. See Figures 9 through 11 for $I_{DD}$ test conditions.
2. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5V$; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = P1.6 = P1.7 = $V_{DD}$. See Figure 9.
3. The idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5V$; XTAL2 not connected; Port 0 = P1.6 = P1.7 = $V_{DD}$; $\overline{EA}$ = RST = $V_{SS}$. See Figure 10.
4. The power-down current is measured with all output pins disconnected; XTAL2 not connected; Port 0 = P1.6 = P1.7 = $V_{DD}$; $\overline{EA}$ = RST = $V_{SS}$. See Figure 11.
5. $2V \leq V_{PD} \leq V_{DD}max$.
6. The input threshold voltage of P1.6 and P1.7 (SIO1) meets the $I^2C$ specification, so an input voltage below $0.3V_{DD}$ will be recognized as a logic 0 while an input voltage above $0.7V_{DD}$ will be recognized as a logic 1.
7. Pins of ports 1 , 2, and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
8. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.
9. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows: Maximum $I_{OL}$ = 10mA per port pin; Maximum $I_{OL}$ = 26mA total for Port 0; Maximum $I_{OL}$ = 15mA total for Ports 1, 2, and 3; Maximum $I_{OL}$ = 71mA total for all output pins. If $I_{OL}$ exceeds the test conditions, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
10. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{DD}$ specification when the address bits are stabilizing.
11. $I_{DDMAX}$ for other frequencies can be derived from Figure 1, where FREQ is the external oscillator frequency in MHz. $I_{DDMAX}$ is given in mA.




(1) MAXIMUM OPERATING MODE: $V_{DD} = V_{DDmax}$
(2) MAXIMUM IDLE MODE: $V_{DD} = V_{DDmax}$
These values are valid within the specified frequency range.




(1) MAXIMUM OPERATING MODE: $V_{DD} = V_{DDmax}$
(2) MAXIMUM IDLE MODE: $V_{DD} = V_{DDmax}$
These values are valid within the specified frequency range.

**Figure 1. $I_{DD}$ vs. Frequency**

## AC ELECTRICAL CHARACTERISTICS[1, 2] (16 MHz type)

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 2 | Oscillator frequency | | | 1.2 | 16 | MHz |
| $t_{LHLL}$ | 2 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 2 | Address valid to ALE low | 8 | | $t_{CLCL}-55$ | | ns |
| $t_{LLAX}$ | 2 | Address hold after ALE low | 28 | | $t_{CLCL}-35$ | | ns |
| $t_{LLIV}$ | 2 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 2 | ALE low to $\overline{PSEN}$ low | 23 | | $t_{CLCL}-40$ | | ns |
| $t_{PLPH}$ | 2 | $\overline{PSEN}$ pulse width | 143 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 2 | $\overline{PSEN}$ low to valid instruction in | | 83 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 2 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 2 | Input instruction float after $\overline{PSEN}$ | | 38 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 2 | Address to valid instruction in | | 208 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 2 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 3, 4 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 3, 4 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 3, 4 | $\overline{RD}$ low to valid data in | | 148 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 3, 4 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 3, 4 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-70$ | ns |
| $t_{LLDV}$ | 3, 4 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 3, 4 | Address to valid data in | | 398 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 3, 4 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 138 | 238 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 3, 4 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 120 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 3, 4 | Data valid to $\overline{WR}$ transition | 3 | | $t_{CLCL}-60$ | | ns |
| $t_{DW}$ | 3, 4 | Data setup time before $\overline{WR}$ | 288 | | $7t_{CLCL}-150$ | | ns |
| $t_{WHQX}$ | 3, 4 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{RLAZ}$ | 3, 4 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 3, 4 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 5 | Serial port clock cycle time[3] | 0.75 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 5 | Output data setup to clock rising edge[3] | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 5 | Output data hold after clock rising edge[3] | 80 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 5 | Input data hold after clock rising edge[3] | 0 | | 0 | | ns |
| $t_{XHDV}$ | 5 | Clock rising edge to input data valid[3] | | 492 | | $10t_{CLCL}-133$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 6 | High time[3] | 20 | | 20 | $t_{CLCL}-t_{CLCX}$ | ns |
| $t_{CLCX}$ | 6 | Low time[3] | 20 | | 20 | $t_{CLCL}-t_{CHCX}$ | ns |
| $t_{CLCH}$ | 6 | Rise time[3] | | 20 | | 20 | ns |
| $t_{CHCL}$ | 6 | Fall time[3] | | 20 | | 20 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. These values are characterized but not 100% production tested.

## AC ELECTRICAL CHARACTERISTICS[1, 2] (24 MHz type)

| SYMBOL | FIGURE | PARAMETER | 24MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 2 | Oscillator frequency | | | 1.2 | 24 | MHz |
| $t_{LHLL}$ | 2 | ALE pulse width | 43 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 2 | Address valid to ALE low | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{LLAX}$ | 2 | Address hold after ALE low | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{LLIV}$ | 2 | ALE low to valid instruction in | | 102 | | $4t_{CLCL}-65$ | ns |
| $t_{LLPL}$ | 2 | ALE low to $\overline{PSEN}$ low | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{PLPH}$ | 2 | $\overline{PSEN}$ pulse width | 80 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 2 | $\overline{PSEN}$ low to valid instruction in | | 65 | | $3t_{CLCL}-60$ | ns |
| $t_{PXIX}$ | 2 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 2 | Input instruction float after $\overline{PSEN}$ | | 17 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 2 | Address to valid instruction in | | 128 | | $5t_{CLCL}-80$ | ns |
| $t_{PLAZ}$ | 2 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 3, 4 | $\overline{RD}$ pulse width | 150 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 3, 4 | $\overline{WR}$ pulse width | 150 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 3, 4 | $\overline{RD}$ low to valid data in | | 118 | | $5t_{CLCL}-90$ | ns |
| $t_{RHDX}$ | 3, 4 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 3, 4 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-28$ | ns |
| $t_{LLDV}$ | 3, 4 | ALE low to valid data in | | 180 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 3, 4 | Address to valid data in | | 210 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 3, 4 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 75 | 175 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 3, 4 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 92 | | $4t_{CLCL}-75$ | | ns |
| $t_{QVWX}$ | 3, 4 | Data valid to $\overline{WR}$ transition | 12 | | $t_{CLCL}-30$ | | ns |
| $t_{DW}$ | 3, 4 | Data setup time before $\overline{WR}$ | 162 | | $7t_{CLCL}-130$ | | ns |
| $t_{WHQX}$ | 3, 4 | Data hold after $\overline{WR}$ | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{RLAZ}$ | 3, 4 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 3, 4 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 17 | 67 | $t_{CLCL}-25$ | $t_{CLCL}+25$ | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 5 | Serial port clock cycle time[3] | 0.5 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 5 | Output data setup to clock rising edge[3] | 283 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 5 | Output data hold after clock rising edge[3] | 23 | | $2t_{CLCL}-60$ | | ns |
| $t_{XHDX}$ | 5 | Input data hold after clock rising edge[3] | 0 | | 0 | | ns |
| $t_{XHDV}$ | 5 | Clock rising edge to input data valid[3] | | 283 | | $10t_{CLCL}-133$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 6 | High time[3] | 17 | | 17 | $t_{CLCL} - t_{CLCX}$ | ns |
| $t_{CLCX}$ | 6 | Low time[3] | 17 | | 17 | $t_{CLCL} - t_{CHCX}$ | ns |
| $t_{CLCH}$ | 6 | Rise time[3] | | 5 | | 5 | ns |
| $t_{CHCL}$ | 6 | Fall time[3] | | 5 | | 5 | ns |

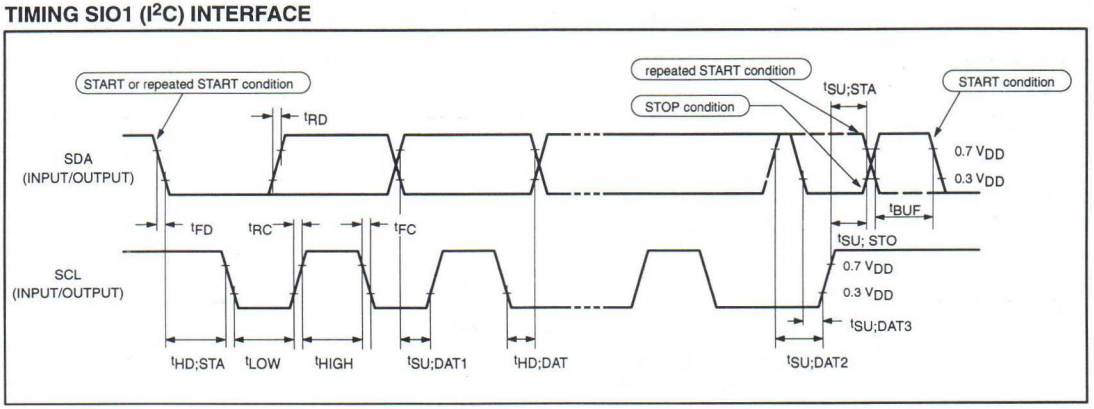
**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. These values are characterized but not 100% production tested.

## AC ELECTRICAL CHARACTERISTICS – I²C INTERFACE

| SYMBOL | PARAMETER | INPUT | OUTPUT |
|---|---|---|---|
| **SCL TIMING CHARACTERISTICS** | | | |
| $t_{HD;STA}$ | START condition hold time | ≥ 14 $t_{CLCL}$ | > 4.0µs[1] |
| $t_{LOW}$ | SCL LOW time | ≥ 16 $t_{CLCL}$ | > 4.7µs[1] |
| $t_{HIGH}$ | SCL HIGH time | ≥ 14 $t_{CLCL}$ | > 4.0µs[1] |
| $t_{RC}$ | SCL rise time | ≤ 1µs | –[2] |
| $t_{FC}$ | SCL fall time | ≤ 0.3µs | < 0.3µs[3] |
| **SDA TIMING CHARACTERISTICS** | | | |
| $t_{SU;DAT1}$ | Data set-up time | ≥ 250ns | > 20 $t_{CLCL}$ – $t_{RD}$ |
| $t_{SU;DAT2}$ | SDA set-up time (before rep. START cond.) | ≥ 250ns | > 1µs[1] |
| $t_{SU;DAT3}$ | SDA set-up time (before STOP cond.) | ≥ 250ns | > 8 $t_{CLCL}$ |
| $t_{HD;DAT}$ | Data hold time | ≥ 0ns | > 8 $t_{CLCL}$ – $t_{FC}$ |
| $t_{SU;STA}$ | Repeated START set-up time | ≥ 14 $t_{CLCL}$ | > 4.7µs[1] |
| $t_{SU;STO}$ | STOP condition set-up time | ≥ 14 $t_{CLCL}$ | > 4.0µs[1] |
| $t_{BUF}$ | Bus free time | ≥ 14 $t_{CLCL}$ | > 4.7µs[1] |
| $t_{RD}$ | SDA rise time | ≤ 1µs | –[2] |
| $t_{FD}$ | SDA fall time | ≤ 0.3µs | < 0.3µs[3] |

**NOTES:**
1. At 100 kbit/s. At other bit rates this value is inversely proportional to the bit-rate of 100 kbit/s.
2. Determined by the external bus-line capacitance and the external bus-line pull-resistor, this must be < 1µs.
3. Spikes on the SDA and SCL lines with a duration of less than 3 $t_{CLCL}$ will be filtered out. Maximum capacitance on bus-lines SDA and SCL = 400pF.
4. $t_{CLCL} = 1/f_{OSC}$ = one oscillator clock period at pin XTAL1. For 63ns (42ns) < $t_{CLCL}$ < 285ns (16MHz (24MHz) > $f_{OSC}$ > 3.5MHz) the SI01 interface meets the I²C-bus specification for bit-rates up to 100 kbit/s.

## TIMING SIO1 (I²C) INTERFACE



## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{PSEN}$ low.

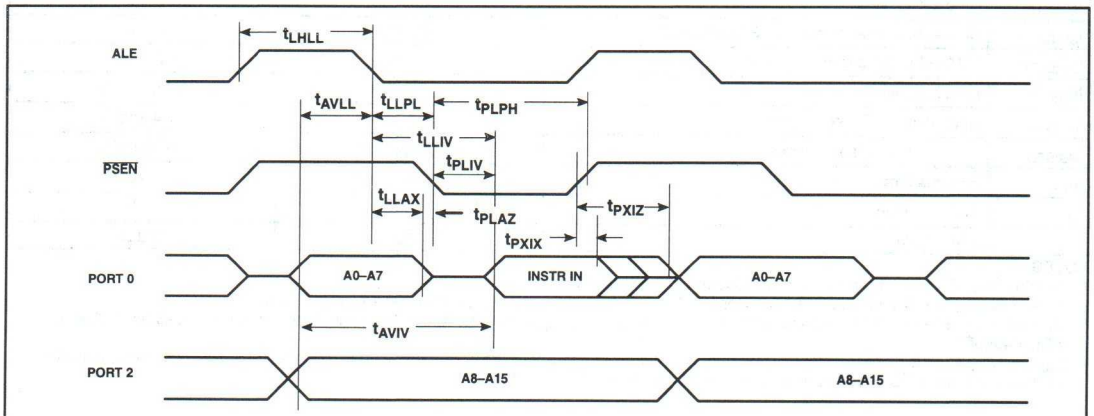



**Figure 2. External Program Memory Read Cycle**

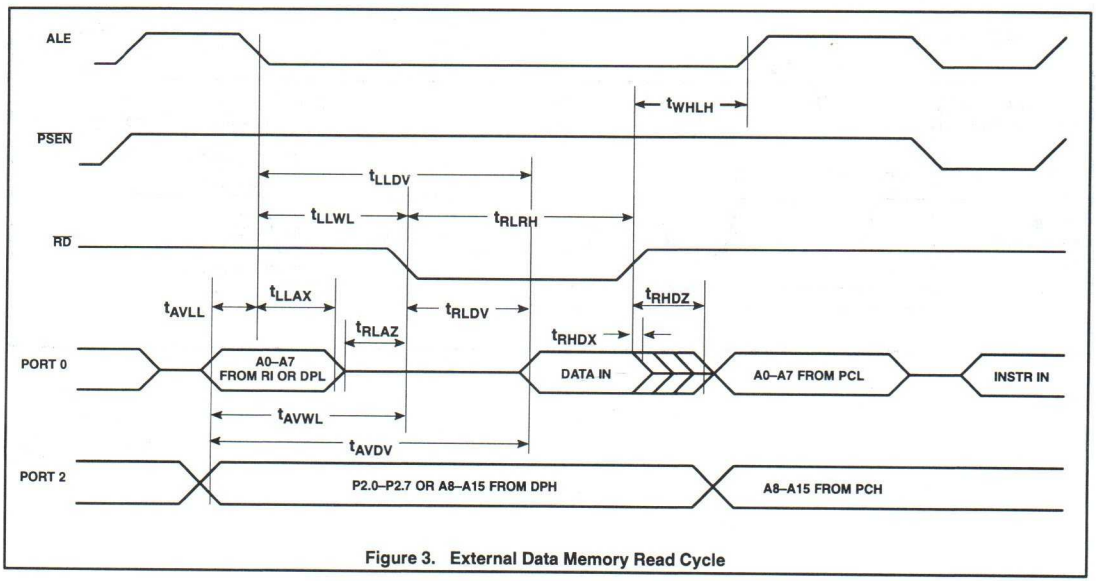



**Figure 3. External Data Memory Read Cycle**

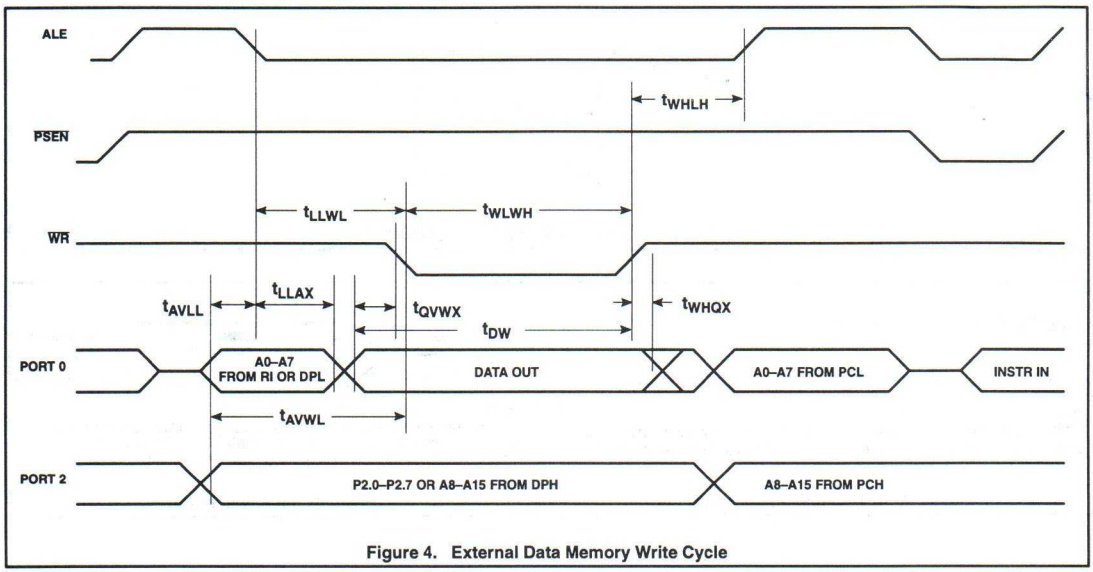



Figure 4. External Data Memory Write Cycle

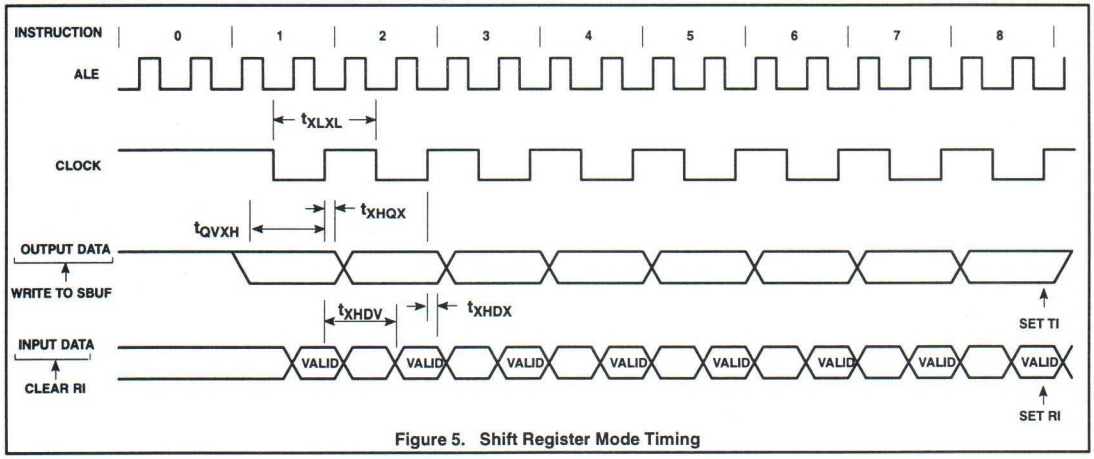



Figure 5. Shift Register Mode Timing

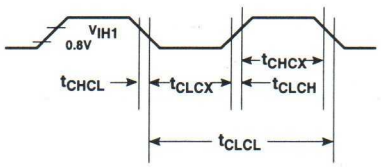
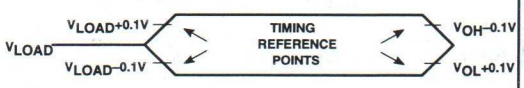



**Figure 6. External Clock Drive at XTAL1**




**NOTE:**
AC INPUTS DURING TESTING ARE DRIVEN AT $V_{DD}$–0.5 FOR A LOGIC '1' AND 0.45V FOR A LOGIC '0'. TIMING MEASUREMENTS ARE MADE AT $V_{IH}$ MIN FOR A LOGIC '1' AND $V_{IL}$ MAX FOR A LOGIC '0'.

**Figure 7. AC Testing Input/Output**




**NOTE:**
FOR TIMING PURPOSES, A PORT IS NO LONGER FLOATING WHEN A 100MV CHANGE FROM LOAD VOLTAGE OCCURS, AND BEGINS TO FLOAT WHEN A 100mV CHANGE FROM THE LOADED $V_{OH}/V_{OL}$ LEVEL OCCURS. $I_{OH}/I_{OL} \geq \pm$ 20mA.

**Figure 8. Float Waveform**

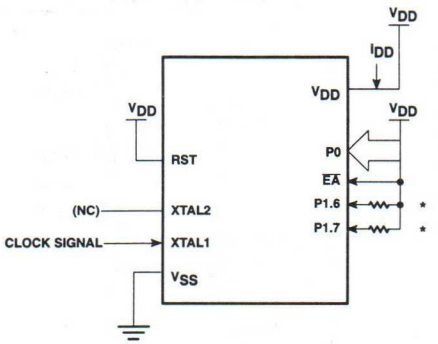



**Figure 9. $I_{DD}$ Test Condition, Active Mode**
**All other pins are disconnected**

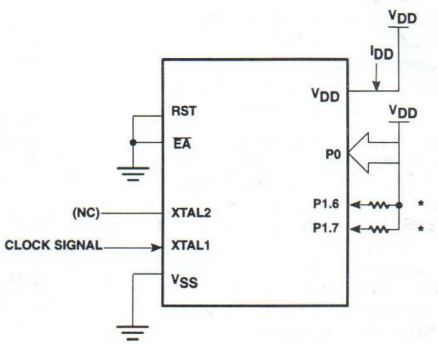



**Figure 10. $I_{DD}$ Test Condition, Idle Mode**
**All other pins are disconnected**

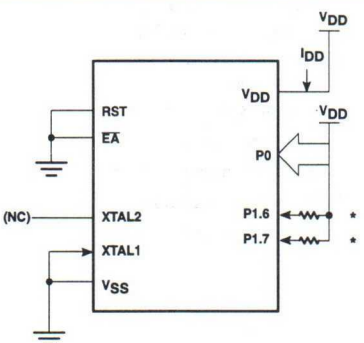



**Figure 11. $I_{DD}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{DD}$ = 2V to 5.5V**

**NOTE:**
* Ports 1.6 and 1.7 should be connected to $V_{CC}$ through resistors of sufficiently high value such that the sink current into these pins does not exceed the $I_{OL1}$ specification.

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# CMOS single-chip 8-bit microcontroller 87C652

## DESCRIPTION

The 87C652 Single-Chip 8-Bit Microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 87C652 has the same instruction set as the 80C51.

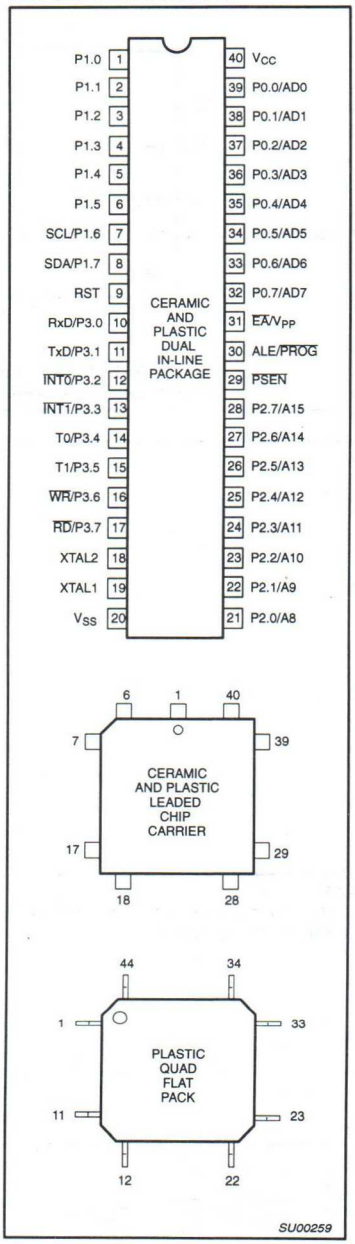
This device provides architectural enhancements that make it applicable in a variety of applications for general control systems. The 87C652 contains a non-volatile 8k × 8 EPROM, a volatile 256 × 8 read/write data memory, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a multi-source, two-priority-level, nested interrupt structure, an I²C interface, UART and on-chip oscillator and timing circuits. For systems that require extra capability, the 87C652 can be expanded using standard TTL compatible memories and logic.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte and 17 three-byte. With a 16MHz crystal, 58% of the instructions are executed in 0.75µs and 40% in 1.5µs. Multiply and divide instructions require 3µs.

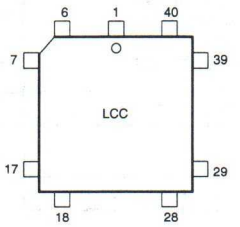
## FEATURES

- 80C51 central processing unit
- 8k × 8 EPROM expandable externally to 64k bytes (EPROM is not expandable)
- 256 × 8 RAM, expandable externally to 64k bytes
- Two standard 16-bit timer/counters
- Four 8-bit I/O ports
- I²C-bus serial I/O port with byte oriented master and slave functions
- Full-duplex UART facilities
- Power control modes
  - Idle mode
  - Power-down mode
- Five package styles
- Extended temperature range
- OTP package available
- Two speed ranges
  - 16MHz
  - 20MHz

## PIN CONFIGURATION



## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

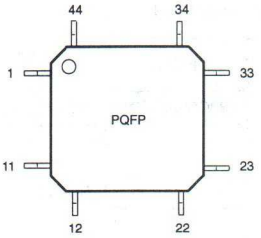



| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC* | 23 | NC8 |
| 2 | P1.0 | 24 | P2.0/A8 |
| 3 | P1.1 | 25 | P2.1/A9 |
| 4 | P1.2 | 26 | P2.2/A10 |
| 5 | P1.3 | 27 | P2.3/A11 |
| 6 | P1.4 | 28 | P2.4/A12 |
| 7 | P1.5 | 29 | P2.5/A13 |
| 8 | P1.6/SCL | 30 | P2.6/A14 |
| 9 | P1.7/SDA | 31 | P2.7/A15 |
| 10 | RST | 32 | $\overline{PSEN}$ |
| 11 | P3.0/RxD | 33 | ALE/$\overline{PROG}$ |
| 12 | NC8 | 34 | NC8 |
| 13 | P3.1/TxD | 35 | $\overline{EA}$/$V_{PP}$ |
| 14 | P3.2/$\overline{INT0}$ | 36 | P0.7/AD7 |
| 15 | P3.3/$\overline{INT1}$ | 37 | P0.6/AD6 |
| 16 | P3.4/T0 | 38 | P0.5/AD5 |
| 17 | P3.5/T1 | 39 | P0.4/AD4 |
| 18 | P3.6/$\overline{WR}$ | 40 | P0.3/AD3 |
| 19 | P3.7/$\overline{RD}$ | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | $V_{SS}$ | 44 | $V_{CC}$ |

* DO NOT CONNECT

SU00260

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS

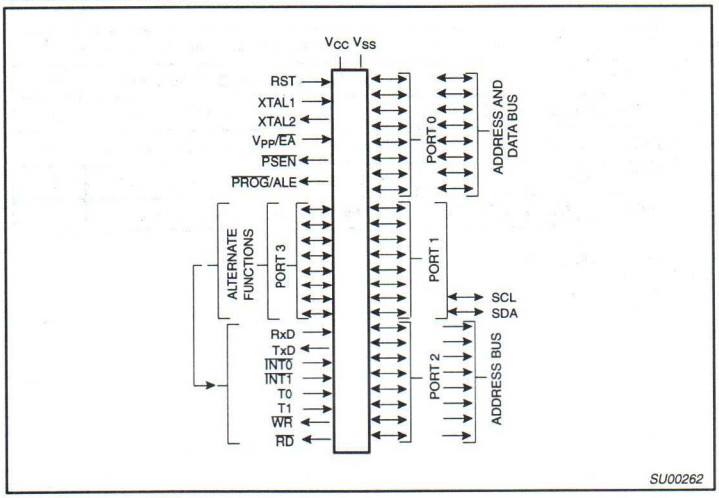



| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5 | 23 | P2.5/A13 |
| 2 | P1.6/SCL | 24 | P2.6/A14 |
| 3 | P1.7/SDA | 25 | P2.7/A15 |
| 4 | RST | 26 | $\overline{PSEN}$ |
| 5 | P3.0/RxD | 27 | ALE/$\overline{PROG}$ |
| 6 | NC* | 28 | NC* |
| 7 | P3.1/TxD | 29 | $\overline{EA}$/$V_{PP}$ |
| 8 | P3.2/$\overline{INT0}$ | 30 | P0.7/AD7 |
| 9 | P3.3/$\overline{INT1}$ | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/$\overline{WR}$ | 34 | P0.3/AD3 |
| 13 | P3.7/$\overline{RD}$ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | $V_{SS}$ | 38 | $V_{CC}$ |
| 17 | NC* | 39 | NC* |
| 18 | P2.0/A8 | 40 | P1.0 |
| 19 | P2.1/A9 | 41 | P1.1 |
| 20 | P2.2/A10 | 42 | P1.2 |
| 21 | P2.3/A11 | 43 | P.13 |
| 22 | P2.4/A12 | 44 | P1.4 |

* DO NOT CONNECT

SU00261

## LOGIC SYMBOL




SU00262

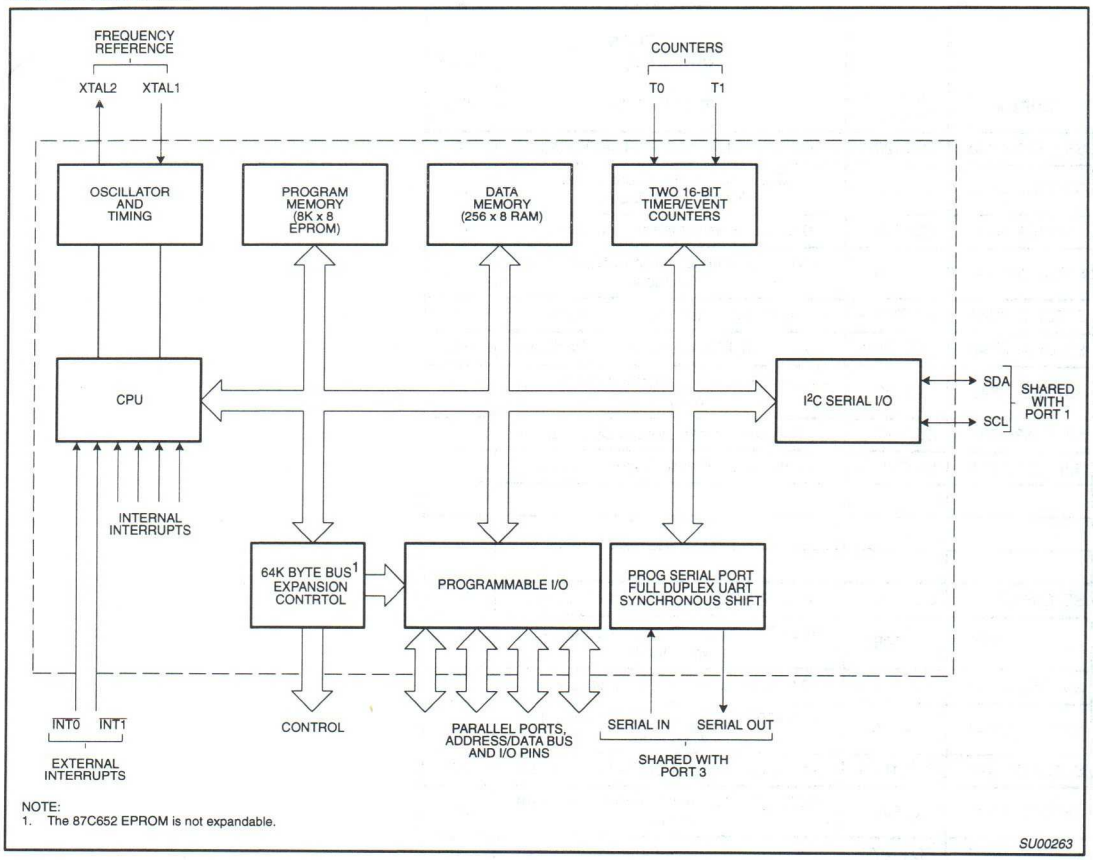
## ORDER INFORMATION

| PHILIPS PART ORDER NUMBER PART MARKING | | PHILIPS NORTH AMERICA PART ORDER NUMBER | | Drawing Number | TEMPERATURE RANGE (°C) AND PACKAGE | FREQ MHz |
|---|---|---|---|---|---|---|
| ROMless | ROM | ROMless | ROM | | | |
| P80C652FBP | P83C652FBP/xxx | P80C652FBPN | P83C652FBPN | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 16 |
| | | | | | | |
| P80C652FBA | P83C652FBA/xxx | P80C652FBAA | P83C652FBAA | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| | | | | | | |
| P80C652FBB | P83C652FBB/xxx | P80C652FBBB | P83C652FBBA | SOT307-2[5] | 0 to +70, Plastic Quad Flat Pack | 16 |
| P80C652FFP | P83C652FFP/xxx | P80C652FFPN | P83C652FFPN | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 16 |
| | | | | | | |
| P80C652FFA | P83C652FFA/xxx | P80C652FFAA | P83C652FFAA | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| P80C652FFB | P83C652FFB/xxx | P80C652FFBB | P83C652FFBB | SOT307-2[5] | –40 to +85, Plastic Quad Flat Pack | 16 |
| P80C652FHP | P83C652FHP/xxx | P80C652FHPN | P83C652FHPN | SOT129-1 | –40 to +125, Plastic Dual In-line Package | 16 |
| P80C652FHA | P83C652FHA/xxx | P80C652FHAA | P83C652FHAA | SOT187-2 | –40 to +125, Plastic Leaded Chip Carrier | 16 |
| P80C652FHB | P83C652FHB/xxx | P80C652FHBB | P83C652FHBB | SOT307-2[5] | –40 to +125, Plastic Quad Flat Pack | 16 |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| P80C652IBP | P83C652IBP/xxx | P80C652IBPN | P83C652IBPN | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 24 |
| P80C652IBA | P83C652IBA/xxx | P80C652IBAA | P83C652IBAA | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 24 |
| P80C652IBB | P83C652IBB/xxx | P80C652IBBB | P83C652IBBB | SOT307-2[5] | 0 to +70, Plastic Quad Flat Pack | 24 |
| P80C652IFP | P83C652IFP/xxx | P80C652IFPN | P83C652IFPN | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 24 |
| P80C652IFA | P83C652IFA/xxx | P80C652IFAA | P83C652IFAA | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 24 |
| P80C652IFB | P83C652IFB/xxx | P80C652IFBB | P83C652IFBB | SOT307-2[5] | –40 to +85, Plastic Quad Flat Pack | 24 |

**NOTES:**
1. 80C652 and 83C652 frequency range is 1.2MHz–16MHz or 1.2 to 24MHz.
2. 87C652 frequency range is 3.5MHz–16MHz or 3.5MHz–20MHz.
3. The 87C652 EPROM is not expandable.
4. xxx denotes the ROM code number.
5. SOT311 replaced by SOT307-2.

| EPROM | Drawing Number | TEMPERATURE RANGE (°C) AND PACKAGE | FREQ MHz |
|---|---|---|---|
| S87C652-4N40 | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 16 |
| S87C652-4F40 | 0590B | 0 to +70, Ceramic Dual In-line Package w/Window | 16 |
| S87C652-4A44 | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| S87C652-4K44 | 1472A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 16 |
| S87C652-4B44 | SOT307-2 | 0 to +70, Plastic Quad Flat Pack | 16 |
| S87C652-5N40 | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 16 |
| S87C652-5F40 | 0590B | –40 to +85, Ceramic Dual In-line Package w/Window | 16 |
| S87C652-5A44 | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| S87C652-5B44 | SOT307-2 | –40 to +85, Plastic Quad Flat Pack | 16 |
| | | | |
| | | | |
| | | | |
| S87C652-7N40 | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 20 |
| S87C652-7F40 | 0590B | 0 to +70, Ceramic Dual In-line Package w/Window | 20 |
| S87C652-7A44 | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 20 |
| S87C652-7K44 | 1472A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 20 |
| S87C652-8N40 | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 20 |
| S87C652-8F40 | 0590B | –40 to +85, Ceramic Dual In-line Package w/Window | 20 |
| S87C652-8A44 | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 20 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

## BLOCK DIAGRAM




NOTE:
1. The 87C652 EPROM is not expandable.

SU00263

## PIN DESCRIPTION

| MNEMONIC | PIN NO. DIP | PIN NO. LCC | PIN NO. QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification in the 87C652. External pull-ups are required during program verification. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups, except P1.6 and P1.7 which are open drain. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also receives the low-order address byte during program memory verification. Alternate functions include: |
| P1.6 | 7 | 8 | 2 | I/O | **SCL:** $I^2C$-bus serial port clock line. |
| P1.7 | 8 | 9 | 3 | I/O | **SDA:** $I^2C$-bus serial port data line. |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{CC}$. |
| ALE/$\overline{PROG}$ | 30 | 33 | 27 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the 87C652 is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}/V_{PP}$ | 31 | 35 | 29 | I | **External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H and 1FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 3FFFH. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTE:**
To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher than $V_{CC}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

**Table 1. 8XC652/654 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes) | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 | 0x000000B |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP*# | Interrupt priority | B8H | – | | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 | xx000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1*# | Port 1 | 90H | SDA | SCL | | | | | | | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TXD | RXD | FFH |
| PCON | Power control | 87H | SMOD | – | – | – | GF1 | GF0 | PD | IDL | 0xxx0000B |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| S0CON*# | Serial 0 port control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| S0BUF# | Serial 0 data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | F1 | P | 00H |
| S1DAT# | Serial 1 data | DAH | | | | | | | | | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| S1ADR# | Serial 1 address | DBH | SLAVE ADDRESS | | | | | | | GC | 00H |
| | | | | | | | | | | | |
| S1STA# | Serial 1 status | D9H | SC4 | SC3 | SC2 | SC1 | SC0 | 0 | 0 | 0 | F8H |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| S1CON*# | Serial 1 control | D8H | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 | 00000000B |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| TH1 | Timer high 1 | 8DH | | | | | | | | | 00H |
| TH0 | Timer high 0 | 8CH | | | | | | | | | 00H |
| TL1 | Timer low 1 | 8BH | | | | | | | | | 00H |
| TL0 | Timer low 0 | 8AH | | | | | | | | | 00H |
| TMOD | Timer mode | 89H | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the logic symbol, page 3-1011.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up.

## IDLE MODE

In the idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. the control bits for the reduced power modes are in the special function register PCON. Table 2 shows the state of the I/O ports during low current operating modes.

## I²C SERIAL COMMUNICATION — SIO1

The I²C serial port is identical to the I²C serial port on the 8XC552. The operation of this subsystem is described in detail in the 8XC552 section of this manual.

Note that in both the 8XC652/4 and the 8XC552 the I²C pins are alternate functions to port pins P1.6 and P1.7. Because of this, P1.6 and P1.7 on these parts do not have a pull-up structure as found on the 80C51. Therefore P1.6 and P1.7 have open drain outputs on the 8XC652/4.

**Table 2. External Pin Status During Idle and Power-Down Mode**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

### Serial Control Register (S1CON) – See Table 3

**S1CON (D8H)**

| CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |
|---|---|---|---|---|---|---|---|

**Bits CR0, CR1 and CR2 determine the serial clock frequency that is generated in the master mode of operation.**

**Table 3. Serial Clock Rates**

| CR2 | CR1 | CR0 | BIT FREQUENCY (kHz) AT $f_{OSC}$ 6MHz | 12MHz | 16MHz | 20MHz | $f_{OSC}$ DIVIDED BY |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 23 | 47 | 62.5 | 78 | 256 |
| 0 | 0 | 1 | 27 | 54 | 71 | 89 | 224 |
| 0 | 1 | 0 | 31.25 | 62.5 | 83.3 | 104[1] | 192 |
| 0 | 1 | 1 | 37 | 75 | 100 | 125[1] | 160 |
| 1 | 0 | 0 | 6.25 | 12.5 | 17 | 21 | 960 |
| 1 | 0 | 1 | 50 | 100 | 133[1] | 166[1] | 120 |
| 1 | 1 | 0 | 100 | 200[1] | 267[1] | 334[1] | 60 |
| 1 | 1 | 1 | > 0.25 < 62.5<br>0 to 255 | > 0.5 < 62.5<br>0 to 254 | > 0.65 < 55.6<br>0 to 253 | > 0.81 < 69.4<br>0 to 253 | 96 × (256 – (reload value Timer 1))<br>(Reload value range: 0 – 254 in mode 2) |

**NOTES:**
1. These frequencies exceed the upper limit of 100kHz of the I²C-bus specification and cannot be used in an I²C-bus application.

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ to $V_{SS}$ | –0.5 to + 13 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to + 6.5 | V |
| Input, output current on any single pin | ±5 | mA |
| Input, output current on any two pins | ±10 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DEVICE SPECIFICATIONS

| TYPE | SUPPLY VOLTAGE (V) | | FREQUENCY (MHz) | | TEMPERATURE (°C) |
|---|---|---|---|---|---|
| | MIN. | MAX. | MIN. | MAX. | |
| S87C652–4 | 4.5 | 5.5 | 3.5 | 16 | 0 to +70 |
| S87C652–5 | 4.5 | 5.5 | 3.5 | 16 | –40 to +85 |
| S87C652–7 | 4.5 | 5.5 | 3.5 | 20 | 0 to +70 |
| S87C652–8 | 4.5 | 5.5 | 3.5 | 20 | –40 to +85 |

## DC ELECTRICAL CHARACTERISTICS

$V_{SS}$ = 0V

| SYMBOL | PARAMETER | PART TYPE | TEST CONDITIONS | LIMITS MIN. | LIMITS MAX. | UNIT |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input low voltage,<br>except $\overline{EA}$, P1.6/SCL, P1.7/SDA | 0°C to +70°C<br>–40°C to +85°C | | –0.5<br>–0.5 | $0.2V_{CC} - 0.1$<br>$0.2V_{CC} - 0.15$ | V<br>V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | 0°C to +70°C<br>–40°C to +85°C | | –0.5<br>–0.5 | $0.2V_{CC} - 0.3$<br>$0.2V_{CC} - 0.35$ | V<br>V |
| $V_{IL2}$ | Input low voltage to P1.6/SCL, P1.7/SDA[6] | | | –0.5 | 1.5 | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST,<br>P1.6/SCL, P1.7/SDA | 0°C to +70°C<br>–40°C to +85°C | | $0.2V_{CC} + 0.9$<br>$0.2V_{CC} + 1.0$ | $V_{CC} + 0.5$<br>$V_{CC} + 0.5$ | V<br>V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | 0°C to +70°C<br>–40°C to +85°C | | $0.7V_{CC}$<br>$0.7V_{CC} + 0.1$ | $V_{CC} + 0.5$<br>$V_{CC} + 0.5$ | V<br>V |
| $V_{IH2}$ | Input high voltage, P1.6/SCL, P1.7/SDA[6] | | | 3.0 | 6.0 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3,<br>except P1.6/SCL, P1.7/SDA | | $I_{OL}$ = 1.6mA[8] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$ | | $I_{OL}$ = 3.2mA[8] | | 0.45 | V |
| $V_{OL2}$ | Output low voltage, P1.6/SCL, P1.7/SDA | | $I_{OL}$ = 3.0mA | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3 | 0°C to +70°C<br>–40°C to +85°C | $I_{OH}$ = –60µA<br>$I_{OH}$ = –25µA | 2.4<br>$0.75V_{CC}$ | | V<br>V |
| $V_{OH1}$ | Output high voltage, Port 0 in external bus<br>mode, ALE, $\overline{PSEN}$, RST[9] | 0°C to +70°C<br>–40°C to +85°C | $I_{OH}$ = –400µA<br>$I_{OH}$ = –150µA | 2.4<br>$0.75V_{CC}$ | | V<br>V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3, except<br>P1.6/SCL, P1.7/SDA | 0°C to +70°C<br>–40°C to +85°C | $V_{IN}$ = 0.45V | | –50<br>–75 | µA<br>µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2,<br>3, except P1.6/SCL, P1.7/SDA | 0°C to +70°C<br>–40°C to +85°C | See Note 7 | | –650<br>–750 | µA<br>µA |
| $I_{L1}$ | Input leakage current, port 0 | | 0.45 < $V_i$ < $V_{CC}$ | | ±10 | µA |
| $I_{L2}$ | Input leakage current, P1.6/SCL, P1.7/SDA | | 0V < $V_i$ < 6.0V<br>0V < $V_{CC}$ < 6.0V | | ±10 | µA<br>µA |
| $I_{CC}$ | Power supply current:<br><br>Active mode @ 16MHz[2]<br>Idle mode @ 16MHz[3]<br>Power down mode [4, 5]<br>Power down mode [4, 5] | <br><br><br><br>0°C to +70°C<br>–40°C to +125°C | See Note 1<br>$V_{CC}$ = 6.0V | | <br><br>25<br>6<br>50<br>135 | <br><br>mA<br>mA<br>µA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | | 50 | 150 | kΩ |
| $C_{IO}$ | Pin Capacitance | | Freq. = 1MHz | | 10 | pF |

**NOTES:**

1. See Figures 9 through 12 for $I_{CC}$ test conditions.
2. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 10ns; $V_{IL} = V_{SS}$ + 0.5V; $V_{IH} = V_{CC}$ –0.5V; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = P1.6 = P1.7 = $V_{CC}$; $f_{CLK}$ = 16MHz. See Figure 9.
3. The idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 10ns; $V_{IL} = V_{SS}$ + 0.5V; $V_{IH} = V_{CC}$ –0.5V; XTAL2 not connected; Port 0 = P1.6 = P1.7 = $V_{CC}$; $\overline{EA}$ = RST = $V_{SS}$; $f_{CLK}$ = 16MHz. See Figure 10.
4. The power-down current is measured with all output pins disconnected; XTAL2 not connected; Port 0 = P1.6 = P1.7 = $V_{CC}$; $\overline{EA}$ = RST = $V_{SS}$. See Figure 12.
5. 2V ≤ $V_{PD}$ ≤ $V_{CC}$max.
6. The input threshold voltage of P1.6 and P1.7 (SIO1) meets the $I^2C$ specification, so an input voltage below 1.5V will be recognized as a logic 0 while an input voltage above 3.0V will be recognized as a logic 1.
7. Pins of ports 1 , 2, and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
8. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
9. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, or $T_{amb}$ = –40°C to +85°C, $V_{SS}$ = 0V[1, 2]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 1 | Oscillator frequency | | | 3.5 | 16 | MHz |
| $t_{LHLL}$ | 1 | ALE pulse width | 85 | | $2t_{CLCL}$–40 | | ns |
| $t_{AVLL}$ | 1 | Address valid to ALE low | 8 | | $t_{CLCL}$–55 | | ns |
| $t_{LLAX}$ | 1 | Address hold after ALE low | 28 | | $t_{CLCL}$–35 | | ns |
| $t_{LLIV}$ | 1 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}$–100 | ns |
| $t_{LLPL}$ | 1 | ALE low to $\overline{PSEN}$ low | 23 | | $t_{CLCL}$–40 | | ns |
| $t_{PLPH}$ | 1 | $\overline{PSEN}$ pulse width | 143 | | $3t_{CLCL}$–45 | | ns |
| $t_{PLIV}$ | 1 | $\overline{PSEN}$ low to valid instruction in | | 83 | | $3t_{CLCL}$–105 | ns |
| $t_{PXIX}$ | 1 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 1 | Input instruction float after $\overline{PSEN}$ | | 38 | | $t_{CLCL}$–25 | ns |
| $t_{AVIV}$ | 1 | Address to valid instruction in | | 208 | | $5t_{CLCL}$–105 | ns |
| $t_{PLAZ}$ | 1 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{AVLL}$ | 2, 3 | Address valid to ALE low | 28 | | $t_{CLCL}$–35 | | ns |
| $t_{RLRH}$ | 2, 3 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}$–100 | | ns |
| $t_{WLWH}$ | 2, 3 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}$–100 | | ns |
| $t_{RLDV}$ | 2, 3 | $\overline{RD}$ low to valid data in | | 148 | | $5t_{CLCL}$–165 | ns |
| $t_{RHDX}$ | 2, 3 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 2, 3 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}$–70 | ns |
| $t_{LLDV}$ | 2, 3 | ALE low to valid data in | | 350 | | $8t_{CLCL}$–150 | ns |
| $t_{AVDV}$ | 2, 3 | Address to valid data in | | 398 | | $9t_{CLCL}$–165 | ns |
| $t_{LLWL}$ | 2, 3 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 138 | 238 | $3t_{CLCL}$–50 | $3t_{CLCL}$+50 | ns |
| $t_{AVWL}$ | 2, 3 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 120 | | $4t_{CLCL}$–130 | | ns |
| $t_{QVWX}$ | 2, 3 | Data valid to $\overline{WR}$ transition | 3 | | $t_{CLCL}$–60 | | ns |
| $t_{DW}$ | 2, 3 | Data setup time before $\overline{WR}$ | 288 | | $7t_{CLCL}$–150 | | ns |
| $t_{WHQX}$ | 2, 3 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}$–50 | | ns |
| $t_{RLAZ}$ | 2, 3 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 2, 3 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}$–40 | $t_{CLCL}$+40 | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 4 | Serial port clock cycle time[4] | 0.75 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 4 | Output data setup to clock rising edge[4] | 492 | | $10t_{CLCL}$–133 | | ns |
| $t_{XHQX}$ | 4 | Output data hold after clock rising edge[4] | 8 | | $2t_{CLCL}$–117 | | ns |
| $t_{XHDX}$ | 4 | Input data hold after clock rising edge[4] | 0 | | 0 | | ns |
| $t_{XHDV}$ | 4 | Clock rising edge to input data valid[4] | | 498 | | $10t_{CLCL}$–133 | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 5 | High time[4] | 20 | | 20 | $t_{CLCL} - t_{LOW}$ | ns |
| $t_{CLCX}$ | 5 | Low time[4] | 20 | | 20 | $t_{CLCL} - t_{HIGH}$ | ns |
| $t_{CLCH}$ | 5 | Rise time[4] | | 20 | | 20 | ns |
| $t_{CHCL}$ | 5 | Fall time[4] | | 20 | | 20 | ns |

## AC ELECTRICAL CHARACTERISTICS (Continued)

$T_{amb}$ = 0°C to +70°C, or $T_{amb}$ = –40°C to +85°C, $V_{SS}$ = 0V[1, 2]

| SYMBOL | FIGURE | PARAMETER | 20MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 1 | Oscillator frequency | | | 3.5 | 20 | MHz |
| $t_{LHLL}$ | 1 | ALE pulse width | 60 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 1 | Address valid to ALE low | 25 | | $t_{CLCL}-25$ | | ns |
| $t_{LLAX}$ | 1 | Address hold after ALE low | 25 | | $t_{CLCL}-25$ | | ns |
| $t_{LLIV}$ | 1 | ALE low to valid instruction in | | 135 | | $4t_{CLCL}-65$ | ns |
| $t_{LLPL}$ | 1 | ALE low to $\overline{PSEN}$ low | 25 | | $t_{CLCL}-25$ | | ns |
| $t_{PLPH}$ | 1 | $\overline{PSEN}$ pulse width | 105 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 1 | $\overline{PSEN}$ low to valid instruction in | | 90 | | $3t_{CLCL}-60$ | ns |
| $t_{PXIX}$ | 1 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 1 | Input instruction float after $\overline{PSEN}$ | | 25 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 1 | Address to valid instruction in | | 170 | | $5t_{CLCL}-80$ | ns |
| $t_{PLAZ}$ | 1 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{AVLL}$ | 2, 3 | Address valid to ALE low | 25 | | $t_{CLCL}-25$ | | ns |
| $t_{RLRH}$ | 2, 3 | $\overline{RD}$ pulse width | 200 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 2, 3 | $\overline{WR}$ pulse width | 200 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 2, 3 | $\overline{RD}$ low to valid data in | | 160 | | $5t_{CLCL}-90$ | ns |
| $t_{RHDX}$ | 2, 3 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 2, 3 | Data float after $\overline{RD}$ | | 72 | | $2t_{CLCL}-28$ | ns |
| $t_{LLDV}$ | 2, 3 | ALE low to valid data in | | 250 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 2, 3 | Address to valid data in | | 285 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 2, 3 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 100 | 200 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 2, 3 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 125 | | $4t_{CLCL}-75$ | | ns |
| $t_{QVWX}$ | 2, 3 | Data valid to $\overline{WR}$ transition | 20 | | $t_{CLCL}-30$ | | ns |
| $t_{DW}$ | 2, 3 | Data setup time before $\overline{WR}$ | 220 | | $7t_{CLCL}-130$ | | ns |
| $t_{WHQX}$ | 2, 3 | Data hold after $\overline{WR}$ | 25 | | $t_{CLCL}-25$ | | ns |
| $t_{RLAZ}$ | 2, 3 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 2, 3 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 25 | 75 | $t_{CLCL}-25$ | $t_{CLCL}+25$ | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 4 | Serial port clock cycle time[4] | 0.6 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 4 | Output data setup to clock rising edge[4] | 367 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 4 | Output data hold after clock rising edge[4] | 40 | | $2t_{CLCL}-60$ | | ns |
| $t_{XHDX}$ | 4 | Input data hold after clock rising edge[4] | 0 | | 0 | | ns |
| $t_{XHDV}$ | 4 | Clock rising edge to input data valid[4] | | 367 | | $10t_{CLCL}-133$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 5 | High time[4] | 17 | | 17 | $t_{CLCL}-t_{LOW}$ | ns |
| $t_{CLCX}$ | 5 | Low time[4] | 17 | | 17 | $t_{CLCL}-t_{HIGH}$ | ns |
| $t_{CLCH}$ | 5 | Rise time[4] | | 20 | | 20 | ns |
| $t_{CHCL}$ | 5 | Fall time[4] | | 20 | | 20 | ns |

## AC ELECTRICAL CHARACTERISTICS (Continued)

$T_{amb}$ = 0°C to +70°C, or $T_{amb}$ = –40°C to +85°C, $V_{SS}$ = 0V[1, 2]

| SYMBOL | PARAMETER | INPUT | OUTPUT |
|---|---|---|---|
| **I²C Interface** | | | |
| $t_{HD;STA}$ | START condition hold time | ≥ 14 $t_{CLCL}$ | > 4.0µs [4] |
| $t_{LOW}$ | SCL low time | ≥ 16 $t_{CLCL}$ | > 4.7µs [4] |
| $t_{HIGH}$ | SCL high time | ≥ 14 $t_{CLCL}$ | > 4.0µs [4] |
| $t_{RC}$ | SCL rise time | ≤ 1µs | – [5] |
| $t_{FC}$ | SCL fall time | ≤ 0.3µs | < 0.3µs [6] |
| $t_{SU;DAT1}$ | Data set-up time | ≥ 250ns | > 20 $t_{CLCL}$ – $t_{RD}$ |
| $t_{SU;DAT2}$ | SDA set-up time (before rep. START cond.) | ≥ 250ns | > 1µs [4] |
| $t_{SU;DAT3}$ | SDA set-up time (before STOP cond.) | ≥ 250ns | > 8 $t_{CLCL}$ |
| $t_{HD;DAT}$ | Data hold time | ≥ 0ns | > 8 $t_{CLCL}$ – $t_{FC}$ |
| $t_{SU;STA}$ | Repeated START set-up time | ≥ 14 $t_{CLCL}$ [4] | > 4.7µs [4] |
| $t_{SU;STO}$ | STOP condition set-up time | ≥ 14 $t_{CLCL}$ [4] | > 4.0µs [4] |
| $t_{BUF}$ | Bus free time | ≥ 14 $t_{CLCL}$ [4] | > 4.7µs [4] |
| $t_{RD}$ | SDA rise time | ≤ 1µs[7] | – [5] |
| $t_{FD}$ | SDA fall time | ≤ 300ns[7] | < 0.3µs [6] |

**NOTES:**

1. Parameters are valid over operating temperature range and voltage range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. These values are characterized but not 100% production tested.
4. At 100 kbit/s. At other bit rates this value is inversely proportional to the bit-rate of 100 kbit/s.
5. Determined by the external bus-line capacitance and the external bus-line pull-resistor, this must be < 1µs.
6. Spikes on the SDA and SCL lines with a duration of less than 3 $t_{CLCL}$ will be filtered out. Maximum capacitance on bus-lines SDA and SCL = 400pF.
7. $t_{CLCL} = 1/f_{OSC}$ = one oscillator clock period at pin XTAL1. For 63ns < $t_{CLCL}$ < 285ns (16MHz > $f_{OSC}$ > 3.5MHz) the I²C interface meets the I²C-bus specification for bit-rates up to 100 kbit/s.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{\text{PSEN}}$
Q – Output data
R – $\overline{\text{RD}}$ signal
t – Time
V – Valid
W – $\overline{\text{WR}}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{\text{PSEN}}$ low.

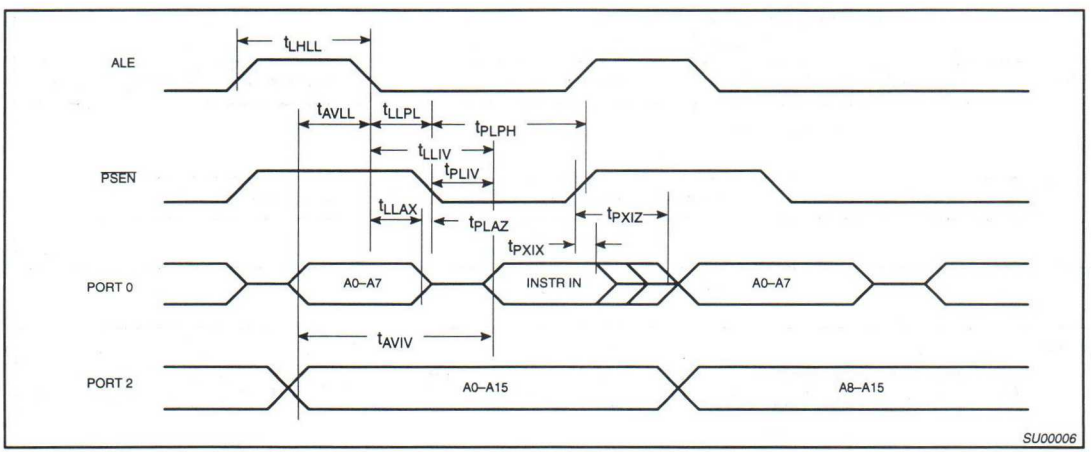



**Figure 1. External Program Memory Read Cycle**

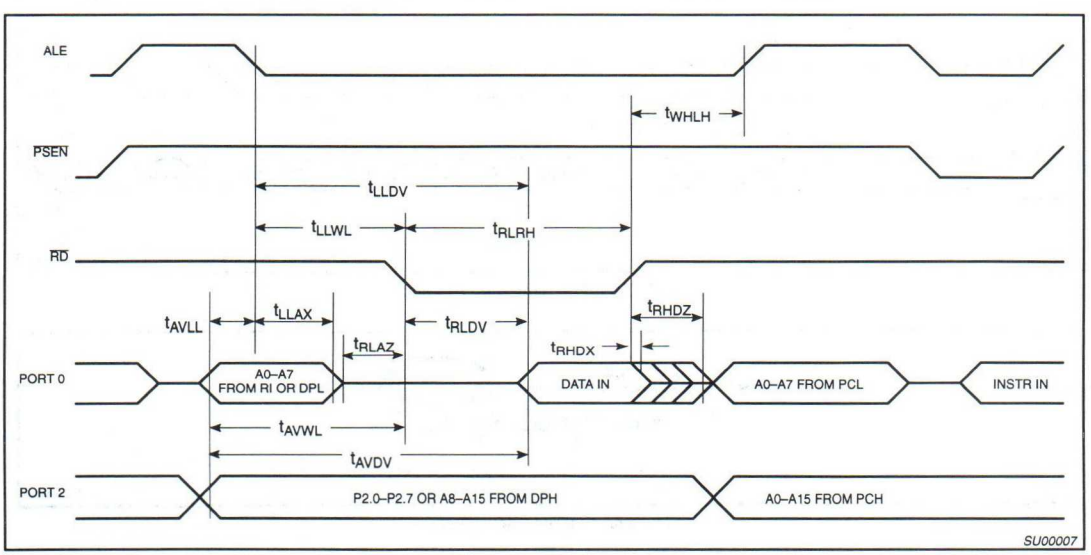



**Figure 2. External Data Memory Read Cycle**

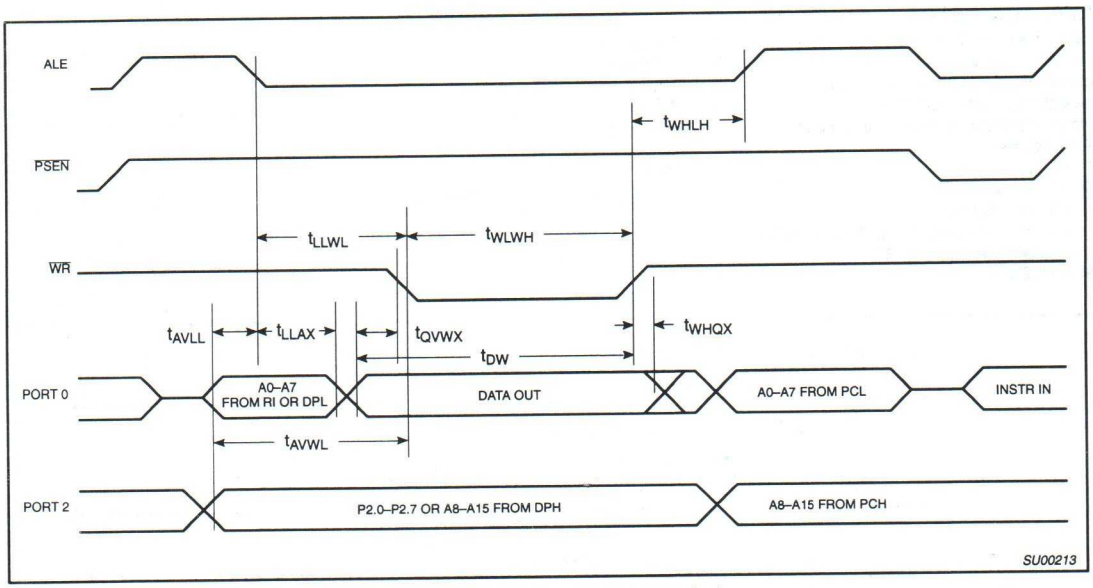



**Figure 3. External Data Memory Write Cycle**

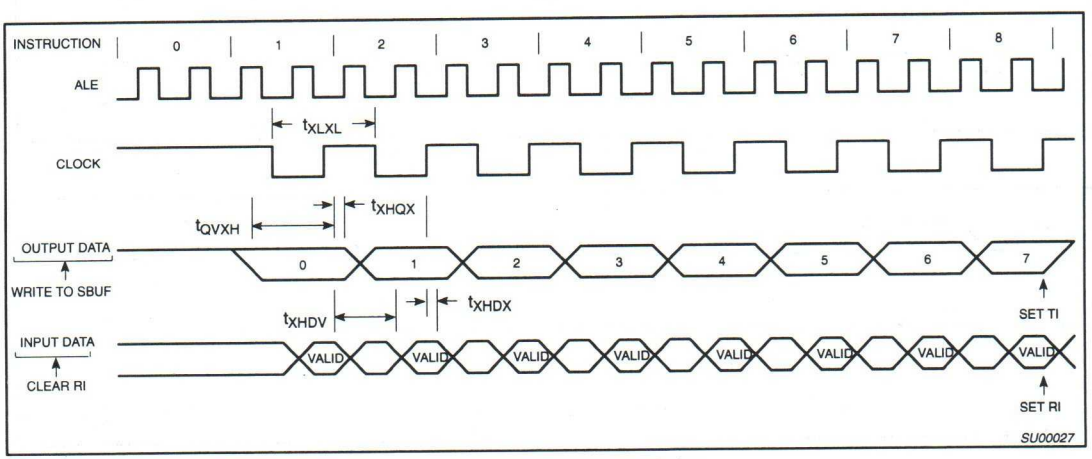



**Figure 4. Shift Register Mode Timing**

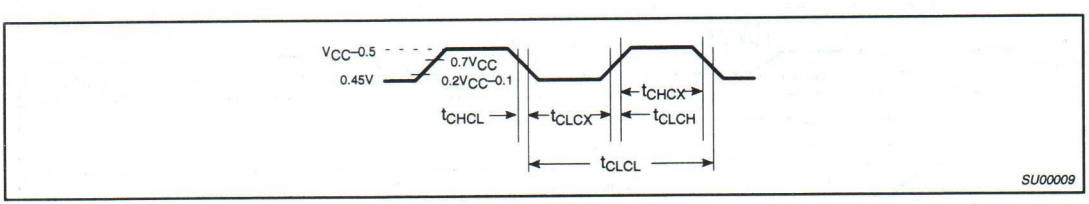



**Figure 5. External Clock Drive**

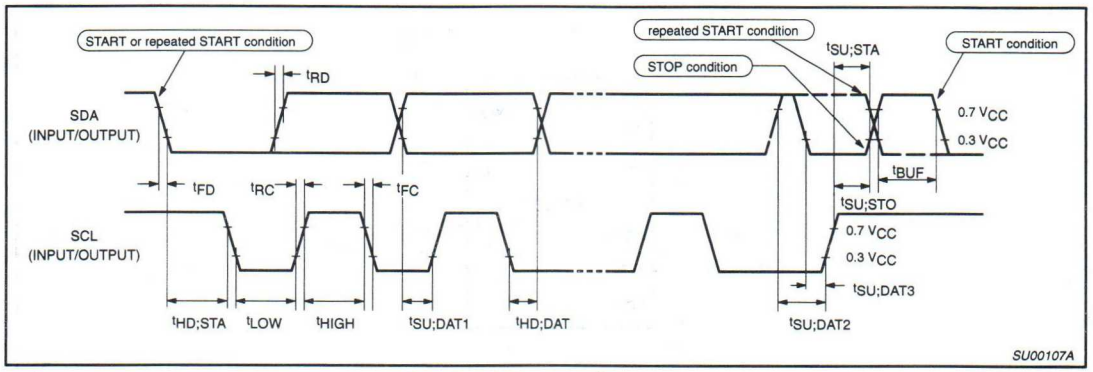



**Figure 6. Timing SIO1 ($I^2C$) Interface**

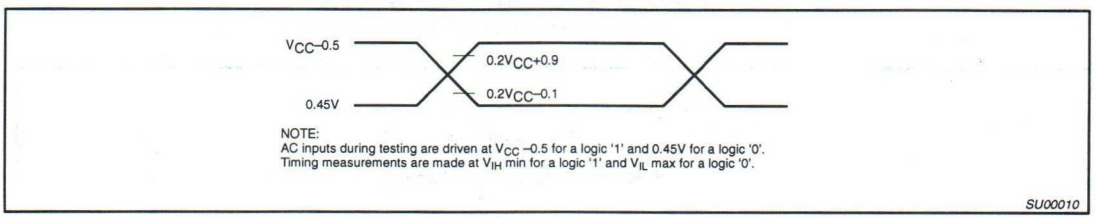



**Figure 7. AC Testing Input/Output**

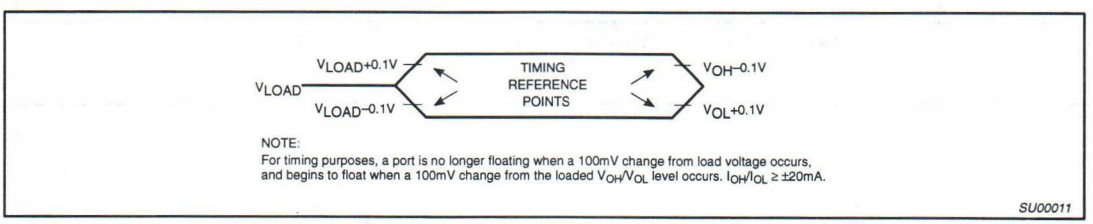



**Figure 8. Float Waveform**

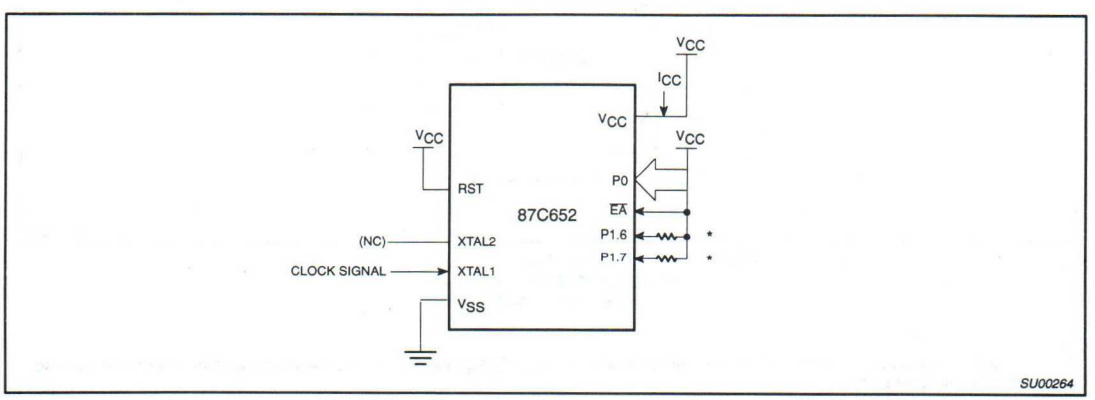



**Figure 9. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

**NOTE:**
* Ports 1.6 and 1.7 should be connected to $V_{CC}$ through resistors of sufficiently high value such that the sink current into these pins does not exceed the $I_{OL1}$ specification.

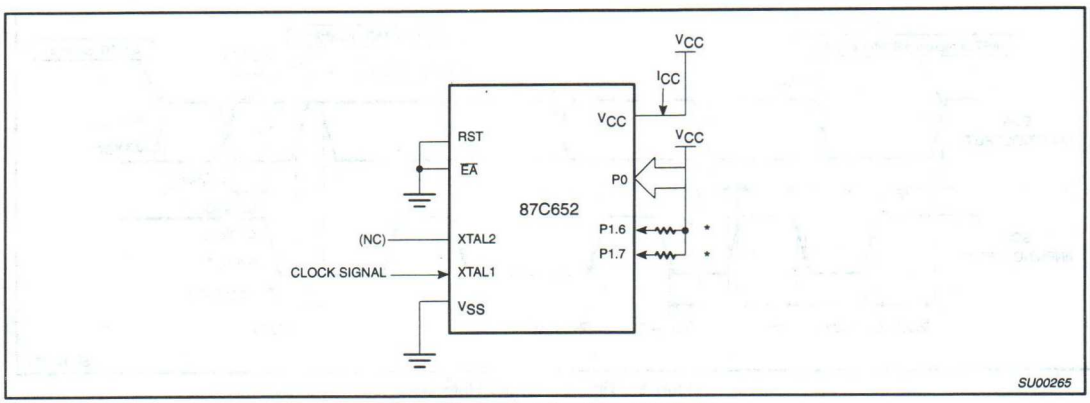



**Figure 10. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

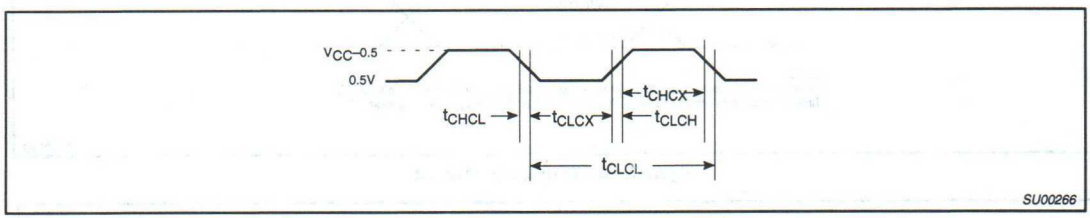



**Figure 11. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCL} = t_{CHCL} = 10ns$**

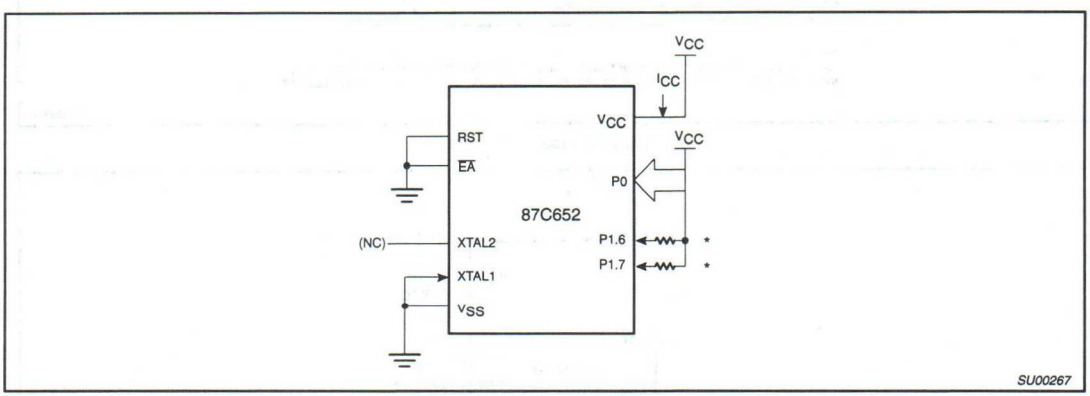



**Figure 12. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected**
**$V_{CC}$ = 2V to 5.5V**

**NOTE:**
* Ports 1.6 and 1.7 should be connected to $V_{CC}$ through resistors of sufficiently high value such that the sink current into these pins does not exceed the $I_{OL1}$ specification.

## EPROM CHARACTERISTICS

The 87C652 is programmed by using a modified Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C652 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C652 manufactured by Philips Components.

Table 4 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the lock bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 13 and 14. Figure 15 shows the circuit configuration for normal program memory verification.

## Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 13. Note that the 87C652 is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1 and 2, as shown in Figure 13. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 3 specified in Table 4 are held at the 'Program Code Data' levels indicated in Table 4. The ALE/$\overline{PROG}$ is pulsed low 25 times as shown in Figure 14.

To program the encryption table, repeat the 25 pulse programming sequence for addresses 0 through 1FH, using the "Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the lock bits, repeat the 25 pulse programming sequence using the 'Pgm Lock Bit' levels. After one lock bit is programmed, further programming of the code memory and encryption table is disabled. However, the other lock bit can still be programmed.

Note that the $\overline{EA}$/$V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

### Program Verification

If lock bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 1 and 2 as shown in Figure 15. The other pins are held at the 'Verify Code Data' levels indicated in Table 4. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

### Reading the Signature Bytes

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:

(030H) = 5H indicates manufactured by Philips

(031H) = 99H indicates 87C652

## Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 4, and which satisfies the timing specifications, is suitable.

## Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-sec/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000uW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient. Erasure leaves the array in an all 1s state.

**Table 4. EPROM Programming Modes**

| MODE | RST | $\overline{PSEN}$ | ALE/$\overline{PROG}$ | $\overline{EA}$/$V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm lock bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm lock bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |

**NOTES:**
1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V±10% during programming and verification.

* ALE/$\overline{PROG}$ receives 25 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

™Trademark phrase of Intel Corporation.

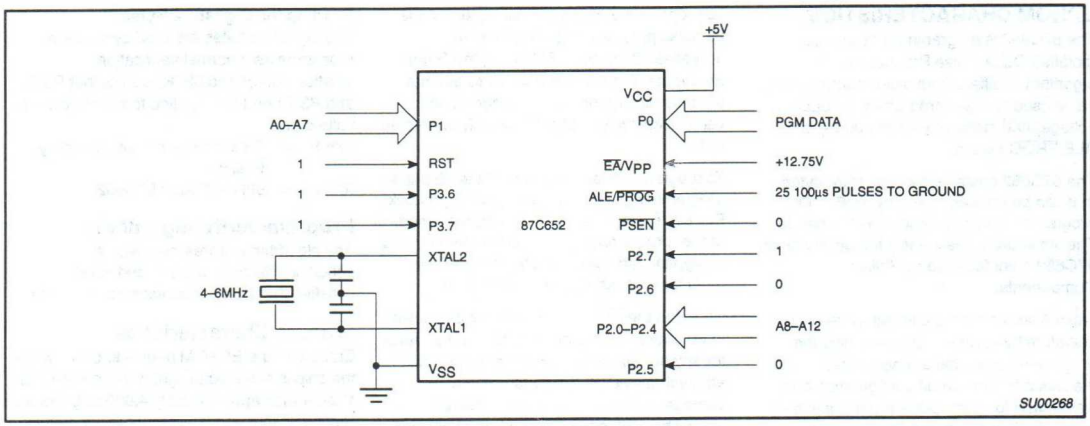



**Figure 13. Programming Configuration**

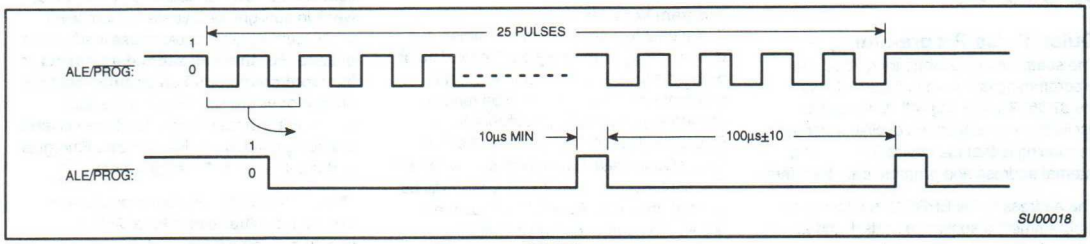



**Figure 14. PROG Waveform**

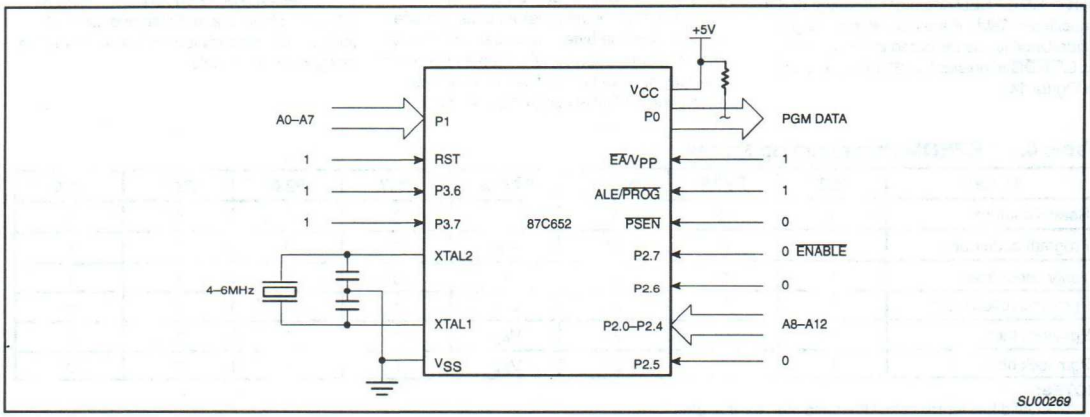
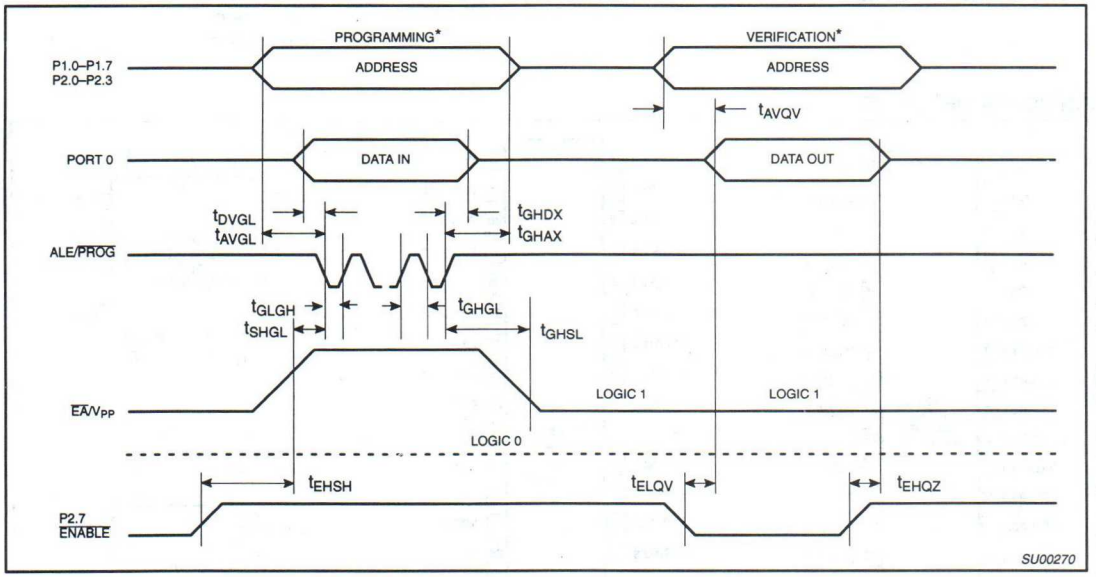



**Figure 15. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (see Figure 16)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




FOR PROGRAMMING VERIFICATION SEE FIGURE 13.
FOR VERIFICATION CONDITIONS SEE FIGURE 15.

**Figure 16. EPROM Programming and Verification**

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# CMOS single-chip 8-bit microcontroller — 83C654

## DESCRIPTION

The P83C654 Single-Chip 8-Bit Microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 83C654 has the same instruction set as the 80C51. Two versions of the derivative exist:

83C654 — 16k bytes mask programmable ROM

87C654 — EPROM version (described in a separate data sheet)

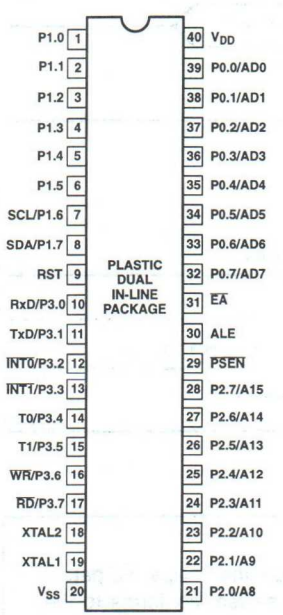
This device provides architectural enhancements that make it applicable in a variety of applications for general control systems. The 83C654 contains a non-volatile 16k × 8 read-only program memory, a volatile 256 × 8 read/write data memory, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a multi-source, two-priority-level, nested interrupt structure, an I²C interface, UART and on-chip oscillator and timing circuits. For systems that require extra capability, the 8XC654 can be expanded using standard TTL compatible memories and logic.

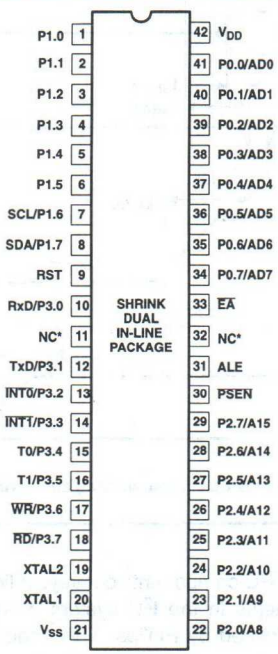
The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte and 17 three-byte. With a 16(24)MHz crystal, 58% of the instructions are executed in 0.75(0.5)µs and 40% in 1.5(1)µs. Multiply and divide instructions require 3(2)µs.

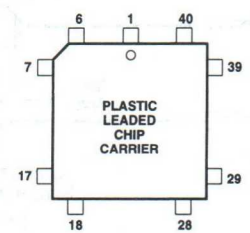
## FEATURES

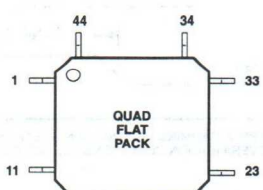
- 80C51 central processing unit
- 16k × 8 ROM expandable externally to 64k bytes
- 256 × 8 RAM, expandable externally to 64k bytes
- Two standard 16-bit timer/counters
- Four 8-bit I/O ports
- I²C-bus serial I/O port with byte oriented master and slave functions
- Full-duplex UART facilities
- Power control modes
  - Idle mode
  - Power-down mode
- ROM code protection
- Extended frequency range: 1.2 to 24 MHz
- Three operating ambient temperature ranges:
  - 0 to +70°C
  - –40 to +85°C
  - –40 to +125°C

## PIN CONFIGURATIONS



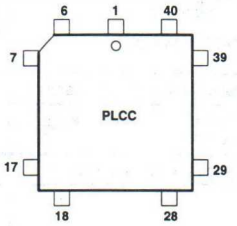




SEE PAGE 3-1031 FOR QFP AND LCC PIN FUNCTIONS.

* DO NOT CONNECT

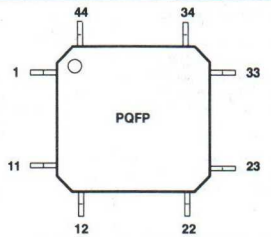
## PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC* | 23 | NC* |
| 2 | P1.0 | 24 | P2.0/A8 |
| 3 | P1.1 | 25 | P2.1/A9 |
| 4 | P1.2 | 26 | P2.2/A10 |
| 5 | P1.3 | 27 | P2.3/A11 |
| 6 | P1.4 | 28 | P2.4/A12 |
| 7 | P1.5 | 29 | P2.5/A13 |
| 8 | P1.6/SCL | 30 | P2.6/A14 |
| 9 | P1.7/SDA | 31 | P2.7/A15 |
| 10 | RST | 32 | $\overline{PSEN}$ |
| 11 | P3.0/RxD | 33 | ALE |
| 12 | NC* | 34 | NC* |
| 13 | P3.1/TxD | 35 | $\overline{EA}$ |
| 14 | P3.2/$\overline{INT0}$ | 36 | P0.7/AD7 |
| 15 | P3.3/$\overline{INT1}$ | 37 | P0.6/AD6 |
| 16 | P3.4/T0 | 38 | P0.5/AD5 |
| 17 | P3.5/T1 | 39 | P0.4/AD4 |
| 18 | P3.6/$\overline{WR}$ | 40 | P0.3/AD3 |
| 19 | P3.7/$\overline{RD}$ | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | $V_{SS}$ | 44 | $V_{DD}$ |

* DO NOT CONNECT

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS



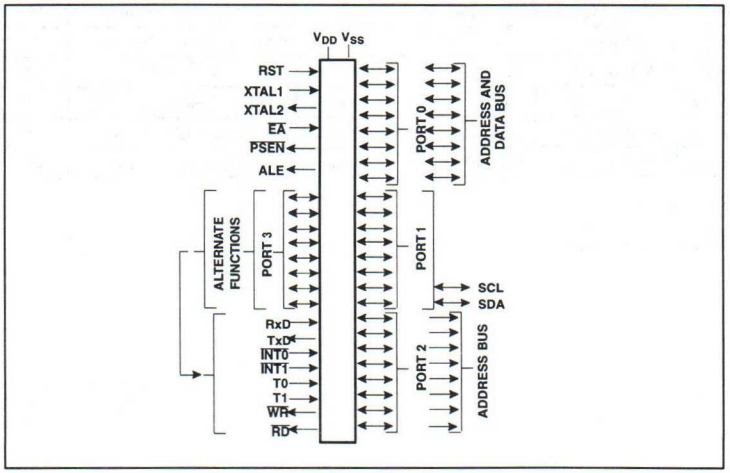

| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5 | 23 | P2.5/A13 |
| 2 | P1.6/SCL | 24 | P2.6/A14 |
| 3 | P1.7/SDA | 25 | P2.7/A15 |
| 4 | RST | 26 | $\overline{PSEN}$ |
| 5 | P3.0/RxD | 27 | ALE |
| 6 | $V_{SS4}$ | 28 | $V_{SS2}$ |
| 7 | P3.1/TxD | 29 | $\overline{EA}$/$V_{PP}$ |
| 8 | P3.2/$\overline{INT0}$ | 30 | P0.7/AD7 |
| 9 | P3.3/$\overline{INT1}$ | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/$\overline{WR}$ | 34 | P0.3/AD3 |
| 13 | P3.7/$\overline{RD}$ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | $V_{SS1}$ | 38 | $V_{DD}$ |
| 17 | NC* | 39 | $V_{SS3}$ |
| 18 | P2.0/A8 | 40 | P1.0 |
| 19 | P2.1/A9 | 41 | P1.1 |
| 20 | P2.2/A10 | 42 | P1.2 |
| 21 | P2.3/A11 | 43 | P1.3 |
| 22 | P2.4/A12 | 44 | P1.4 |

* DO NOT CONNECT

**NOTES TO QFP ONLY:**

1. Due to EMC improvements, all $V_{SS}$ pins (6, 16, 28, 39) must be connected to $V_{SS}$ on the 80C652/83C654.

## LOGIC SYMBOL



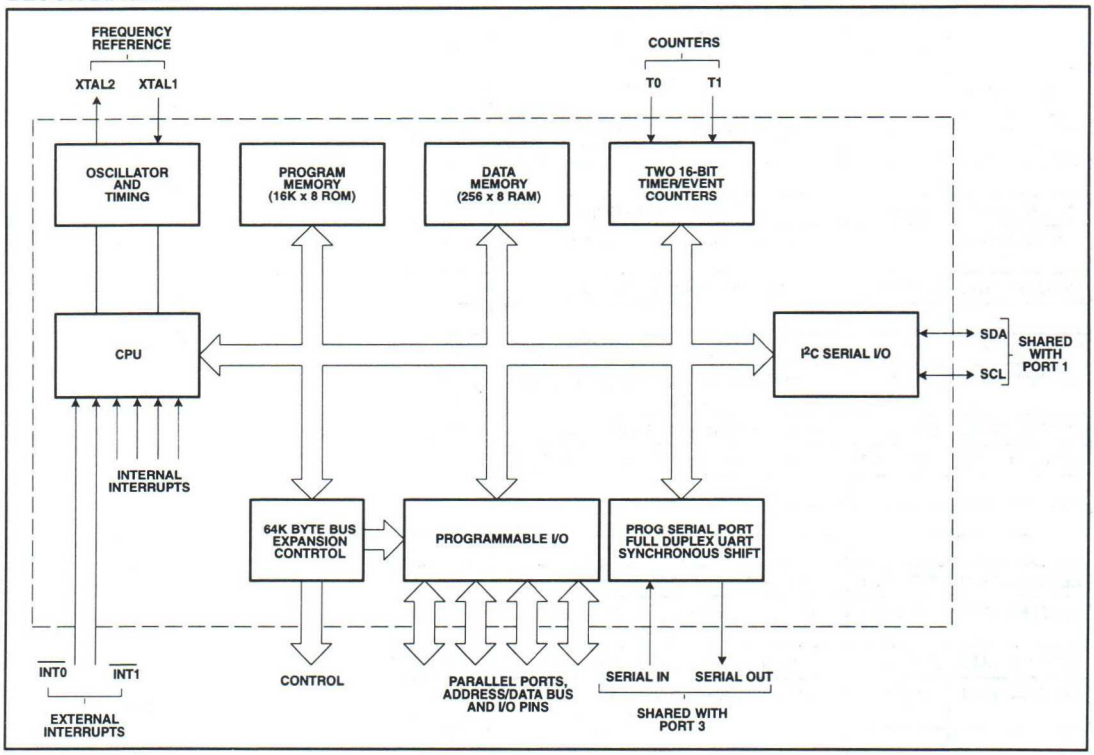
## ORDERING INFORMATION

| PHILIPS PART ORDER NUMBER PART MARKING | | PHILIPS NORTH AMERICA PART ORDER NUMBER | | Drawing Number | TEMPERATURE RANGE °C AND PACKAGE | FREQ MHz[2,3] |
|---|---|---|---|---|---|---|
| ROMless[1] | ROM | ROMless[1] | ROM | | | |
| P80C652FBP | P83C654FBP/xxx | P80C652FBPN | P83C654FBPN | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 16 |
| | | | | | 0 to +70, Ceramic Dual In-line Package w/Window | 16 |
| P80C652FBA | P83C654FBA/xxx | P80C652FBAA | P83C654FBAA | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| | | | | | | |
| P80C652FBB | P83C654FBB/xxx | P80C652FBBB | P83C654FBBB | SOT307-2[5] | 0 to +70, Plastic Quad Flat Pack | 16 |
| | P83C654FBR/xxx | | | SOT270-1 | 0 to +70, Plastic Shrink Dual In-Line Package | |
| P80C652FFP | P83C654FFP/xxx | P80C652FFPN | P83C654FFPN | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 16 |
| | | | | | | |
| P80C652FFA | P83C654FFA/xxx | P80C652FFAA | P83C654FFAA | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| P80C652FFB | P83C654FFB/xxx | P80C652FFBB | P83C654FFBB | SOT307-2[5] | –40 to +85, Plastic Quad Flat Pack | 16 |
| P80C652FHP | P83C654FHP/xxx | P80C652FHPN | P83C654FHPN | SOT129-1 | –40 to +125, Plastic Dual In-line Package | 16 |
| P80C652FHA | P83C654FHA/xxx | P80C652FHAA | P83C654FHAA | SOT187-2 | –40 to +125, Plastic Leaded Chip Carrier | 16 |
| P80C652FHB | P83C654FHB/xxx | P80C652FHBB | P83C654FHBB | SOT307-2[5] | –40 to +125, Plastic Quad Flat Pack | 16 |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| P80C652IBP | P83C654IBP/xxx | P80C652IBPN | P83C654IBPN | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 24 |
| P80C652IBA | P83C654IBA/xxx | P80C652IBAA | P83C654IBAA | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 24 |
| P80C652IBB | P83C654IBB/xxx | P80C652IBBB | P83C654IBBB | SOT307-2[5] | 0 to +70, Plastic Quad Flat Pack | 24 |
| P80C652IFP | P83C654IFP/xxx | P80C652IFPN | P83C654IFPN | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 24 |
| P80C652IFA | P83C654IFA/xxx | P80C652IFAA | P83C654IFAA | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 24 |
| P80C652IFB | P83C654IFB/xxx | P80C652IFBB | P83C654IFBB | SOT307-2[5] | –40 to +85, Plastic Quad Flat Pack | 24 |

**NOTES:**
1. For full specification, see the 80C652/83C652 data sheet.
2. 83C654 frequency range is 1.2MHz – 16MHz or 1.2MHz – 24MHz.
3. For specification of the EPROM version, see the 87C654 data sheet.
4. xxx denotes the ROM code number.
5. SOT311 replaced by SOT307-2.

| EPROM[3] | Drawing Number | TEMPERATURE RANGE °C AND PACKAGE | FREQ MHz[2,3] |
|---|---|---|---|
| S87C654-4N40 | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 16 |
| S87C654-4F40 | 0590B | 0 to +70, Ceramic Dual In-line Package w/Window | 16 |
| S87C654-4A44 | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| S87C654-4K44 | 1472A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 16 |
| S87C654–4B44 | SOT307-2 | 0 to +70, PlasticQuad Flat Pack | 16 |
| | | | |
| S87C654-5N40 | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 16 |
| S87C654-5F40 | 0590B | –40 to +85, Ceramic Dual In-line Package w/Window | 16 |
| S87C654-5A44 | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| S87C654-5B44 | SOT307-2 | –40 to +85, Plastic Quad Flat Pack | 16 |
| | | | |
| | | | |
| | | | |
| S87C654–7N40 | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 20 |
| S87C654–7F40 | 0590B | 0 to +70, Ceramic Dual In-line Package w/Window | 20 |
| S87C654–7A44 | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 20 |
| S87C654–7K44 | 1472A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 20 |
| | | | |
| S87C654–8N40 | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 20 |
| S87C654–8F40 | 0590B | –40 to +85, Ceramic Dual In-line Package w/Window | 20 |
| S87C654–8A44 | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 20 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

## BLOCK DIAGRAM



## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER DIP | PLCC | QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 6, 16, 28, 39 | I | **Ground:** 0V reference. With the QFP package all $V_{SS}$ pins ($V_{SS1}$ to $V_{SS4}$) must be connected. |
| $V_{DD}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups, except P1.6 and P1.7 which are open drain. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Alternate functions include: |
| P1.6 | 7 | 8 | 2 | I/O | **SCL:** $I^2C$-bus serial port clock line. |
| P1.7 | 8 | 9 | 3 | I/O | **SDA:** $I^2C$-bus serial port data line. |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{DD}$. |
| ALE | 30 | 33 | 27 | I/O | **Address Latch Enable:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency. Note that one ALE pulse is skipped during each access to external data memory. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** Read strobe to external program memory via Port 0 and Port 2. It is activated twice each machine cycle during fetches from the external program memory. When executing out of external program memory two activations of $\overline{PSEN}$ are skipped during each access to external data memory. $\overline{PSEN}$ is not activated (remains HIGH) during no fetches from external program memory. $\overline{PSEN}$ can sink/source 8 LSTTL inputs and can drive CMOS inputs without external pull–ups. |
| $\overline{EA}$ | 31 | 35 | 29 | I | **External Access:** If during a RESET, $\overline{EA}$ is held at TTL, level HIGH, the CPU executes out of the internal program memory ROM provided the Program Counter is less than 16384. If during a RESET, $\overline{EA}$ is held a TTL LOW level, the CPU executes out of external program memory. $\overline{EA}$ is not allowed to float. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTE:**
To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher than $V_{DD}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

**Table 1. 8XC652/654 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes) | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 | 0x000000B |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP*# | Interrupt priority | B8H | – | | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 | xx000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1*# | Port 1 | 90H | SDA | SCL | | | | | | | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TXD | RXD | FFH |
| PCON# | Power control | 87H | SMOD | – | – | – | GF1 | GF0 | PD | IDL | 0xxx0000B |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| S0CON*# | Serial 0 port control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| S0BUF# | Serial 0 data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | F1 | P | 00H |
| S1DAT# | Serial 1 data | DAH | | | | | | | | | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| S1ADR# | Serial 1 address | DBH | SLAVE ADDRESS | | | | | | | GC | 00H |
| S1STA# | Serial 1 status | D9H | SC4 | SC3 | SC2 | SC1 | SC0 | 0 | 0 | 0 | F8H |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| S1CON*# | Serial 1 control | D8H | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 | 00000000B |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| TH1 | Timer high 1 | 8DH | | | | | | | | | 00H |
| TH0 | Timer high 0 | 8CH | | | | | | | | | 00H |
| TL1 | Timer low 1 | 8BH | | | | | | | | | 00H |
| TL0 | Timer low 0 | 8AH | | | | | | | | | 00H |
| TMOD | Timer mode | 89H | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

## ROM CODE PROTECTION (83C654)

The 83C654 has an additional security feature. ROM code protection may be selected by setting a mask–programmable security bit (i.e., user dependent). This feature may be requested during ROM code submission. When selected, the ROM code is protected and cannot be read out at any time by any test mode or by any instruction in the external program memory space.

The MOVC instructions are the only instructions that have access to program code in the internal or external program memory. The $\overline{EA}$ input is latched during RESET and is "don't care" after RESET (also if the security bit is not set). This implementation prevents reading internal program code by switching from external program memory to internal program memory during a MOVC instruction or any other instruction that uses immediate data.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the Logic Symbol, page 3-1031.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## Reset

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{DD}$ and RST must come up at the same time for a proper start-up.

## Idle Mode

In the idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## Power-Down Mode

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. The control bits for the reduced power modes are in the special function register PCON. Table 2 shows the state of the I/O ports during low current operating modes.

## I²C SERIAL COMMUNICATION — SIO1

The I²C serial port is identical to the I²C serial port on the 8XC552. The operation of this subsystem is described in detail in the 8XC552 section of this manual.

Note that in both the 8XC652/4 and the 8XC552 the I²C pins are alternate functions to port pins P1.6 and P1.7. Because of this, P1.6 and P1.7 on these parts do not have a pull-up structure as found on the 80C51. Therefore P1.6 and P1.7 have open drain outputs on the 8XC652/4.

**Table 2. External Pin Status During Idle and Power-Down Mode**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

**Serial Control Register (S1CON) – See Table 3**

| S1CON (D8H) | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |
|---|---|---|---|---|---|---|---|---|

**Bits CR0, CR1 and CR2 determine the serial clock frequency that is generated in the master mode of operation.**

**Table 3. Serial Clock Rates**

| CR2 | CR1 | CR0 | BIT FREQUENCY (kHz) AT $f_{OSC}$ 6MHz | 12MHz | 16MHz | 24MHz | $f_{OSC}$ DIVIDED BY |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 23 | 47 | 62.5 | 94 | 256 |
| 0 | 0 | 1 | 27 | 54 | 71 | 107[1] | 224 |
| 0 | 1 | 0 | 31.25 | 62.5 | 83.3 | 125[1] | 192 |
| 0 | 1 | 1 | 37 | 75 | 100 | 150[1] | 160 |
| 1 | 0 | 0 | 6.25 | 12.5 | 17 | 25 | 960 |
| 1 | 0 | 1 | 50 | 100 | 133[1] | 200[1] | 120 |
| 1 | 1 | 0 | 100 | 200[1] | 267[1] | 400[1] | 60 |
| 1 | 1 | 1 | 0.24 < 62.5 0 to 255 | 0.49 < 62.5 0 to 254 | 0.65 < 55.6 0 to 253 | 0.98 < 50.0 0 to 251 | 96 × (256 – (reload value Timer 1)) reload value range Timer 1 (in mode 2) |

**NOTES:**
1. These frequencies exceed the upper limit of 100kHz of the I²C-bus specification and cannot be used in an I²C-bus application.

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage on any other pin to $V_{SS}$ | –0.5 to + 6.5 | V |
| Input, output current on any single pin | ±5 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DEVICE SPECIFICATIONS

| TYPE | SUPPLY VOLTAGE (V) | | FREQUENCY (MHz) | | TEMPERATURE RANGE (°C) |
|---|---|---|---|---|---|
| | MIN. | MAX. | MIN. | MAX. | |
| P83C654FBx | 4.0 | 6.0 | 1.2 | 16 | 0 to +70 |
| P83C654FFx | 4.0 | 6.0 | 1.2 | 16 | –40 to +85 |
| P83C654FHx | 4.5 | 5.5 | 1.2 | 16 | –40 to +125 |
| P83C654IBx | 4.5 | 5.5 | 1.2 | 24 | 0 to +70 |
| P83C654IFx | 4.5 | 5.5 | 1.2 | 24 | –40 to +85 |

## DC ELECTRICAL CHARACTERISTICS

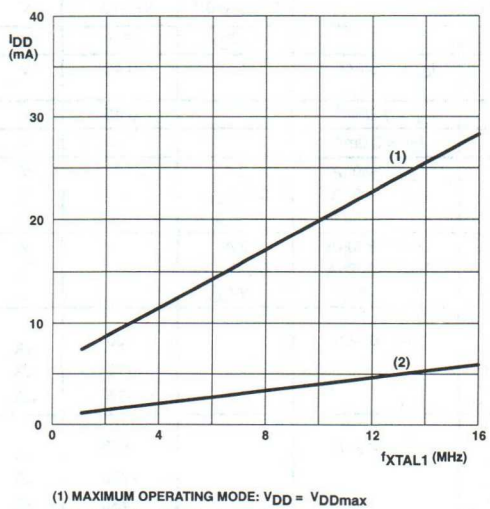
$V_{SS}$ = 0V, $V_{DD}$ = 5V ± 10%

| SYMBOL | PARAMETER | PART TYPE | TEST CONDITIONS | LIMITS MIN. | LIMITS MAX. | UNIT |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input low voltage, except $\overline{EA}$, P1.6/SCL, P1.7/SDA | 0 to +70°C | | –0.5 | $0.2V_{DD}$–0.1 | V |
| | | –40 to +85°C | | –0.5 | $0.2V_{DD}$–0.15 | V |
| | | –40 to +125°C | | –0.5 | $0.2V_{DD}$–0.25 | V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | 0 to +70°C | | –0.5 | $0.2V_{DD}$–0.3 | V |
| | | –40 to +85°C | | –0.5 | $0.2V_{DD}$–0.35 | V |
| | | –40 to +125°C | | –0.5 | $0.2V_{DD}$–0.45 | V |
| $V_{IL2}$ | Input low voltage to P1.6/SCL, P1.7/SDA[6] | | | –0.5 | $0.3V_{DD}$ | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST, P1.6/SCL, P1.7/SDA | 0 to +70°C | | $0.2V_{DD}$+0.9 | $V_{DD}$+0.5 | V |
| | | –40 to +85°C | | $0.2V_{DD}$+1.0 | $V_{DD}$+0.5 | V |
| | | –40 to +125°C | | $0.2V_{DD}$+1.0 | $V_{DD}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | 0 to +70°C | | $0.7V_{DD}$ | $V_{DD}$+0.5 | V |
| | | –40 to +85°C | | $0.7V_{DD}$+0.1 | $V_{DD}$+0.5 | V |
| | | –40 to +125°C | | $0.7V_{DD}$+0.1 | $V_{DD}$+0.5 | V |
| $V_{IH2}$ | Input high voltage, P1.6/SCL, P1.7/SDA[6] | | | $0.7V_{DD}$ | 6.0 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3, except P1.6/SCL, P1.7/SDA | | $I_{OL}$ = 1.6mA[8, 9] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$ | | $I_{OL}$ = 3.2mA[8, 9] | | 0.45 | V |
| $V_{OL2}$ | Output low voltage, P1.6/SCL, P1.7/SDA | | $I_{OL}$ = 3.0mA | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3, ALE, $\overline{PSEN}$[10] | | $I_{OH}$ = –60µA | 2.4 | | V |
| | | | $I_{OH}$ = –25µA | $0.75V_{DD}$ | | V |
| | | | $I_{OH}$ = –10µA | $0.9V_{DD}$ | | V |
| $V_{OH1}$ | Output high voltage; port 0 in external bus mode | | $I_{OH}$ = –800µA | 2.4 | | V |
| | | | $I_{OH}$ = –300µA | $0.75V_{DD}$ | | V |
| | | | $I_{OH}$ = –80µA | $0.9V_{DD}$ | | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3, except P1.6/SCL, P1.7/SDA | 0 to +70°C | $V_{IN}$ = 0.45V | | –50 | µA |
| | | –40 to +85°C | | | –75 | µA |
| | | –40 to +125°C | | | –75 | µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3, except P1.6/SCL, P1.7/SDA | 0 to +70°C | See note 7 | | –650 | µA |
| | | –40 to +85°C | | | –750 | µA |
| | | –40 to +125°C | | | –750 | µA |
| $I_{L1}$ | Input leakage current, port 0, $\overline{EA}$ | | 0.45V < $V_I$ < $V_{DD}$ | | ±10 | µA |
| $I_{L2}$ | Input leakage current, P1.6/SCL, P1.7/SDA | | 0V < $V_I$ < 6.0V | | ±10 | µA |
| | | | 0V < $V_{DD}$ < 6.0V | | | µA |
| $I_{DD}$ | Power supply current: | | See note 1 | | | |
| | Active mode @ 16MHz[2, 11] | | $V_{DD}$=6.0V | | 28.0 | mA |
| | Active mode @ 24MHz[2, 11] | | $V_{DD}$=5.5V | | 35.0 | mA |
| | Idle mode @ 16MHz[3, 11] | | | | 6 | mA |
| | Idle mode @ 24MHz[3, 11] | | | | 7 | mA |
| | Power down mode[4, 5] | | | | 50 | µA |
| | Power down mode[4, 5] | –40 to +125°C | | | 100 | µA |
| $R_{RST}$ | Internal reset pull-down resistor | | | 50 | 150 | kΩ |
| $C_{IO}$ | Pin capacitance | | Freq.=1MHz | | 10 | pF |

**NOTES ON NEXT PAGE.**

**NOTES FOR DC ELECTRICAL CHARACTERISTICS:**

1. See Figures 9 through 11 for $I_{DD}$ test conditions.
2. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5V$; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = P1.6 = P1.7 = $V_{DD}$. See Figure 9.
3. The idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5V$; XTAL2 not connected; Port 0 = P1.6 = P1.7 = $V_{DD}$; $\overline{EA}$ = RST = $V_{SS}$. See Figure 10.
4. The power-down current is measured with all output pins disconnected; XTAL2 not connected; Port 0 = P1.6 = P1.7 = $V_{DD}$; $\overline{EA}$ = RST = $V_{SS}$. See Figure 11.
5. $2V \leq V_{PD} \leq V_{DD}max$.
6. The input threshold voltage of P1.6 and P1.7 (SIO1) meets the $I^2C$ specification, so an input voltage below $0.3V_{DD}$ will be recognized as a logic 0 while an input voltage above $0.7V_{DD}$ will be recognized as a logic 1.
7. Pins of ports 1 , 2, and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
8. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.
9. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows: Maximum $I_{OL}$ = 10mA per port pin; Maximum $I_{OL}$ = 26mA total for Port 0; Maximum $I_{OL}$ = 15mA total for Ports 1, 2, and 3; Maximum $I_{OL}$ = 71mA total for all output pins. If $I_{OL}$ exceeds the test conditions, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
10. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{DD}$ specification when the address bits are stabilizing.
11. $I_{DDMAX}$ for other frequencies can be derived from Figure 1, where FREQ is the external oscillator frequency in MHz. $I_{DDMAX}$ is given in mA.



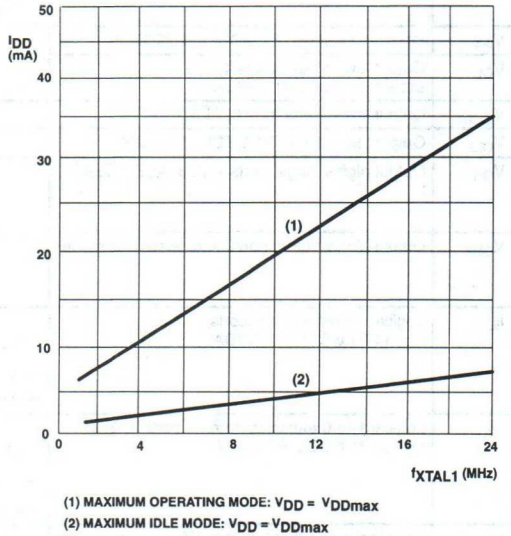




**Figure 1. $I_{DD}$ vs. Frequency**

## AC ELECTRICAL CHARACTERISTICS[1, 2] (16 MHz type)

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 2 | Oscillator frequency | | | 1.2 | 16 | MHz |
| $t_{LHLL}$ | 2 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 2 | Address valid to ALE low | 8 | | $t_{CLCL}-55$ | | ns |
| $t_{LLAX}$ | 2 | Address hold after ALE low | 28 | | $t_{CLCL}-35$ | | ns |
| $t_{LLIV}$ | 2 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 2 | ALE low to $\overline{PSEN}$ low | 23 | | $t_{CLCL}-40$ | | ns |
| $t_{PLPH}$ | 2 | $\overline{PSEN}$ pulse width | 143 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 2 | $\overline{PSEN}$ low to valid instruction in | | 83 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 2 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 2 | Input instruction float after $\overline{PSEN}$ | | 38 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 2 | Address to valid instruction in | | 208 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 2 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 3, 4 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 3, 4 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 3, 4 | $\overline{RD}$ low to valid data in | | 148 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 3, 4 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 3, 4 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-70$ | ns |
| $t_{LLDV}$ | 3, 4 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 3, 4 | Address to valid data in | | 398 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 3, 4 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 138 | 238 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 3, 4 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 120 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 3, 4 | Data valid to $\overline{WR}$ transition | 3 | | $t_{CLCL}-60$ | | ns |
| $t_{DW}$ | 3, 4 | Data setup time before $\overline{WR}$ | 288 | | $7t_{CLCL}-150$ | | ns |
| $t_{WHQX}$ | 3, 4 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{RLAZ}$ | 3, 4 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 3, 4 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 5 | Serial port clock cycle time[3] | 0.75 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 5 | Output data setup to clock rising edge[3] | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 5 | Output data hold after clock rising edge[3] | 80 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 5 | Input data hold after clock rising edge[3] | 0 | | 0 | | ns |
| $t_{XHDV}$ | 5 | Clock rising edge to input data valid[3] | | 492 | | $10t_{CLCL}-133$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 6 | High time[3] | 20 | | 20 | $t_{CLCL}-t_{CLCX}$ | ns |
| $t_{CLCX}$ | 6 | Low time[3] | 20 | | 20 | $t_{CLCL}-t_{CHCX}$ | ns |
| $t_{CLCH}$ | 6 | Rise time[3] | | 20 | | 20 | ns |
| $t_{CHCL}$ | 6 | Fall time[3] | | 20 | | 20 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. These values are characterized but not 100% production tested.

## AC ELECTRICAL CHARACTERISTICS[1, 2] (24 MHz type)

| SYMBOL | FIGURE | PARAMETER | 24MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 2 | Oscillator frequency | | | 1.2 | 24 | MHz |
| $t_{LHLL}$ | 2 | ALE pulse width | 43 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 2 | Address valid to ALE low | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{LLAX}$ | 2 | Address hold after ALE low | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{LLIV}$ | 2 | ALE low to valid instruction in | | 102 | | $4t_{CLCL}-65$ | ns |
| $t_{LLPL}$ | 2 | ALE low to $\overline{PSEN}$ low | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{PLPH}$ | 2 | $\overline{PSEN}$ pulse width | 80 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 2 | $\overline{PSEN}$ low to valid instruction in | | 65 | | $3t_{CLCL}-60$ | ns |
| $t_{PXIX}$ | 2 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 2 | Input instruction float after $\overline{PSEN}$ | | 17 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 2 | Address to valid instruction in | | 128 | | $5t_{CLCL}-80$ | ns |
| $t_{PLAZ}$ | 2 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 3, 4 | $\overline{RD}$ pulse width | 150 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 3, 4 | $\overline{WR}$ pulse width | 150 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 3, 4 | $\overline{RD}$ low to valid data in | | 118 | | $5t_{CLCL}-90$ | ns |
| $t_{RHDX}$ | 3, 4 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 3, 4 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-28$ | ns |
| $t_{LLDV}$ | 3, 4 | ALE low to valid data in | | 180 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 3, 4 | Address to valid data in | | 210 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 3, 4 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 75 | 175 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 3, 4 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 92 | | $4t_{CLCL}-75$ | | ns |
| $t_{QVWX}$ | 3, 4 | Data valid to $\overline{WR}$ transition | 12 | | $t_{CLCL}-30$ | | ns |
| $t_{DW}$ | 3, 4 | Data setup time before $\overline{WR}$ | 162 | | $7t_{CLCL}-130$ | | ns |
| $t_{WHQX}$ | 3, 4 | Data hold after $\overline{WR}$ | 17 | | $t_{CLCL}-25$ | | ns |
| $t_{RLAZ}$ | 3, 4 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 3, 4 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 17 | 67 | $t_{CLCL}-25$ | $t_{CLCL}+25$ | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 5 | Serial port clock cycle time[3] | 0.5 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 5 | Output data setup to clock rising edge[3] | 283 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 5 | Output data hold after clock rising edge[3] | 23 | | $2t_{CLCL}-60$ | | ns |
| $t_{XHDX}$ | 5 | Input data hold after clock rising edge[3] | 0 | | 0 | | ns |
| $t_{XHDV}$ | 5 | Clock rising edge to input data valid[3] | | 283 | | $10t_{CLCL}-133$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 6 | High time[3] | 17 | | 17 | $t_{CLCL} - t_{CLCX}$ | ns |
| $t_{CLCX}$ | 6 | Low time[3] | 17 | | 17 | $t_{CLCL} - t_{CHCX}$ | ns |
| $t_{CLCH}$ | 6 | Rise time[3] | | 5 | | 5 | ns |
| $t_{CHCL}$ | 6 | Fall time[3] | | 5 | | 5 | ns |

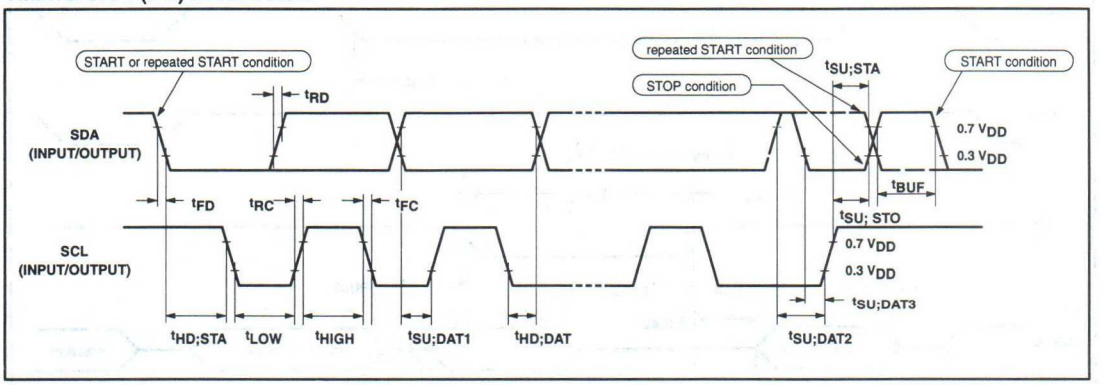
**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. These values are characterized but not 100% production tested.

## AC ELECTRICAL CHARACTERISTICS – I²C INTERFACE

| SYMBOL | PARAMETER | INPUT | OUTPUT |
|---|---|---|---|
| **SCL TIMING CHARACTERISTICS** | | | |
| $t_{HD;STA}$ | START condition hold time | $\geq 14\ t_{CLCL}$ | $> 4.0\mu s$[1] |
| $t_{LOW}$ | SCL LOW time | $\geq 16\ t_{CLCL}$ | $> 4.7\mu s$[1] |
| $t_{HIGH}$ | SCL HIGH time | $\geq 14\ t_{CLCL}$ | $> 4.0\mu s$[1] |
| $t_{RC}$ | SCL rise time | $\leq 1\mu s$ | –[2] |
| $t_{FC}$ | SCL fall time | $\leq 0.3\mu s$ | $< 0.3\mu s$[3] |
| **SDA TIMING CHARACTERISTICS** | | | |
| $t_{SU;DAT1}$ | Data set-up time | $\geq 250ns$ | $> 20\ t_{CLCL} - t_{RD}$ |
| $t_{SU;DAT2}$ | SDA set-up time (before rep. START cond.) | $\geq 250ns$ | $> 1\mu s$[1] |
| $t_{SU;DAT3}$ | SDA set-up time (before STOP cond.) | $\geq 250ns$ | $> 8\ t_{CLCL}$ |
| $t_{HD;DAT}$ | Data hold time | $\geq 0ns$ | $> 8\ t_{CLCL} - t_{FC}$ |
| $t_{SU;STA}$ | Repeated START set-up time | $\geq 14\ t_{CLCL}$ | $> 4.7\mu s$[1] |
| $t_{SU;STO}$ | STOP condition set-up time | $\geq 14\ t_{CLCL}$ | $> 4.0\mu s$[1] |
| $t_{BUF}$ | Bus free time | $\geq 14\ t_{CLCL}$ | $> 4.7\mu s$[1] |
| $t_{RD}$ | SDA rise time | $\leq 1\mu s$ | –[2] |
| $t_{FD}$ | SDA fall time | $\leq 0.3\mu s$ | $< 0.3\mu s$[3] |

**NOTES:**
1. At 100 kbit/s. At other bit rates this value is inversely proportional to the bit-rate of 100 kbit/s.
2. Determined by the external bus-line capacitance and the external bus-line pull-resistor, this must be < 1µs.
3. Spikes on the SDA and SCL lines with a duration of less than 3 $t_{CLCL}$ will be filtered out. Maximum capacitance on bus-lines SDA and SCL = 400pF.
4. $t_{CLCL} = 1/f_{OSC}$ = one oscillator clock period at pin XTAL1. For 63ns (42ns) < $t_{CLCL}$ < 285ns (16MHz (24MHz) > $f_{OSC}$ > 3.5MHz) the SIO1 interface meets the I²C-bus specification for bit-rates up to 100 kbit/s.

## TIMING SIO1 (I²C) INTERFACE



## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{\text{PSEN}}$
Q – Output data
R – $\overline{\text{RD}}$ signal
t – Time
V – Valid
W – $\overline{\text{WR}}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{\text{PSEN}}$ low.

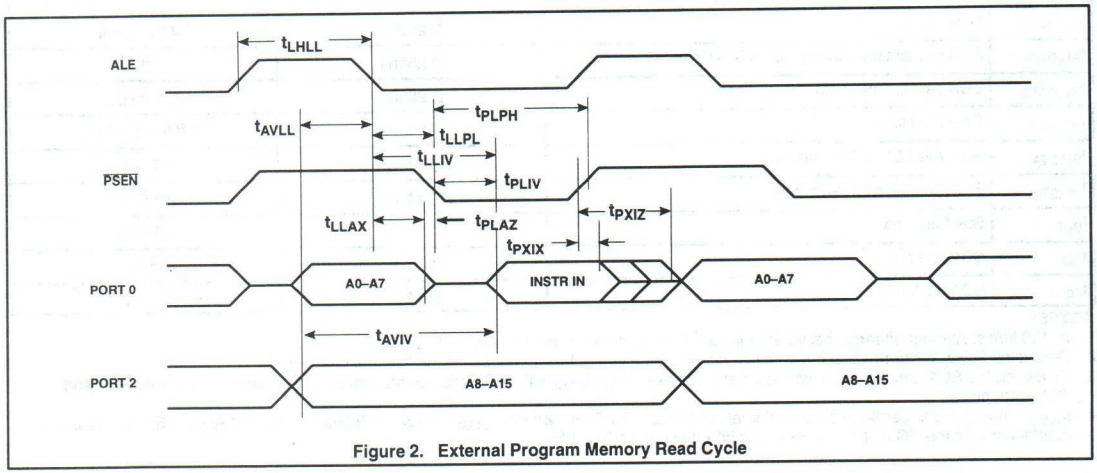



**Figure 2. External Program Memory Read Cycle**

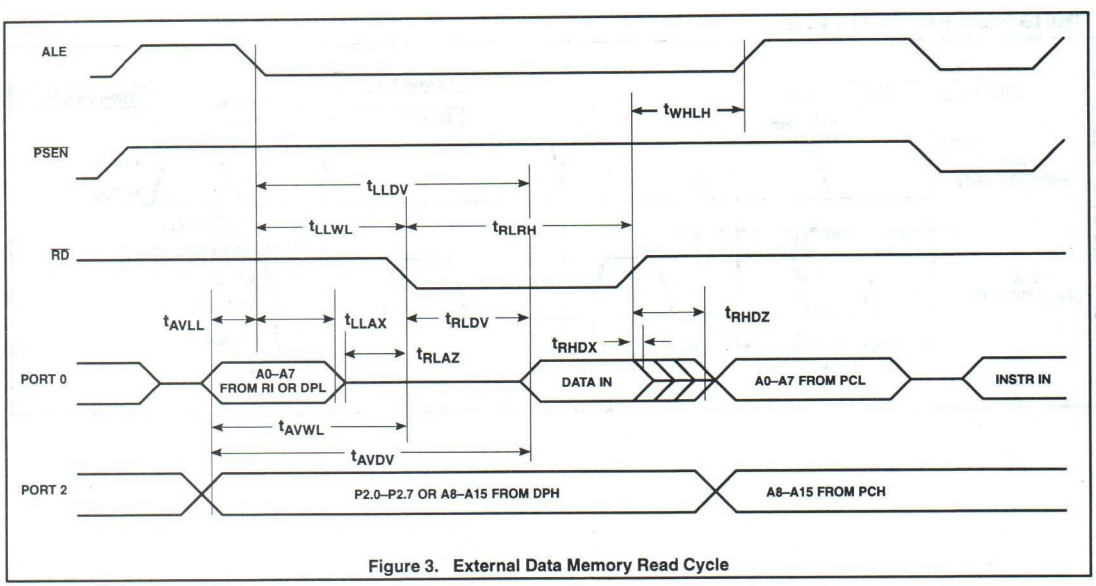



**Figure 3. External Data Memory Read Cycle**

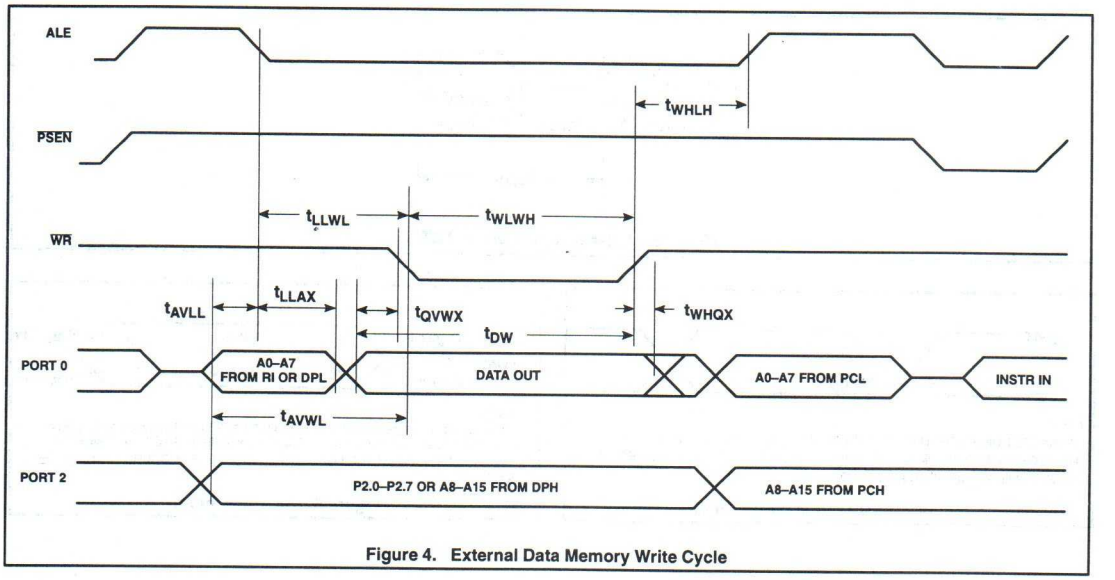



**Figure 4. External Data Memory Write Cycle**

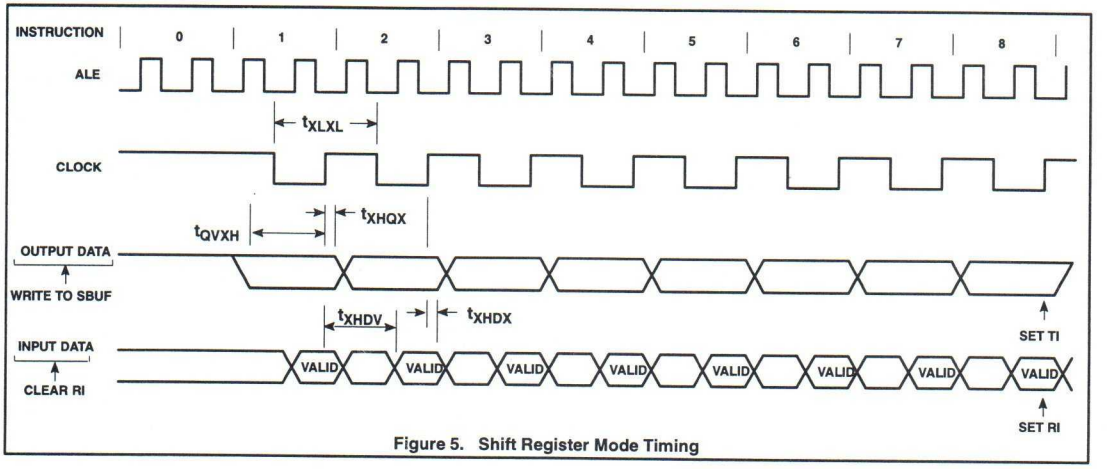



**Figure 5. Shift Register Mode Timing**

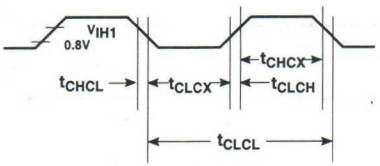
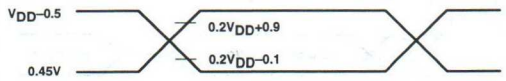
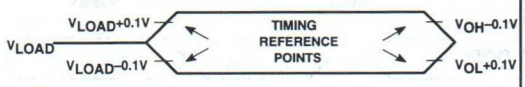



**Figure 6. External Clock Drive at XTAL1**




**NOTE:**
AC INPUTS DURING TESTING ARE DRIVEN AT $V_{DD}$–0.5 FOR A LOGIC '1' AND 0.45V FOR A LOGIC '0'. TIMING MEASUREMENTS ARE MADE AT $V_{IH}$ MIN FOR A LOGIC '1' AND $V_{IL}$ MAX FOR A LOGIC '0'.

**Figure 7. AC Testing Input/Output**




**NOTE:**
FOR TIMING PURPOSES, A PORT IS NO LONGER FLOATING WHEN A 100MV CHANGE FROM LOAD VOLTAGE OCCURS, AND BEGINS TO FLOAT WHEN A 100mV CHANGE FROM THE LOADED $V_{OH}/V_{OL}$ LEVEL OCCURS. $I_{OH}/I_{OL} \geq \pm$ 20mA.

**Figure 8. Float Waveform**

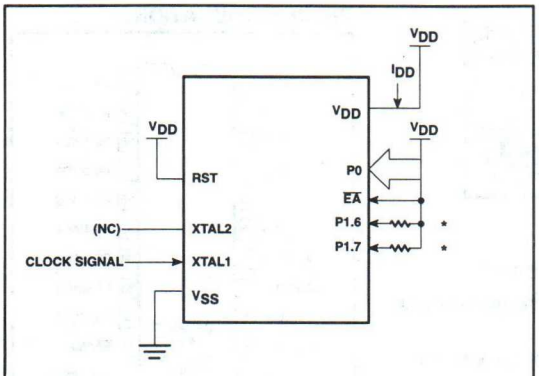



**Figure 9. $I_{DD}$ Test Condition, Active Mode**
**All other pins are disconnected**

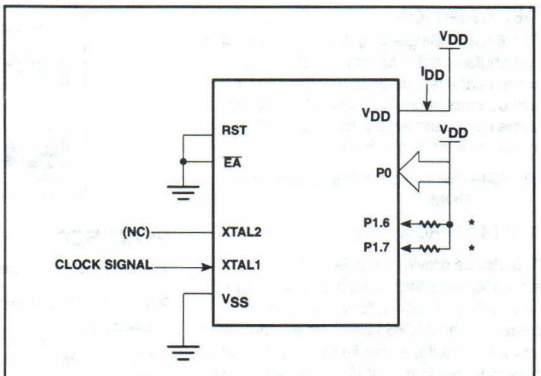



**Figure 10. $I_{DD}$ Test Condition, Idle Mode**
**All other pins are disconnected**

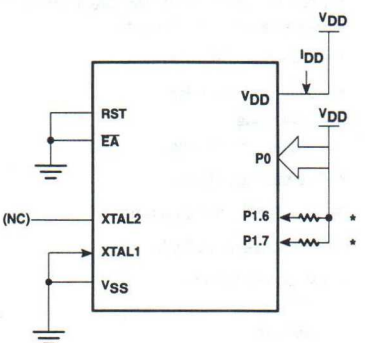



**Figure 11. $I_{DD}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{DD}$ = 2V to 5.5V**

**NOTE:**
* Ports 1.6 and 1.7 should be connected to $V_{CC}$ through resistors of sufficiently high value such that the sink current into these pins does not exceed the $I_{OL1}$ specification.

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# CMOS single-chip 8-bit microcontroller — 87C654

## DESCRIPTION

The 87C654 Single-Chip 8-Bit Microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 87C654 has the same instruction set as the 80C51. Two versions of the derivative exist:

83C654—16k bytes mask programmable ROM

87C654—EPROM version

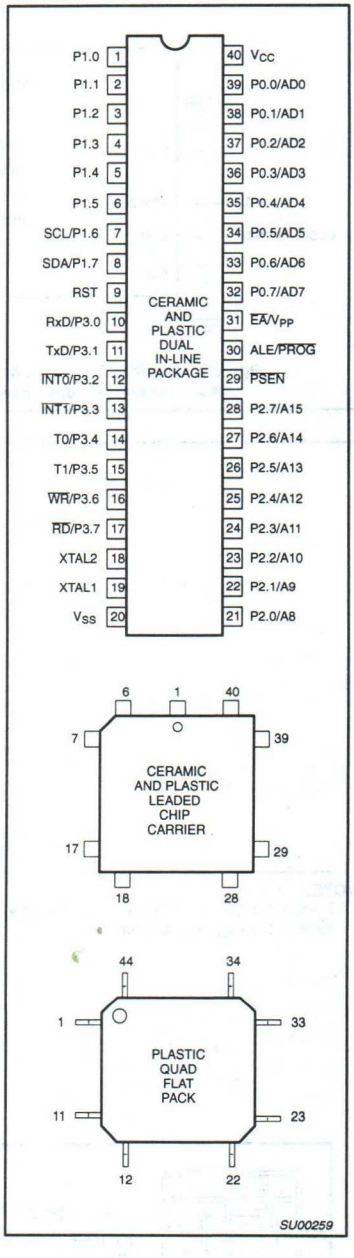
This device provides architectural enhancements that make it applicable in a variety of applications for general control systems. The 87C654 contains a non-volatile 16k × 8 EPROM, a volatile 256 × 8 read/write data memory, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a multi-source, two-priority-level, nested interrupt structure, an $I^2C$ interface, UART and on-chip oscillator and timing circuits. For systems that require extra capability, the 87C654 can be expanded using standard TTL compatible memories and logic.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte and 17 three-byte. With a 16MHz crystal, 58% of the instructions are executed in 0.75µs and 40% in 1.5µs. Multiply and divide instructions require 3µs.

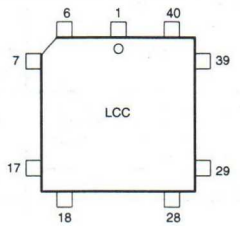
## FEATURES

- 80C51 central processing unit
- 16k × 8 EPROM expandable externally to 64k bytes
- 256 × 8 RAM, expandable externally to 64k bytes
- Two standard 16-bit timer/counters
- Four 8-bit I/O ports
- $I^2C$-bus serial I/O port with byte oriented master and slave functions
- Full-duplex UART facilities
- Power control modes
  - Idle mode
  - Power-down mode
- Five package styles
- Extended temperature range
- OTP package available
- Two speed ranges
  - 16MHz
  - 20MHz

## PIN CONFIGURATIONS



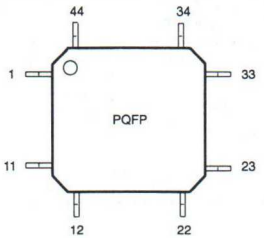
## CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC* | 23 | NC8 |
| 2 | P1.0 | 24 | P2.0/A8 |
| 3 | P1.1 | 25 | P2.1/A9 |
| 4 | P1.2 | 26 | P2.2/A10 |
| 5 | P1.3 | 27 | P2.3/A11 |
| 6 | P1.4 | 28 | P2.4/A12 |
| 7 | P1.5 | 29 | P2.5/A13 |
| 8 | P1.6/SCL | 30 | P2.6/A14 |
| 9 | P1.7/SDA | 31 | P2.7/A15 |
| 10 | RST | 32 | $\overline{PSEN}$ |
| 11 | P3.0/RxD | 33 | ALE/$\overline{PROG}$ |
| 12 | NC8 | 34 | NC8 |
| 13 | P3.1/TxD | 35 | $\overline{EA}$/$V_{PP}$ |
| 14 | P3.2/$\overline{INT0}$ | 36 | P0.7/AD7 |
| 15 | P3.3/$\overline{INT1}$ | 37 | P0.6/AD6 |
| 16 | P3.4/T0 | 38 | P0.5/AD5 |
| 17 | P3.5/T1 | 39 | P0.4/AD4 |
| 18 | P3.6/$\overline{WR}$ | 40 | P0.3/AD3 |
| 19 | P3.7/$\overline{RD}$ | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | $V_{SS}$ | 44 | $V_{CC}$ |

* DO NOT CONNECT SU00260

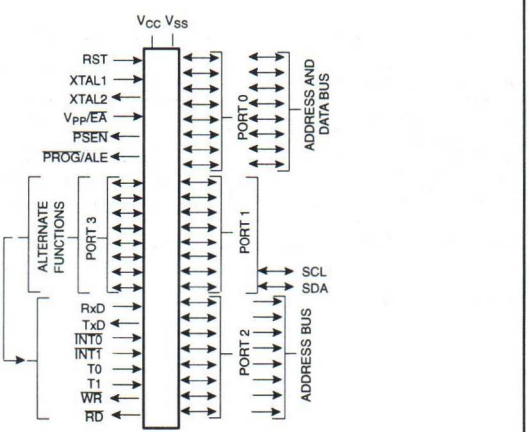
## PLASTIC QUAD FLAT PACK PIN FUNCTIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5 | 23 | P2.5/A13 |
| 2 | P1.6/SCL | 24 | P2.6/A14 |
| 3 | P1.7/SDA | 25 | P2.7/A15 |
| 4 | RST | 26 | $\overline{PSEN}$ |
| 5 | P3.0/RxD | 27 | ALE/$\overline{PROG}$ |
| 6 | NC* | 28 | NC* |
| 7 | P3.1/TxD | 29 | $\overline{EA}$/$V_{PP}$ |
| 8 | P3.2/$\overline{INT0}$ | 30 | P0.7/AD7 |
| 9 | P3.3/$\overline{INT1}$ | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/$\overline{WR}$ | 34 | P0.3/AD3 |
| 13 | P3.7/$\overline{RD}$ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | $V_{SS}$ | 38 | $V_{CC}$ |
| 17 | NC* | 39 | NC* |
| 18 | P2.0/A8 | 40 | P1.0 |
| 19 | P2.1/A9 | 41 | P1.1 |
| 20 | P2.2/A10 | 42 | P1.2 |
| 21 | P2.3/A11 | 43 | P.13 |
| 22 | P2.4/A12 | 44 | P1.4 |

* DO NOT CONNECT SU00261

## LOGIC SYMBOL




SU00262

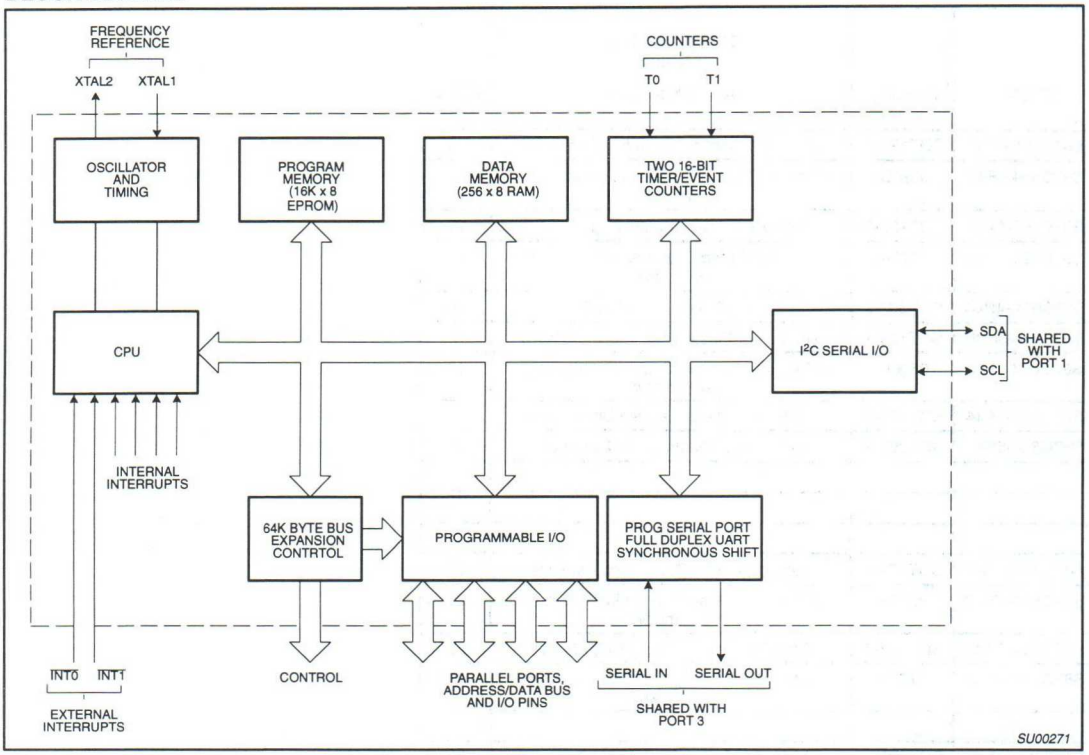
## ORDERING INFORMATION

| PHILIPS PART ORDER NUMBER PART MARKING | | PHILIPS NORTH AMERICA PART ORDER NUMBER | | | TEMPERATURE RANGE °C AND PACKAGE | FREQ MHz |
|---|---|---|---|---|---|---|
| ROMless | ROM | ROMless | ROM | Drawing Number | | |
| P80C652FBP | P83C654FBP/xxx | S80C652FBPN | S83C654FBPN | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 16 |
| | | | | | 0 to +70, Ceramic Dual In-line Package w/Window | 16 |
| P80C652FBA | P83C654FBA/xxx | S80C652FBAA | S83C654FBAA | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| | | | | | | |
| P80C652FBB | P83C654FBB/xxx | S80C652FBBB | S83C654FBBB | SOT307-2[4] | 0 to +70, Plastic Quad Flat Pack | 16 |
| P80C652FFP | P83C654FFP/xxx | S80C652FFPN | S83C654FFPN | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 16 |
| | | | | | | |
| P80C652FFA | P83C654FFA/xxx | S80C652FFAA | S83C654FFAA | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| P80C652FFB | P83C654FFB/xxx | S80C652FFBB | S83C654FFBB | SOT307-2[4] | –40 to +85, Plastic Quad Flat Pack | 16 |
| P80C652FHP | P83C654FHP/xxx | S80C652FHPN | S83C654FHPN | SOT129-1 | –40 to +125, Plastic Dual In-line Package | 16 |
| P80C652FHA | P83C654FHA/xxx | S80C652FHAA | S83C654FHAA | SOT187-2 | –40 to +125, Plastic Leaded Chip Carrier | 16 |
| P80C652FHB | P83C654FHB/xxx | S80C652FHBB | S83C654FHBB | SOT307-2[4] | –40 to +125, Plastic Quad Flat Pack | 16 |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| P80C652IBP | P83C654IBP/xxx | S80C652IBPN | S83C654IBPN | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 24 |
| P80C652IBA | P83C654IBA/xxx | S80C652IBAA | S83C654IBAA | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 24 |
| P80C652IBB | P83C654IBB/xxx | S80C652IBBB | S83C654IBBB | SOT307-2[4] | 0 to +70, Plastic Quad Flat Pack | 24 |
| P80C652IFP | P83C654IFP/xxx | S80C652IFPN | S83C654IFPN | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 24 |
| P80C652IFA | P83C654IFA/xxx | S80C652IFAA | S83C654IFAA | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 24 |
| P80C652IFB | P83C654IFB/xxx | S80C652IFBB | S83C654IFBB | SOT307-2[4] | –40 to +85, Plastic Quad Flat Pack | 24 |

**NOTES:**
1. For full specification, see the 87C652 data sheet.
2. 87C654 frequency range is 3.5MHz – 16MHz or 3.5MHz – 24MHz.
3. xxx denotes the ROM code number.
4. SOT311 replaced by SOT307-2.

| EPROM | Drawing Number | TEMPERATURE RANGE °C AND PACKAGE | FREQ MHz |
|---|---|---|---|
| S87C654-4N40 | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 16 |
| S87C654-4F40 | 0590B | 0 to +70, Ceramic Dual In-line Package w/Window | 16 |
| S87C654-4A44 | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 16 |
| S87C654-4K44 | 1472A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 16 |
| S87C654–4B44 | SOT307-2 | 0 to +70, Plastic Quad Flat Pack | 16 |
| S87C654-5N40 | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 16 |
| S87C654-5F40 | 0590B | –40 to +85, Ceramic Dual In-line Package w/Window | 16 |
| S87C654-5A44 | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 16 |
| S87C654-5B44 | SOT307-2 | –40 to +85, Plastic Quad Flat Pack | 16 |
| | | | |
| | | | |
| | | | |
| S87C654–7N40 | SOT129-1 | 0 to +70, Plastic Dual In-line Package | 20 |
| S87C654–7F40 | 0590B | 0 to +70, Ceramic Dual In-line Package w/Window | 20 |
| S87C654–7A44 | SOT187-2 | 0 to +70, Plastic Leaded Chip Carrier | 20 |
| S87C654–7K44 | 1472A | 0 to +70, Ceramic Leaded Chip Carrier w/Window | 20 |
| | | | |
| S87C654–8N40 | SOT129-1 | –40 to +85, Plastic Dual In-line Package | 20 |
| S87C654–8F40 | 0590B | –40 to +85, Ceramic Dual In-line Package w/Window | 20 |
| S87C654–8A44 | SOT187-2 | –40 to +85, Plastic Leaded Chip Carrier | 20 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

## BLOCK DIAGRAM



## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER DIP | LCC | QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0V reference. |
| $V_{CC}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification in the 87C654. External pull-ups are required during program verification. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups, except P1.6 and P1.7 which are open drain. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also receives the low-order address byte during program memory verification. Alternate functions include: |
| P1.6 | 7 | 8 | 2 | I/O | **SCL:** $I^2C$-bus serial port clock line. |
| P1.7 | 8 | 9 | 3 | I/O | **SDA:** $I^2C$-bus serial port data line. |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{CC}$. |
| ALE/$\overline{PROG}$ | 30 | 33 | 27 | I/O | **Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the 87C654 is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}/V_{PP}$ | 31 | 35 | 29 | I | **External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H and 3FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 3FFFH. This pin also receives the 12.75V programming supply voltage ($V_{PP}$) during EPROM programming. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTE:**
To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher than $V_{CC}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

**Table 1. 8XC652/654 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes) | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 | 0x000000B |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP*# | Interrupt priority | B8H | – | | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 | xx000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1*# | Port 1 | 90H | SDA | SCL | | | | | | | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TXD | RXD | FFH |
| PCON# | Power control | 87H | SMOD | – | – | – | GF1 | GF0 | PD | IDL | 0xxx0000B |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| S0CON*# | Serial 0 port control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| S0BUF# | Serial 0 data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | F1 | P | 00H |
| S1DAT# | Serial 1 data | DAH | | | | | | | | | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| S1ADR# | Serial 1 address | DBH | SLAVE ADDRESS | | | | | | | GC | 00H |
| | | | | | | | | | | | |
| S1STA# | Serial 1 status | D9H | SC4 | SC3 | SC2 | SC1 | SC0 | 0 | 0 | 0 | F8H |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| S1CON*# | Serial 1 control | D8H | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 | 00000000B |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| TH1 | Timer high 1 | 8DH | | | | | | | | | 00H |
| TH0 | Timer high 0 | 8CH | | | | | | | | | 00H |
| TL1 | Timer low 1 | 8BH | | | | | | | | | 00H |
| TL0 | Timer low 0 | 8AH | | | | | | | | | 00H |
| TMOD | Timer mode | 89H | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the Logic Symbol.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

### Reset

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up.

### Idle Mode

In the idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

### Power-Down Mode

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. The control bits for the reduced power modes are in the special function register PCON. Table 2 shows the state of the I/O ports during low current operating modes.

## I²C SERIAL COMMUNICATION—SIO1

The I²C serial port is identical to the I²C serial port on the 8XC552. The operation of this subsystem is described in detail in the 8XC552 section of this manual.

Note that in both the 8XC652/4 and the 8XC552 the I²C pins are alternate functions to port pins P1.6 and P1.7. Because of this, P1.6 and P1.7 on these parts do not have a pull-up structure as found on the 80C51. Therefore P1.6 and P1.7 have open drain outputs on the 8XC652/4.

**Table 2. External Pin Status During Idle and Power-Down Mode**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

**Serial Control Register (S1CON) – See Table 3**

**S1CON (D8H)**

| CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |
|---|---|---|---|---|---|---|---|

**Bits CR0, CR1 and CR2 determine the serial clock frequency that is generated in the master mode of operation.**

**Table 3. Serial Clock Rates**

| CR2 | CR1 | CR0 | BIT FREQUENCY (kHz) AT $f_{OSC}$ 6MHz | 12MHz | 16MHz | 20MHz | $f_{OSC}$ DIVIDED BY |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 23 | 47 | 62.5 | 78 | 256 |
| 0 | 0 | 1 | 27 | 54 | 71 | 89[1] | 224 |
| 0 | 1 | 0 | 31.25 | 62.5 | 83.3 | 104[1] | 192 |
| 0 | 1 | 1 | 37 | 75 | 100 | 125[1] | 160 |
| 1 | 0 | 0 | 6.25 | 12.5 | 17 | 21 | 960 |
| 1 | 0 | 1 | 50 | 100 | 133[1] | 166[1] | 120 |
| 1 | 1 | 0 | 100 | 200[1] | 267[1] | 334[1] | 60 |
| 1 | 1 | 1 | 0.25 < 62.5<br>0 to 255 | 0.5 < 62.5<br>0 to 254 | 0.65 < 55.6<br>0 to 253 | 0.81 < 69.4<br>0 to 253 | 96 × (256 – (reload value Timer 1))<br>(Reload value range: 0 – 254 in mode 2) |

**NOTES:**
1. These frequencies exceed the upper limit of 100kHz of the I²C-bus specification and cannot be used in an I²C-bus application.

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage on $\overline{EA}/V_{PP}$ to $V_{SS}$ | –0.5 to + 13 | V |
| Voltage on any other pin to $V_{SS}$ | –0.5 to + 6.5 | V |
| Input, output current on any single pin | ±5 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1 | W |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DEVICE SPECIFICATIONS

| TYPE | SUPPLY VOLTAGE (V) | | FREQUENCY (MHz) | | TEMPERATURE RANGE (°C) |
|---|---|---|---|---|---|
| | MIN. | MAX. | MIN. | MAX. | |
| S87C654–4 | 4.5 | 5.5 | 3.5 | 16 | 0 to +70 |
| S87C654–5 | 4.5 | 5.5 | 3.5 | 16 | –40 to +85 |
| S87C654–7 | 4.5 | 5.5 | 3.5 | 20 | 0 to +70 |
| S87C654–8 | 4.5 | 5.5 | 3.5 | 20 | –40 to +85 |

## DC ELECTRICAL CHARACTERISTICS

$V_{SS}$ = 0V

| SYMBOL | PARAMETER | PART TYPE | TEST CONDITIONS | LIMITS MIN. | LIMITS MAX. | UNIT |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input low voltage,<br>except $\overline{EA}$, P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C | | –0.5<br>–0.5 | $0.2V_{CC}$–0.1<br>$0.2V_{CC}$–0.15 | V<br>V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | 0 to +70°C<br>–40 to +85°C | | –0.5<br>–0.5 | $0.2V_{CC}$–0.3<br>$0.2V_{CC}$–0.35 | V<br>V |
| $V_{IL2}$ | Input low voltage to P1.6/SCL, P1.7/SDA[1] | | | –0.5 | $0.3V_{CC}$ | V |
| $V_{IH}$ | Input high voltage,<br>except XTAL1, RST, P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C | | $0.2V_{CC}$+0.9<br>$0.2V_{CC}$+1.0 | $V_{CC}$+0.5<br>$V_{CC}$+0.5 | V<br>V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | 0 to +70°C<br>–40 to +85°C | | $0.7V_{CC}$<br>$0.7V_{CC}$+0.1 | $V_{CC}$+0.5<br>$V_{CC}$+0.5 | V<br>V |
| $V_{IH2}$ | Input high voltage, P1.6/SCL, P1.7/SDA[1] | | | $0.7V_{CC}$ | 6.0 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3,<br>except P1.6/SCL, P1.7/SDA | | $I_{OL}$ = 1.6mA[2, 3] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$ | | $I_{OL}$ = 3.2mA[2, 3] | | 0.45 | V |
| $V_{OL2}$ | Output low voltage, P1.6/SCL, P1.7/SDA | | $I_{OL}$ = 3.0mA | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3 | 0 to +70°C<br>–40 to +85°C | $I_{OH}$ = –60µA<br>$I_{OH}$ = –25µA | 2.4<br>$0.75V_{CC}$ | | V<br>V |
| $V_{OH1}$ | Output high voltage; port 0 in external bus mode,<br>ALE, $\overline{PSEN}$, RST[4] | 0 to +70°C<br>–40 to +85°C | $I_{OH}$ = –400µA<br>$I_{OH}$ = –150µA | 2.4<br>$0.75V_{CC}$ | | V<br>V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3, 4,<br>except P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C | $V_{IN}$ = 0.45V | | –50<br>–75 | µA<br>µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3,<br>except P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C | See note 5 | | –650<br>–750 | µA<br>µA |
| $I_{L1}$ | Input leakage current, port 0 | | 0.45V < $V_I$ < $V_{CC}$ | | ±10 | µA |
| $I_{L2}$ | Input leakage current, P1.6/SCL, P1.7/SDA | | 0V < $V_I$ < 6.0V<br>0V < $V_{CC}$ < 6.0V | | ±10 | µA<br>µA |
| $I_{CC}$ | Power supply current:<br><br>Active mode @ 16MHz[7]<br>Idle mode @ 16MHz[8]<br>Power down mode[9, 10]<br>Power down mode[9, 10] | <br><br><br><br>0 to +70°C<br>–40 to +85°C | See note 6<br>$V_{CC}$=6.0V | | <br><br>25<br>6<br>50<br>135 | <br><br>mA<br>mA<br>µA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | | 50 | 150 | kΩ |
| $C_{IO}$ | Pin capacitance | | Freq.=1MHz | | 10 | pF |

**NOTES: See Next Page.**

**NOTES FOR DC ELECTRICAL CHARACTERISTICS:**

1. The input threshold voltage of P1.6 and P1.7 (SIO1) meets the $I^2C$ specification, so an input voltage below $0.3V_{CC}$ will be recognized as a logic 0 while an input voltage above $0.7V_{CC}$ will be recognized as a logic 1.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5mA and no more than two outputs exceed the test conditions.
3. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows: Maximum $I_{OL}$ = 10mA per port pin; Maximum $I_{OL}$ = 26mA total for Port 0; Maximum $I_{OL}$ = 15mA total for Ports 1, 2, and 3; Maximum $I_{OL}$ = 71mA total for all output pins. If $I_{OL}$ exceeds the test conditions, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
4. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address bits are stabilizing.
5. Pins of ports 1 , 2, and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
6. See Figures 9 through 11 for $I_{CC}$ test conditions.
7. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 10ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{CC} - 0.5V$; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = P1.6 = P1.7 = $V_{CC}$; $f_{CLK}$ = 16MHz. See Figure 9.
8. The idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 10ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{CC} - 0.5V$; XTAL2 not connected; Port 0 = P1.6 = P1.7 = $V_{CC}$; $\overline{EA}$ = RST = $V_{SS}$; $f_{CLK}$ = 16MHz. See Figure 10.
9. The power-down current is measured with all output pins disconnected; XTAL2 not connected; Port 0 = P1.6 = P1.7 = $V_{CC}$; $\overline{EA}$ = RST = $V_{SS}$. See Figure 11.
10. $2V \leq V_{PD} \leq V_{CC}$max.

## AC ELECTRICAL CHARACTERISTICS[1, 2]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 2 | Oscillator frequency **Speed Versions** 87C654 –4, –5 | | | 3.5 | 16 | MHz |
| $t_{LHLL}$ | 2 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 2 | Address valid to ALE low | 8 | | $t_{CLCL}-55$ | | ns |
| $t_{LLAX}$ | 2 | Address hold after ALE low | 28 | | $t_{CLCL}-35$ | | ns |
| $t_{LLIV}$ | 2 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 2 | ALE low to $\overline{PSEN}$ low | 23 | | $t_{CLCL}-40$ | | ns |
| $t_{PLPH}$ | 2 | $\overline{PSEN}$ pulse width | 143 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 2 | $\overline{PSEN}$ low to valid instruction in | | 83 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 2 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 2 | Input instruction float after $\overline{PSEN}$ | | 38 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 2 | Address to valid instruction in | | 208 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 2 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{AVLL}$ | 3, 4 | Address valid to ALE low | 28 | | $t_{CLCL}-35$ | | ns |
| $t_{RLRH}$ | 3, 4 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 3, 4 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 3, 4 | $\overline{RD}$ low to valid data in | | 148 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 3, 4 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 3, 4 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-70$ | ns |
| $t_{LLDV}$ | 3, 4 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 3, 4 | Address to valid data in | | 398 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 3, 4 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 138 | 238 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 3, 4 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 120 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 3, 4 | Data valid to $\overline{WR}$ transition | 3 | | $t_{CLCL}-60$ | | ns |
| $t_{DW}$ | 3, 4 | Data setup time before $\overline{WR}$ | 288 | | $7t_{CLCL}-150$ | | ns |
| $t_{WHQX}$ | 3, 4 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{RLAZ}$ | 3, 4 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 3, 4 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 5 | Serial port clock cycle time[3] | 0.75 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 5 | Output data setup to clock rising edge[3] | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 5 | Output data hold after clock rising edge[3] | 80 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 5 | Input data hold after clock rising edge[3] | 0 | | 0 | | ns |
| $t_{XHDV}$ | 5 | Clock rising edge to input data valid[3] | | 492 | | $10t_{CLCL}-133$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 6 | High time[3] | 20 | | 20 | $t_{CLCL}-t_{LOW}$ | ns |
| $t_{CLCX}$ | 6 | Low time[3] | 20 | | 20 | $t_{CLCL}-t_{HIGH}$ | ns |
| $t_{CLCH}$ | 6 | Rise time[3] | | 20 | | 20 | ns |
| $t_{CHCL}$ | 6 | Fall time[3] | | 20 | | 20 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. These values are characterized but not 100% production tested.

## AC ELECTRICAL CHARACTERISTICS[1, 2]

| SYMBOL | FIGURE | PARAMETER | 20MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 2 | Oscillator frequency: **Speed Versions** 87C654 **–7, –8** | | | 3.5 | 20 | MHz |
| $t_{LHLL}$ | 2 | ALE pulse width | 60 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 2 | Address valid to ALE low | 25 | | $t_{CLCL}-25$ | | ns |
| $t_{LLAX}$ | 2 | Address hold after ALE low | 25 | | $t_{CLCL}-25$ | | ns |
| $t_{LLIV}$ | 2 | ALE low to valid instruction in | | 135 | | $4t_{CLCL}-65$ | ns |
| $t_{LLPL}$ | 2 | ALE low to $\overline{PSEN}$ low | 25 | | $t_{CLCL}-25$ | | ns |
| $t_{PLPH}$ | 2 | $\overline{PSEN}$ pulse width | 105 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 2 | $\overline{PSEN}$ low to valid instruction in | | 90 | | $3t_{CLCL}-60$ | ns |
| $t_{PXIX}$ | 2 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 2 | Input instruction float after $\overline{PSEN}$ | | 25 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 2 | Address to valid instruction in | | 170 | | $5t_{CLCL}-80$ | ns |
| $t_{PLAZ}$ | 2 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{AVLL}$ | 3, 4 | Address valid to ALE low | 25 | | $t_{CLCL}-25$ | | ns |
| $t_{RLRH}$ | 3, 4 | $\overline{RD}$ pulse width | 200 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 3, 4 | $\overline{WR}$ pulse width | 200 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 3, 4 | $\overline{RD}$ low to valid data in | | 160 | | $5t_{CLCL}-90$ | ns |
| $t_{RHDX}$ | 3, 4 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 3, 4 | Data float after $\overline{RD}$ | | 72 | | $2t_{CLCL}-28$ | ns |
| $t_{LLDV}$ | 3, 4 | ALE low to valid data in | | 250 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 3, 4 | Address to valid data in | | 285 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 3, 4 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 100 | 200 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 3, 4 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 125 | | $4t_{CLCL}-75$ | | ns |
| $t_{QVWX}$ | 3, 4 | Data valid to $\overline{WR}$ transition | 20 | | $t_{CLCL}-30$ | | ns |
| $t_{DW}$ | 3, 4 | Data setup time before $\overline{WR}$ | 220 | | $7t_{CLCL}-130$ | | ns |
| $t_{WHQX}$ | 3, 4 | Data hold after $\overline{WR}$ | 25 | | $t_{CLCL}-25$ | | ns |
| $t_{RLAZ}$ | 3, 4 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 3, 4 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 25 | 75 | $t_{CLCL}-25$ | $t_{CLCL}+25$ | ns |
| **Shift Register** | | | | | | | |
| $t_{XLXL}$ | 5 | Serial port clock cycle time[3] | 0.6 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 5 | Output data setup to clock rising edge[3] | 367 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 5 | Output data hold after clock rising edge[3] | 40 | | $2t_{CLCL}-60$ | | ns |
| $t_{XHDX}$ | 5 | Input data hold after clock rising edge[3] | 0 | | 0 | | ns |
| $t_{XHDV}$ | 5 | Clock rising edge to input data valid[3] | | 367 | | $10t_{CLCL}-133$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 6 | High time[3] | 17 | | 17 | $t_{CLCL}-t_{LOW}$ | ns |
| $t_{CLCX}$ | 6 | Low time[3] | 17 | | 17 | $t_{CLCL}-t_{HIGH}$ | ns |
| $t_{CLCH}$ | 6 | Rise time[3] | | 20 | | 20 | ns |
| $t_{CHCL}$ | 6 | Fall time[3] | | 20 | | 20 | ns |

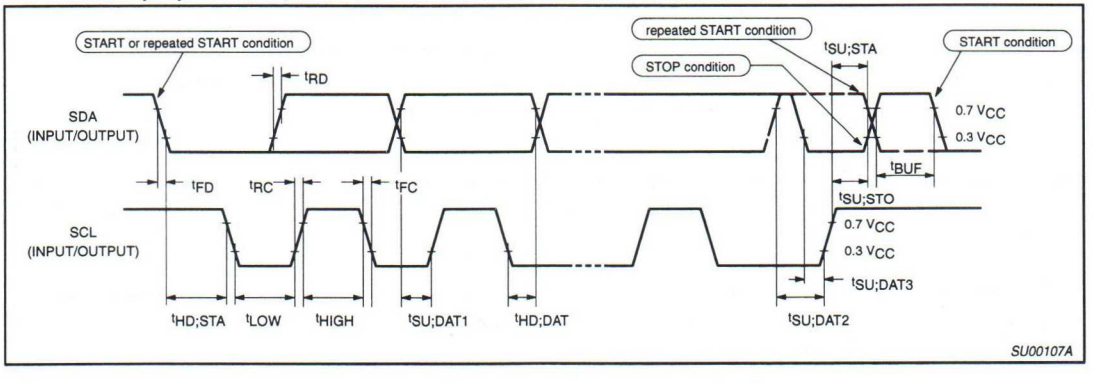
**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. These values are characterized but not 100% production tested.

## AC ELECTRICAL CHARACTERISTICS – I²C INTERFACE

| SYMBOL | PARAMETER | INPUT | OUTPUT |
|---|---|---|---|
| **SCL TIMING CHARACTERISTICS** | | | |
| $t_{HD;STA}$ | START condition hold time | ≥ 14 $t_{CLCL}$ | > 4.0µs[1] |
| $t_{LOW}$ | SCL LOW time | ≥ 16 $t_{CLCL}$ | > 4.7µs[1] |
| $t_{HIGH}$ | SCL HIGH time | ≥ 14 $t_{CLCL}$ | > 4.0µs[1] |
| $t_{RC}$ | SCL rise time | ≤ 1µs | – [2] |
| $t_{FC}$ | SCL fall time | ≤ 0.3µs | < 0.3µs[3] |
| **SDA TIMING CHARACTERISTICS** | | | |
| $t_{SU;DAT1}$ | Data set-up time | ≥ 250ns | > 20 $t_{CLCL}$ – $t_{RD}$ |
| $t_{SU;DAT2}$ | SDA set-up time (before rep. START cond.) | ≥ 250ns | > 1µs[1] |
| $t_{SU;DAT3}$ | SDA set-up time (before STOP cond.) | ≥ 250ns | > 8 $t_{CLCL}$ |
| $t_{HD;DAT}$ | Data hold time | ≥ 0ns | > 8 $t_{CLCL}$ – $t_{FC}$ |
| $t_{SU;STA}$ | Repeated START set-up time | ≥ 14 $t_{CLCL}$ | > 4.7µs[1] |
| $t_{SU;STO}$ | STOP condition set-up time | ≥ 14 $t_{CLCL}$ | > 4.0µs[1] |
| $t_{BUF}$ | Bus free time | ≥ 14 $t_{CLCL}$ | > 4.7µs[1] |
| $t_{RD}$ | SDA rise time | ≤ 1µs | – [2] |
| $t_{FD}$ | SDA fall time | ≤ 0.3µs | < 0.3µs[3] |

**NOTES:**
1. At 100 kbit/s. At other bit rates this value is inversely proportional to the bit-rate of 100 kbit/s.
2. Determined by the external bus-line capacitance and the external bus-line pull-resistor, this must be < 1µs.
3. Spikes on the SDA and SCL lines with a duration of less than 3 $t_{CLCL}$ will be filtered out. Maximum capacitance on bus-lines SDA and SCL = 400pF.
4. $t_{CLCL} = 1/f_{OSC}$ = one oscillator clock period at pin XTAL1. For 62ns < $t_{CLCL}$ < 285ns (16MHz) > $f_{OSC}$ > 3.5MHz) the SI01 interface meets the I²C-bus specification for bit-rates up to 100 kbit/s.

## TIMING SIO1 (I²C) INTERFACE



## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{PSEN}$ low.

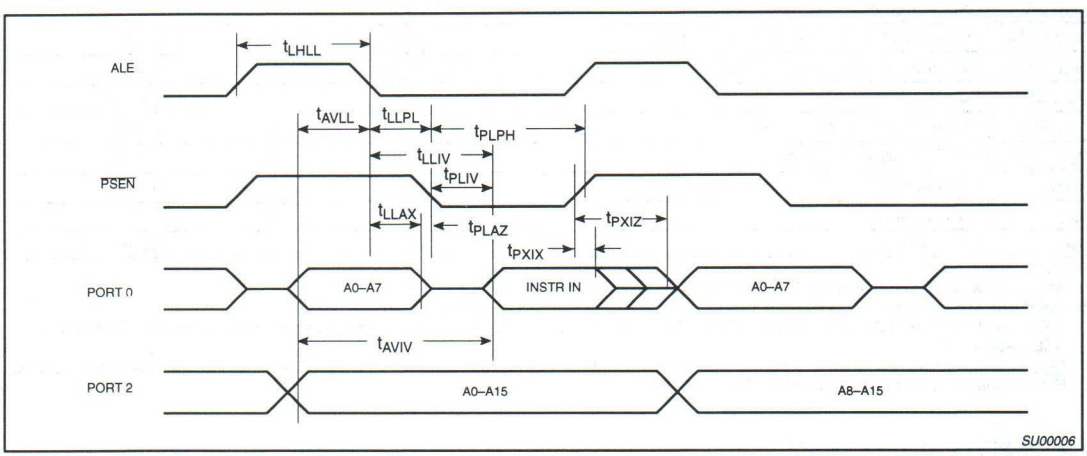



**Figure 1. External Program Memory Read Cycle**

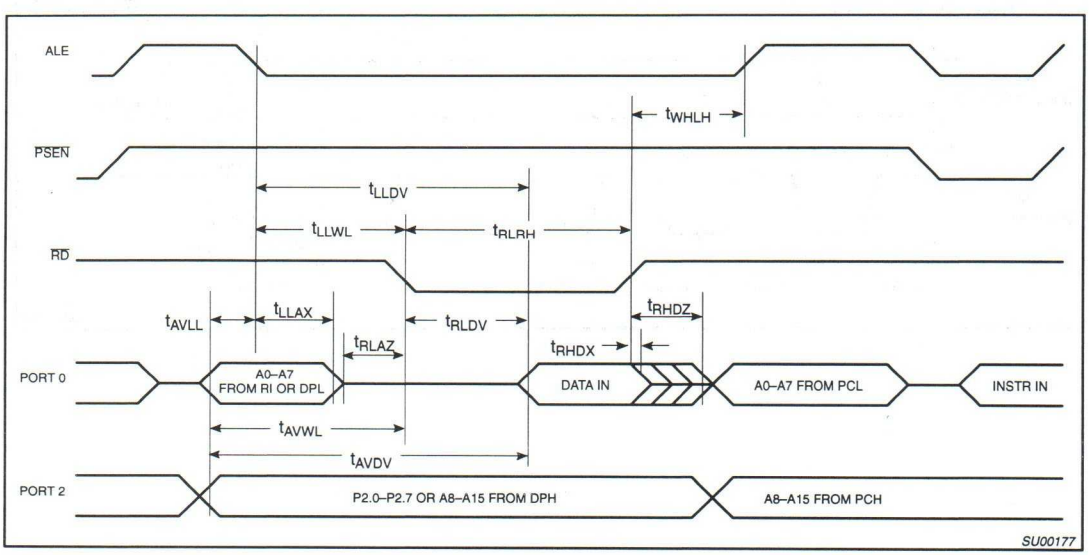



**Figure 2. External Data Memory Read Cycle**

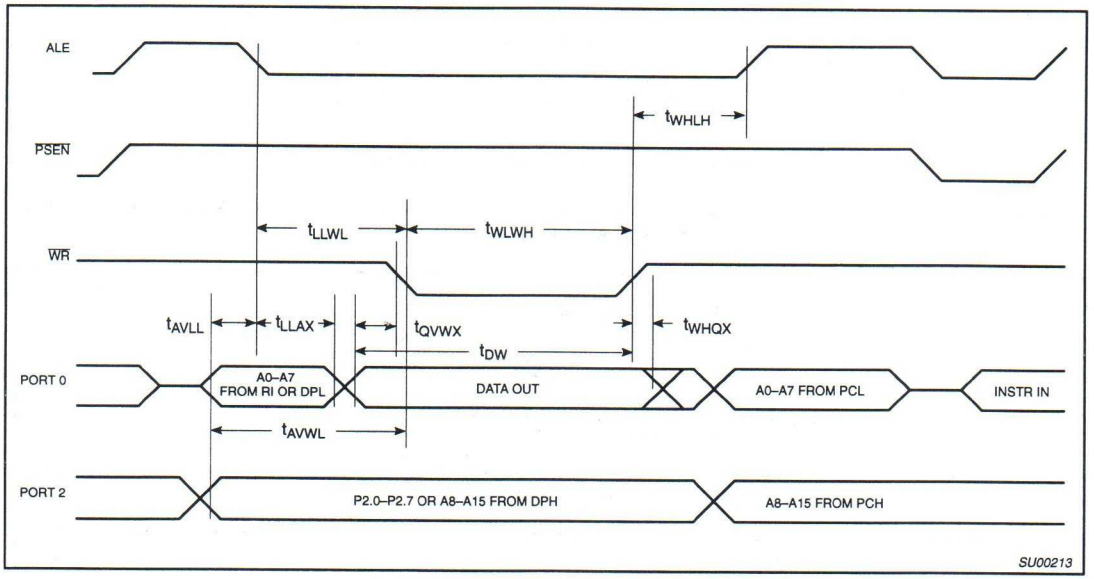



**Figure 3. External Data Memory Write Cycle**

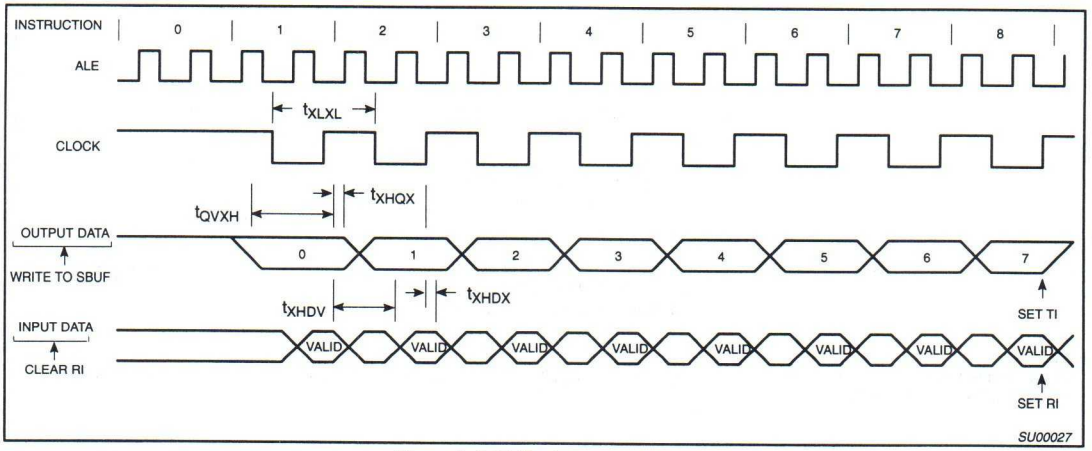



**Figure 4. Shift Register Mode Timing**

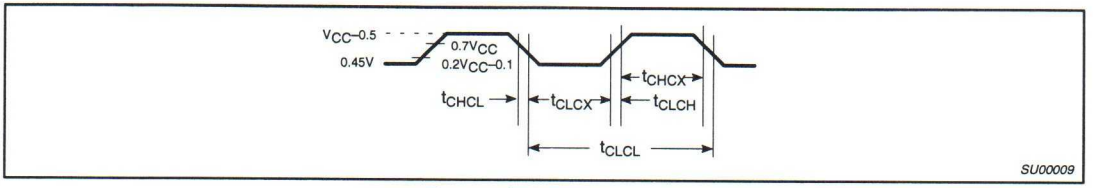



**Figure 5. External Clock Drive**

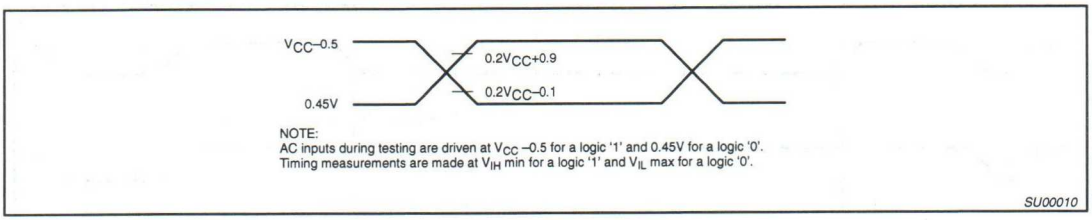



**Figure 6. AC Testing Input/Output**

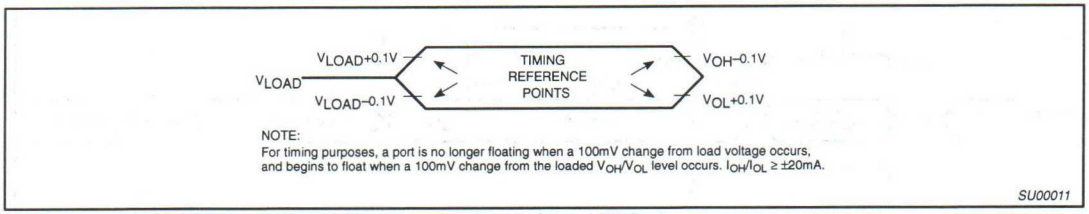



**Figure 7. Float Waveform**

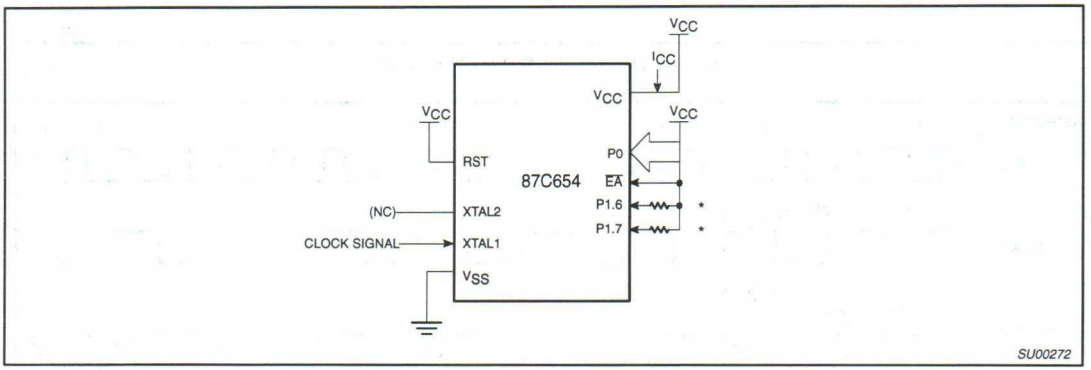



**Figure 8. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

**NOTE:**
* Ports 1.6 and 1.7 should be connected to $V_{CC}$ through resistors of sufficiently high value such that the sink current into these pins does not exceed the $I_{OL1}$ specification.

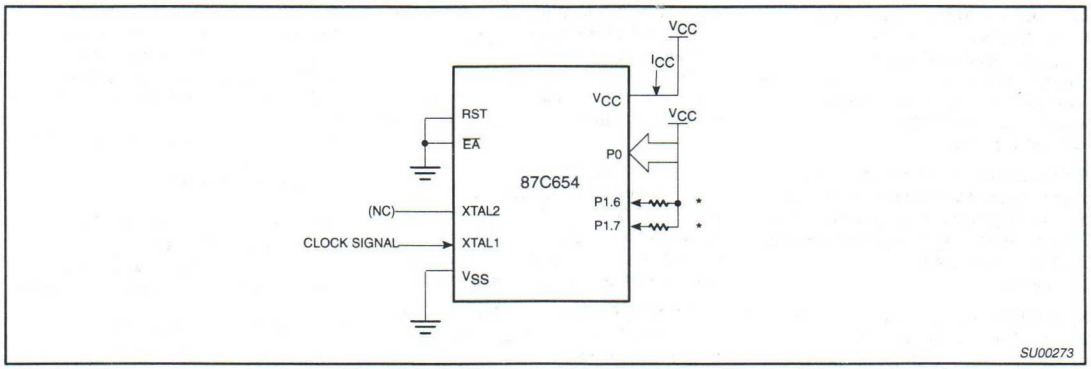



**Figure 9. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

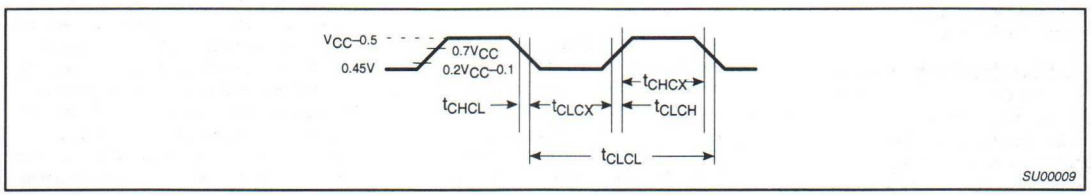



**Figure 10. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 10ns$**

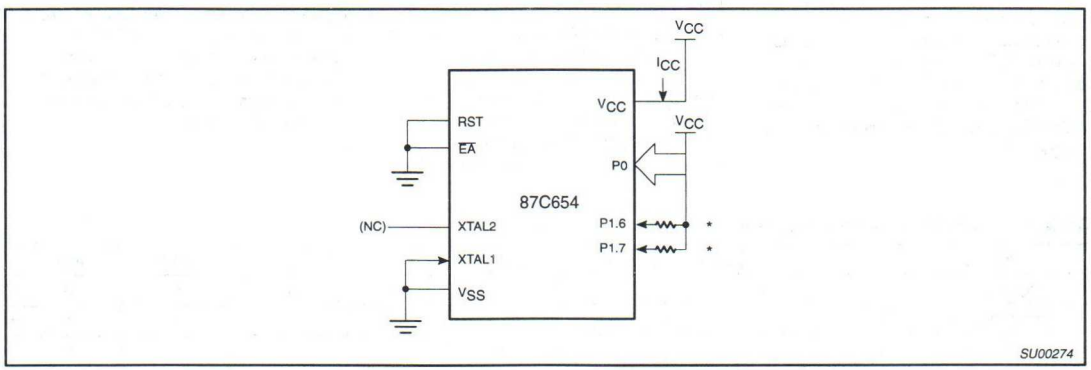



**Figure 11. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2V to 5.5V**

**NOTE:**

* Ports 1.6 and 1.7 should be connected to $V_{CC}$ through resistors of sufficiently high value such that the sink current into these pins does not exceed the $I_{OL1}$ specification.

## EPROM CHARACTERISTICS

The 87C654 is programmed by using a modified Quick-Pulse Programming™ algorithm. It differs from older methods in the value used for $V_{PP}$ (programming supply voltage) and in the width and number of the ALE/$\overline{PROG}$ pulses.

The 87C654 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an 87C654 manufactured by Philips Components.

Table 4 shows the logic levels for reading the signature byte, and for programming the program memory, the encryption table, and the lock bits. The circuit configuration and waveforms for quick-pulse programming are shown in Figures 12 and 13. Figure 14 shows the circuit configuration for normal program memory verification.

## Quick-Pulse Programming

The setup for microcontroller quick-pulse programming is shown in Figure 12. Note that the 87C654 is running with a 4 to 6MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to ports 1 and 2, as shown in Figure 12. The code byte to be programmed into that location is applied to port 0. RST, $\overline{PSEN}$ and pins of ports 2 and 3 specified in Table 4 are held at the 'Program Code Data' levels indicated in Table 4. The ALE/$\overline{PROG}$ is pulsed low 25 times as shown in Figure 13.

To program the encryption table, repeat the 25 pulse programming sequence for addresses 0 through 1FH, using the 'Pgm Encryption Table' levels. Do not forget that after the encryption table is programmed, verification cycles will produce only encrypted data.

To program the lock bits, repeat the 25 pulse programming sequence using the 'Pgm Lock Bit' levels. After one lock bit is programmed, further programming of the code memory and encryption table is disabled. However, the other lock bit can still be programmed.

Note that the $\overline{EA}$/$V_{PP}$ pin must not be allowed to go above the maximum specified $V_{PP}$ level for any amount of time. Even a narrow glitch above that voltage can cause permanent damage to the device. The $V_{PP}$ source should be well regulated and free of glitches and overshoot.

### Program Verification

If lock bit 2 has not been programmed, the on-chip program memory can be read out for program verification. The address of the program memory locations to be read is applied to ports 1 and 2 as shown in Figure 14. The other pins are held at the 'Verify Code Data' levels indicated in Table 4. The contents of the address location will be emitted on port 0. External pull-ups are required on port 0 for this operation.

If the encryption table has been programmed, the data presented at port 0 will be the exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the encryption table contents in order to correctly decode the verification data. The encryption table itself cannot be read out.

### Reading the Signature Bytes

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values are:

(030H) = 15H indicates manufactured by Philips

(031H) = 99H indicates 87C654

## Program/Verify Algorithms

Any algorithm in agreement with the conditions listed in Table 4, and which satisfies the timing specifications, is suitable.

## Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Fluorglas part number 2345–5, or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-sec/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient. Erasure leaves the array in an all 1s state.

**Table 4. EPROM Programming Modes**

| MODE | RST | $\overline{PSEN}$ | ALE/$\overline{PROG}$ | $\overline{EA}$/$V_{PP}$ | P2.7 | P2.6 | P3.7 | P3.6 |
|---|---|---|---|---|---|---|---|---|
| Read signature | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Program code data | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 1 |
| Verify code data | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Pgm encryption table | 1 | 0 | 0* | $V_{PP}$ | 1 | 0 | 1 | 0 |
| Pgm lock bit 1 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 1 | 1 |
| Pgm lock bit 2 | 1 | 0 | 0* | $V_{PP}$ | 1 | 1 | 0 | 0 |

**NOTES:**

1. '0' = Valid low for that pin, '1' = valid high for that pin.
2. $V_{PP}$ = 12.75V ±0.25V.
3. $V_{CC}$ = 5V±10% during programming and verification.

* ALE/$\overline{PROG}$ receives 25 programming pulses while $V_{PP}$ is held at 12.75V. Each programming pulse is low for 100µs (±10µs) and high for a minimum of 10µs.

™Trademark phrase of Intel Corporation.

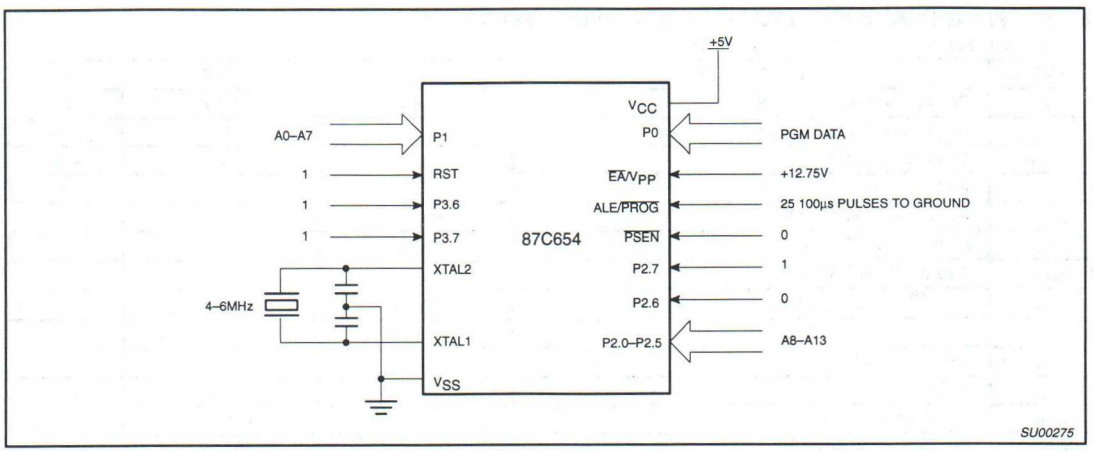



**Figure 12. Programming Configuration**

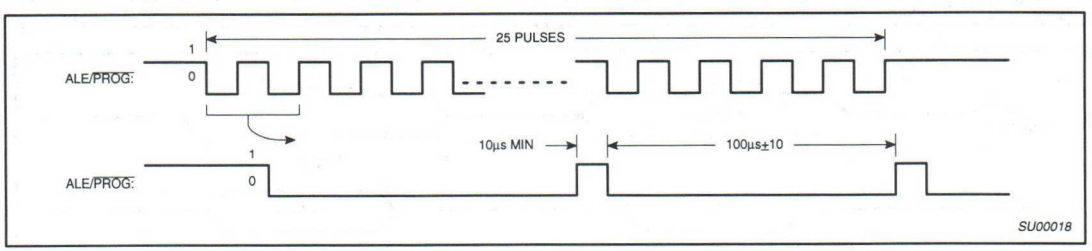



**Figure 13. $\overline{\text{PROG}}$ Waveform**

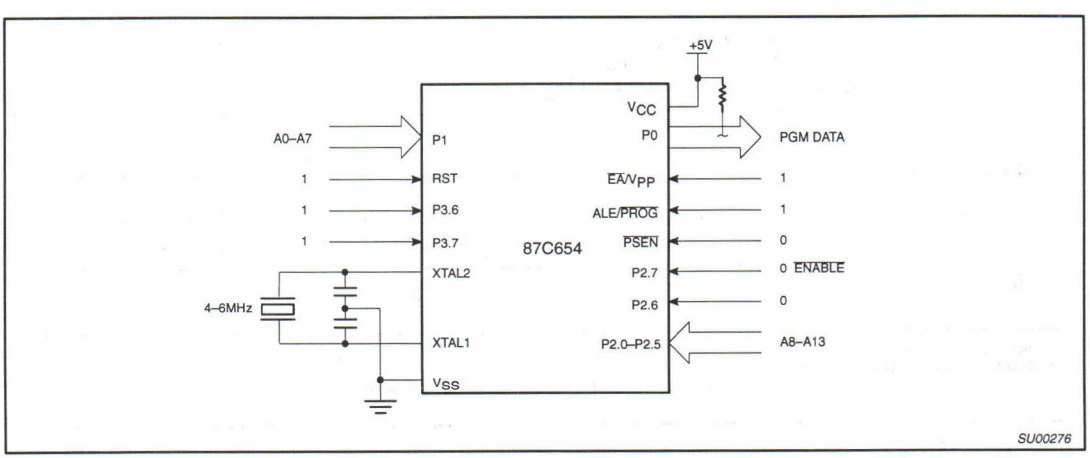
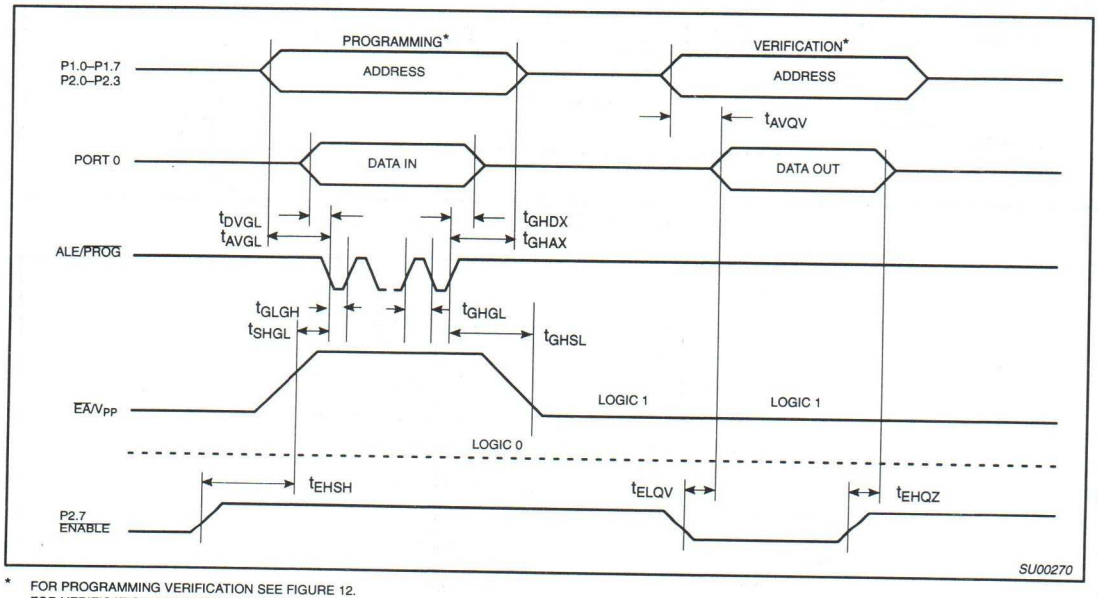



**Figure 14. Program Verification**

## EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V±10%, $V_{SS}$ = 0V (See Figure 15)

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $V_{PP}$ | Programming supply voltage | 12.5 | 13.0 | V |
| $I_{PP}$ | Programming supply current | | 50 | mA |
| $1/t_{CLCL}$ | Oscillator frequency | 4 | 6 | MHz |
| $t_{AVGL}$ | Address setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PROG}$ low | $48t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PROG}$ | $48t_{CLCL}$ | | |
| $t_{EHSH}$ | P2.7 ($\overline{ENABLE}$) high to $V_{PP}$ | $48t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PROG}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PROG}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PROG}$ width | 90 | 110 | µs |
| $t_{AVQV}$ | Address to data valid | | $48t_{CLCL}$ | |
| $t_{ELQZ}$ | $\overline{ENABLE}$ low to data valid | | $48t_{CLCL}$ | |
| $t_{EHQZ}$ | Data float after $\overline{ENABLE}$ | 0 | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PROG}$ high to $\overline{PROG}$ low | 10 | | µs |




* FOR PROGRAMMING VERIFICATION SEE FIGURE 12.
FOR VERIFICATION CONDITIONS SEE FIGURE 14.

**Figure 15. EPROM Programming and Verification**

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# CMOS single-chip 8-bit microcontroller with Electromagnetic Compatibility improvements

# 83CE654

## DESCRIPTION

The 83CE654 Single-Chip 8-Bit Microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The 83CE654 has the same instruction set as the 80C51. The 83CE654 has 16k bytes mask programmable ROM and 256 bytes RAM.

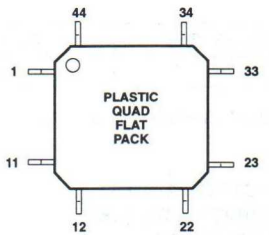
This device provides architectural enhancements that make it applicable in a variety of applications for general control systems. The 83CE654 contains a non-volatile 16k × 8 read-only program memory, a volatile 256 × 8 read/write data memory, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a multi-source, two-priority-level, nested interrupt structure, an $I^2C$ interface, UART and on-chip oscillator and timing circuits. For systems that require extra capability, the 83CE654 can be expanded using standard TTL compatible memories and logic.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte and 17 three-byte. With a 16MHz crystal, 58% of the instructions are executed in 0.75µs and 40% in 1.5µs. Multiply and divide instructions require 3µs.

## FEATURES

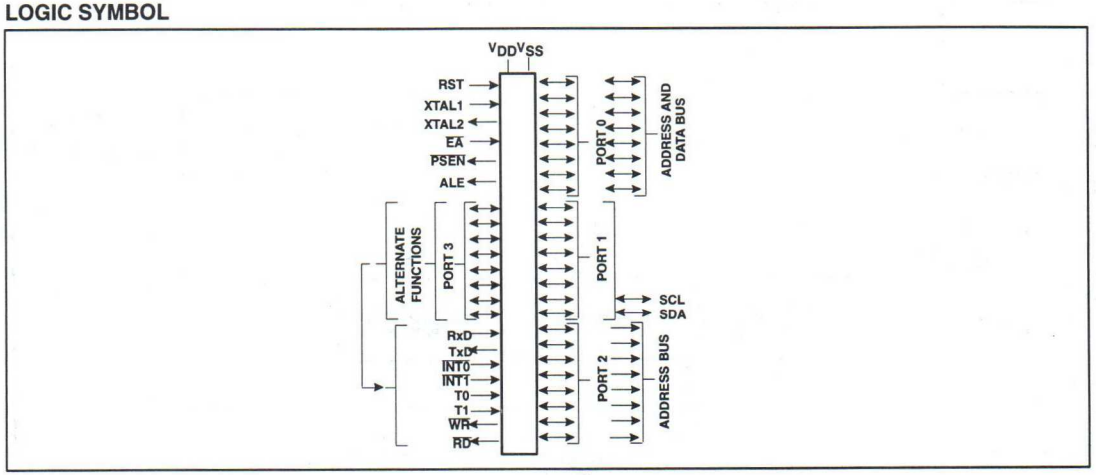
- 80C51 central processing unit
- 16k × 8 ROM expandable externally to 64k bytes
- 256 × 8 RAM, expandable externally to 64k bytes
- Two standard 16-bit timer/counters
- Four 8-bit I/O ports
- $I^2C$-bus serial I/O port with byte oriented master and slave functions
- Full-duplex UART facilities
- ROM code protection
- XTAL frequency range: 1.2MHz to 16MHz
- Software enable/disable of ALE output pulse
- Electromagnetic compatibility (EMC) improvements
- Operating ambient temperature range:
  - P83CE654 FBB $T_{amb}$ 0°C to +70°C
  - P83CE654 FFB $T_{amb}$ –40°C to +85°C

## PIN CONFIGURATION




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5 | 23 | P2.5/A13 |
| 2 | P1.6/SCL | 24 | P2.6/A14 |
| 3 | P1.7/SDA | 25 | P2.7/A15 |
| 4 | RST | 26 | $\overline{PSEN}$ |
| 5 | P3.0/RxD | 27 | ALE |
| 6 | $V_{SS4}$ | 28 | $V_{SS2}$ |
| 7 | P3.1/TxD | 29 | $\overline{EA}$ |
| 8 | P3.2/$\overline{INT0}$ | 30 | P0.7/AD7 |
| 9 | P3.3/$\overline{INT1}$ | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/$\overline{WR}$ | 34 | P0.3/AD3 |
| 13 | P3.7/$\overline{RD}$ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | $V_{SS1}$ | 38 | $V_{DD2}$ |
| 17 | $V_{DD1}$ | 39 | $V_{SS3}$ |
| 18 | P2.0/A8 | 40 | P1.0 |
| 19 | P2.1/A9 | 41 | P1.1 |
| 20 | P2.2/A10 | 42 | P1.2 |
| 21 | P2.3/A11 | 43 | P1.3 |
| 22 | P2.4/A12 | 44 | P1.4 |

## LOGIC SYMBOL



## ORDERING INFORMATION

| ROM | TEMPERATURE RANGE °C AND PACKAGE | FREQUENCY MHz | DRAWING NUMBER |
|---|---|---|---|
| P83CE654FBB | 0 to +70, Plastic Quad Flat Pack | 1.2 to 16 | SOT307-2[1] |
| P83CE654FFB | –40 to +85, Plastic Quad Flat Pack | 1.2 to 16 | SOT307-2[1] |

**NOTE:**
1. SOT311 replaced by SOT307-2.

## ELECTROMAGNETIC COMPATIBILITY (EMC) IMPROVEMENTS

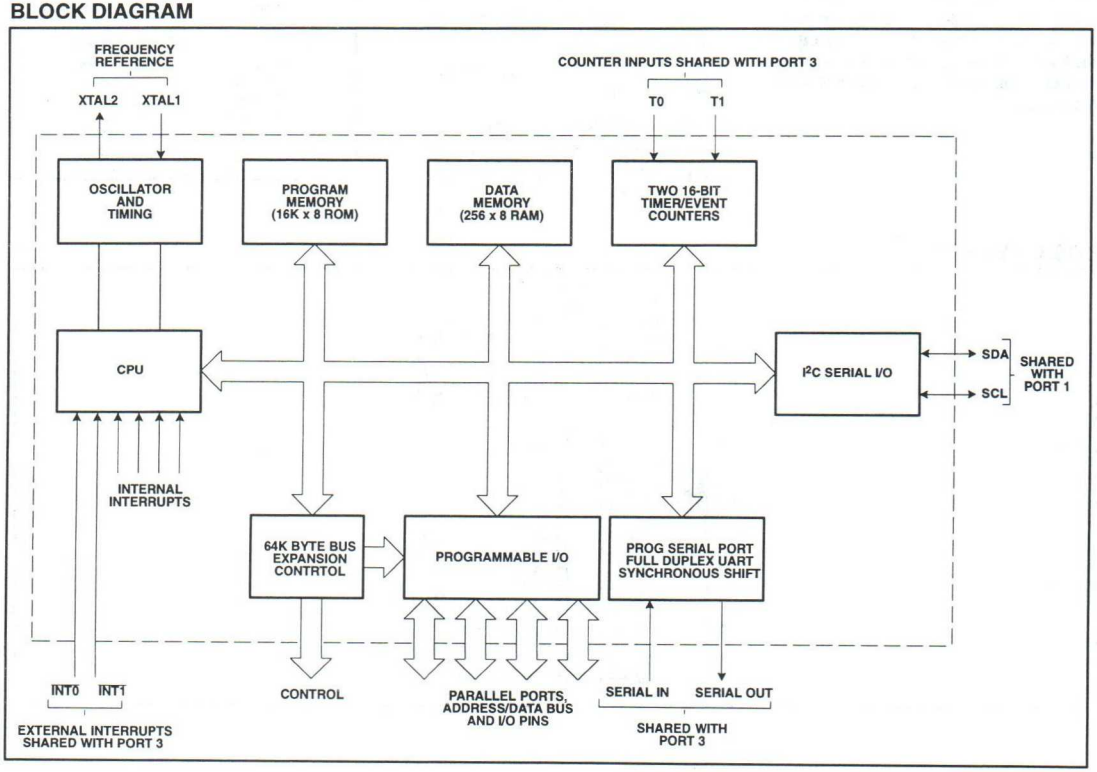
Primary attention is paid on the reduction of electromagnetic emission of the microcontroller P83CE654.

The following features effect in reducing the electromagnetic emission and additionally improve the electromagnetic susceptibility:

- Two supply voltage pins ($V_{DD1}$, $V_{DD2}$) and four ground pins ($V_{SS1}$ to $V_{SS4}$)
- Separate $V_{DD}$ pins for the internal logic and the port buffers
- Internal decoupling capacitance improves the EMC radiation behavior and the EMC immunity
- External capacitors are to be located as close as possible between pins $V_{DD2}$ and $V_{SS3}$ as well as $V_{DD1}$ and $V_{SS1}$; ceramic chip capacitors are recommended (100nF).
- The ALE output signal (pulses at a frequency of $f_{OSC}/6$) can be disabled under software control (bit 5 in the SFR PCON: "RFI"); if disabled, no ALE pulse will occur.

ALE pin will be pulled down internally, switching an external address latch to a quiet state. The MOVX instruction will still toggle ALE as a normal MOVX. ALE will retain its normal high value during Idle mode and a low value during Power-down mode while in the "RFI" reduction mode. Additionally during internal access ($\overline{EA}$ = 1) ALE will toggle normally when the address exceeds the internal program memory size. During external access ($\overline{EA}$ = 0) ALE will always toggle normally, whether the flag "RFI" is set or not.

## BLOCK DIAGRAM



## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER | TYPE | NAME AND FUNCTION |
|---|---|---|---|
| $V_{SS1}$, $V_{SS2}$, $V_{SS3}$, $V_{SS4}$ | 16, 28, 39, 6 | I | **Ground:** 0V reference. All pins must be connected. |
| $V_{DD1}$, $V_{DD2}$ | 17, 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. Both pins must be connected. |
| P0.0–0.7 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 can sink/source 8 LSTTL inputs. |
| P1.0–P1.7 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups, except P1.6 and P1.7 which are open drain. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also receives the low-order address byte during program memory verification. Alternate functions include: |
| P1.6 | 2 | I/O | **SCL:** $I^2C$-bus serial port clock line. |
| P1.7 | 3 | I/O | **SDA:** $I^2C$-bus serial port data line. |
| P2.0–P2.7 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |
| P3.0–P3.7 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the 80C51 family, as listed below: |
| | 5 | I | **RxD (P3.0):** Serial input port |
| | 7 | O | **TxD (P3.1):** Serial output port |
| | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt 0 or gate control input for timer/event counter 0 |
| | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt 1 or gate control input for timer/event counter 1 |
| | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal pull-down resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{DD}$. |
| ALE | 27 | I/O | **Address Latch Enable:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. ALE can sink/source 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up. To prohibit the toggling of ALE pin (RFI noise reduction) the bit RFI in the PCON Register (PCON.5) must be set by software. This bit is cleared on RESET and can be cleared by software. When set, ALE pin will be pulled down internally, switching an external address latch to a quiet state. The MOVX instruction will still toggle ALE as a normal MOVX. ALE will retain its normal high value during Idle mode and a low value during Power-down mode while in the "RFI" mode. Additionally during internal access ($\overline{EA}$ = 1) ALE will toggle normally when the address exceeds the internal program memory size. During external access ($\overline{EA}$ = 0) ALE will always toggle normally, whether the flag "RFI" is set or not. |
| $\overline{PSEN}$ | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the 83CE654 is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. $\overline{PSEN}$ can sink/source 8 LSTTL inputs. |
| $\overline{EA}$ | 29 | I | **External Access Enable:** when, during RESET, $\overline{EA}$ is held at a TTL HIGH level the CPU executes out of the internal program ROM, provided the program counter is less than 16384. When $\overline{EA}$ is held at a TTL LOW level during RESET, the CPU executes out of external program memory via Port 0 and Port 2. $\overline{EA}$ is not allowed to float. |
| XTAL1 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTE:**
To avoid "latch-up" effect at power-on, the voltage on any pin at any time must not be higher or lower than $V_{DD}$ + 0.5V or $V_{SS}$ – 0.5V, respectively.

**Table 1. 83CE654 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes) | | | | | | | | | | |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 | 0x000000B |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP*# | Interrupt priority | B8H | – | | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 | xx000000B |
| | | | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | |
| P0* | Port 0 | 80H | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | AD1 | AD0 | FFH |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1*# | Port 1 | 90H | SDA | SCL | | | | | | | FFH |
| | | | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | |
| P2* | Port 2 | A0H | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | FFH |
| | | | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | |
| P3* | Port 3 | B0H | $\overline{RD}$ | $\overline{WR}$ | T1 | T0 | $\overline{INT1}$ | $\overline{INT0}$ | TXD | RXD | FFH |
| PCON# | Power control | 87H | SMOD | – | RFI | – | GF1 | GF0 | PD | IDL | 0xxx0000B |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| S0CON*# | Serial 0 port control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| S0BUF# | Serial 0 data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | F1 | P | 00H |
| S1DAT# | Serial 1 data | DAH | | | | | | | | | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| S1ADR# | Serial 1 address | DBH | SLAVE ADDRESS | | | | | | | GC | 00H |
| | | | | | | | | | | | |
| S1STA# | Serial 1 status | D9H | SC4 | SC3 | SC2 | SC1 | SC0 | 0 | 0 | 0 | F8H |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| S1CON*# | Serial 1 control | D8H | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 | 00000000B |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| TH1 | Timer high 1 | 8DH | | | | | | | | | 00H |
| TH0 | Timer high 0 | 8CH | | | | | | | | | 00H |
| TL1 | Timer low 1 | 8BH | | | | | | | | | 00H |
| TL0 | Timer low 0 | 8AH | | | | | | | | | 00H |
| TMOD | Timer mode | 89H | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

## ROM CODE PROTECTION

The 83CE654 has an additional security feature. ROM code protection may be selected by setting a mask-programmable security bit (i.e., user dependent). This feature may be requested during ROM code submission. When selected, the ROM code is protected and cannot be read out at any time by any test mode or by any instruction in the external program memory space.

The MOVC instructions are the only instructions that have access to program code in the internal or external program memory. The $\overline{EA}$ input is latched during RESET and is "don't care" after RESET (also if the security bit is not set). This implementation prevents reading internal program code by switching from external program memory to internal program memory during a MOVC instruction or any other instruction that uses immediate data.

Table 2 lists the access to the internal and external program memory by the MOVC instructions when the security bit has been set to a logical "1":

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the Logic Symbol, page 3-1069.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

### Reset

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on $V_{DD}$ and RST must come up at the same time for a proper start-up.

### Power-on Reset (See Figure 1.)

When $V_{DD}$ is turned on, and provided its rise-time does not exceed 10ms, an automatic reset can be obtained by connecting the RST pin to $V_{DD}$ via a 2.2µF capacitor. When the power is switched on, the voltage on the RST pin is equal to $V_{DD}$ minus the capacitor voltage, and decreases from $V_{DD}$ as the capacitor charges through the internal resistor ($R_{RST}$) to ground. The larger the capacitor, the more slowly $V_{RST}$ decreases. $V_{RST}$ must remain above the lower threshold of the Schmitt trigger long enough to effect a complete reset. The time required is the oscillator start-up time, plus 2 machine cycles.

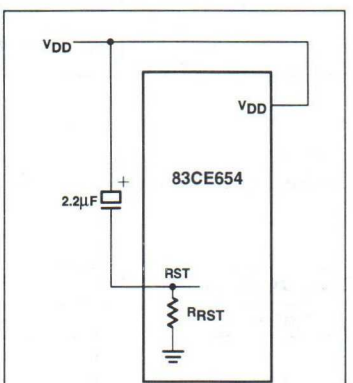



**Figure 1. Power-on Reset**

### Idle Mode

In the idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

### Power-Down Mode

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. the control bits for the reduced power modes are in the special function register PCON. Table 3 shows the state of the I/O ports during low current operating modes.

### Power Control Register PCON

These special modes are activated by software via the Special Function Register PCON. Its hardware address is 87H. PCON is not bit addressable. The reset value of PCON is (0x0x0000).

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PCON (87H) | SMOD | – | RFI | – | GF1 | GF0 | PD | IDL |

| Bit | Symbol | Function |
|---|---|---|
| PCON.7 | SMOD | Double Baud rate bit. When set to logic 1 the baud rate is doubled when Timer 1 is used to generate baud rate, and the Serial Port is used in modes 1, 2 or 3. |
| PCON.6 | – | (reserved for future use*) |
| PCON.5 | RFI | When set to logic 1 the toggling of ALE pin is prohibited. This bit is cleared on RESET. |
| PCON.4 | – | (reserved for future use*) |
| PCON.3 | GF1 | General purpose flag bit. |
| PCON.2 | GF0 | General purpose flag bit. |
| PCON.1 | PD | Power-down bit. Setting this bit activates Power-down mode. |
| PCON.0 | IDL | Idle mode bit. Setting this bit activates the Idle mode. If 1s are written to PD and IDL at the same time, PD takes precedence. |

**NOTE:**

* User software should not write 1s to reserved bits. These bits may be used in future 80C51 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

### I²C Serial Communication—SIO1

The I²C serial port is identical to the I²C serial port on the 8XC552. The operation of this subsystem is described in detail in the 8XC552 section of this manual.

Note that in both the 83CE654 and the 8XC552 the I²C pins are alternate functions to port pins P1.6 and P1.7. Because of this, P1.6 and P1.7 on these parts do not have a pull-up structure as found on the 80C51. Therefore P1.6 and P1.7 have open drain outputs on the 83CE654.

**Table 2.**

| | ACCESS TO INTERNAL PROGRAM MEMORY | ACCESS TO EXTERNAL PROGRAM MEMORY |
|---|---|---|
| MOVC in internal program memory | YES | YES |
| MOVC in external program memory | NO | YES |

**NOTE:**
If the security bit has been set to a logical 0, there are no restrictions for the MOVC instructions.

**Table 3. External Pin Status During Idle and Power-Down Mode**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

**Serial Control Register (S1CON) – See Table 4**

| S1CON (D8H) | CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |
|---|---|---|---|---|---|---|---|---|

**Bits CR0, CR1 and CR2 determine the serial clock frequency that is generated in the master mode of operation.**

**Table 4. Serial Clock Rates**

| CR2 | CR1 | CR0 | BIT FREQUENCY (kHz) AT $f_{OSC}$ 6MHz | 12MHz | 16MHz | $f_{OSC}$ DIVIDED BY |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 23 | 47 | 63 | 256 |
| 0 | 0 | 1 | 27 | 54 | 71 | 224 |
| 0 | 1 | 0 | 31 | 63 | 83 | 192 |
| 0 | 1 | 1 | 37 | 75 | 100 | 160 |
| 1 | 0 | 0 | 6.25 | 12.5 | 17 | 960 |
| 1 | 0 | 1 | 50 | 100 | 133[1] | 120 |
| 1 | 1 | 0 | 100 | 200[1] | 267[1] | 60 |
| 1 | 1 | 1 | 0.24 < 62.5 | 0.49 < 62.5 | 0.65 < 55.6 | 96 × (256 – (reload value Timer 1)) |
| | | | 0 < 255 | 0 < 254 | 0 < 253 | reload value range Timer 1 (in mode 2) |

**NOTES:**
1. These frequencies exceed the upper limit of 100kHz of the I²C-bus specification and cannot be used in an I²C-bus application.

## ABSOLUTE MAXIMUM RATINGS

| PARAMETER | RATING | UNIT |
|---|---|---|
| Voltage on $V_{DD}$ to $V_{SS}$ | –0.5 to +6.5 | V |
| Voltage on any pin to $V_{SS}$ | –0.5 to $V_{DD}$+0.5 | V |
| Storage temperature range | –65 to +150 | °C |
| Power dissipation (based on package heat transfer limitations, not device power consumption)[1] | 1 | W |
| Operating ambient temperature range | | |
| FBB | 0 to +70 | °C |
| FFB | –40 to +85 | °C |

**NOTE:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. this is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied. Exposure to Absolute Maximum Rating conditions for extended periods may affect device reliability.

## DEVICE SPECIFICATIONS

| TYPE | SUPPLY VOLTAGE (V) | | FREQUENCY (MHz) | | TEMPERATURE RANGE (°C) |
|---|---|---|---|---|---|
| | MIN. | MAX. | MIN. | MAX. | |
| P83CE654FBB | 4.5 | 5.5 | 1.2 | 16 | 0 to +70 |
| P83CE654FFB | 4.5 | 5.5 | 1.2 | 16 | –40 to +85 |

## DC ELECTRICAL CHARACTERISTICS

$V_{DD}$ = 5V (±10%), $V_{SS}$ = 0V, $T_{amb}$ = 0°C to +70°C or –40°C to +85°C

| SYMBOL | PARAMETER | PART TYPE | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|---|
| | | | | MIN. | MAX. | |
| $V_{IL}$ | Input low voltage, except $\overline{EA}$, P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C | | –0.5<br>–0.5 | $0.2V_{DD}$–0.1<br>$0.2V_{DD}$–0.15 | V<br>V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | 0 to +70°C<br>–40 to +85°C | | –0.5<br>–0.5 | $0.2V_{DD}$–0.3<br>$0.2V_{DD}$–0.35 | V<br>V |
| $V_{IL2}$ | Input low voltage to P1.6/SCL, P1.7/SDA[6] | | | –0.5 | $0.3V_{DD}$ | V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST, P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C | | $0.2V_{DD}$+0.9<br>$0.2V_{DD}$+1.0 | $V_{DD}$+0.5<br>$V_{DD}$+0.5 | V<br>V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | 0 to +70°C<br>–40 to +85°C | | $0.7V_{DD}$<br>$0.7V_{DD}$+0.1 | $V_{DD}$+0.5<br>$V_{DD}$+0.5 | V<br>V |
| $V_{IH2}$ | Input high voltage, P1.6/SCL, P1.7/SDA[6] | | | $0.7V_{DD}$ | 6.0 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3, except P1.6/SCL, P1.7/SDA[4], ALE, $\overline{PSEN}$ | | $I_{OL}$ = 1.6mA[7] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$[4] | | $I_{OL}$ = 3.2mA[7] | | 0.45 | V |
| $V_{OL2}$ | Output low voltage, P1.6/SCL, P1.7/SDA[4] | | $I_{OL}$ = 3.0mA[7] | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3, except P1.6, P1.7, ALE, $\overline{PSEN}$ | | $I_{OH}$ = –60µA;<br>$V_{DD}$ = 5V (± 10%)<br>$I_{OH}$ = –25µA<br>$I_{OH}$ = –10µA | 2.4<br><br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | V<br><br>V<br>V |
| $V_{OH1}$ | Output high voltage; port 0 in external bus mode[5] | | $I_{OH}$ = –800µA;<br>$V_{DD}$ = 5V (± 10%)<br>$I_{OH}$ = –300µA<br>$I_{OH}$ = –80µA | 2.4<br><br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | V<br><br>V<br>V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3, except P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C | $V_i$ = 0.45V<br>$V_i$ = 0.45V | | –50<br>–75 | µA<br>µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3, except P1.6/SCL, P1.7/SDA | 0 to +70°C<br>–40 to +85°C | $V_i$ = 2.0V<br>$V_i$ = 2.0V | | –650<br>–750 | µA<br>µA |
| $I_{LI1}$ | Input leakage current, port 0, $\overline{EA}$ | | 0.45V < $V_i$ < $V_{DD}$ | | ±10 | µA |
| $I_{LI2}$ | Input leakage current, P1.6/SCL, P1.7/SDA | | 0V < $V_i$ < 5.5V<br>0V < $V_{DD}$ < 5.5V | | ±10 | µA |
| $I_{DD}$ | Power supply current:<br>Active mode @ 16MHz[1, 8]<br>Idle mode @ 16MHz[2, 8]<br>Power down mode[3] | | <br>$V_{DD}$ = 5.5V<br>$V_{DD}$ = 5V ± 10%<br>@2V < $V_{PD}$ < $V_{DDMAX}$ | | <br>22<br>6<br>50 | <br>mA<br>mA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | | 50 | 150 | kΩ |
| $C_{IO}$ | Pin capacitance of I/O buffer | | Freq.=1MHz;<br>$T_{amb}$ = 25°C | | 10 | pF |

**NOTES: See Next Page.**

**NOTES FOR DC ELECTRICAL CHARACTERISTICS:**

1. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5V$; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = P1.6 = P1.7 = $V_{DD}$.
2. The idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5V$; XTAL2 not connected; Port 0 = P1.6 = P1.7 = $V_{DD}$; $\overline{EA}$ = RST = $V_{SS}$.
3. The power-down current is measured with all output pins disconnected; XTAL2 not connected; Port 0 = P1.6 = P1.7 = $V_{DD}$; $\overline{EA}$ = XTAL1 = RST = $V_{SS}$.
4. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the $V_{OL}$s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.
5. Capacitive loading on ports 0 and 2 may cause the $V_{OH}$ on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{DD}$ specification when the address bits are stabilizing.
6. The input threshold voltage of P1.6 and P1.7 (SIO1) meets the $I^2C$ specification, so an input voltage below $0.3V_{DD}$ will be recognized as a logic 0 while an input voltage above $0.7V_{DD}$ will be recognized as a logic 1.
7. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows: Maximum $I_{OL}$ = 10mA per port pin; Maximum $I_{OL}$ = 26mA total for Port 0; Maximum $I_{OL}$ = 15mA total for Ports 1, 2, and 3; Maximum $I_{OL}$ = 71mA total for all output pins. If $I_{OL}$ exceeds the test conditions, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
8. $I_{DDMAX}$ for the 80/83CE654 at the other frequencies can be derived from Figure 2, where FREQ is the external oscillator frequency in MHz. $I_{DDMAX}$ is given in mA.

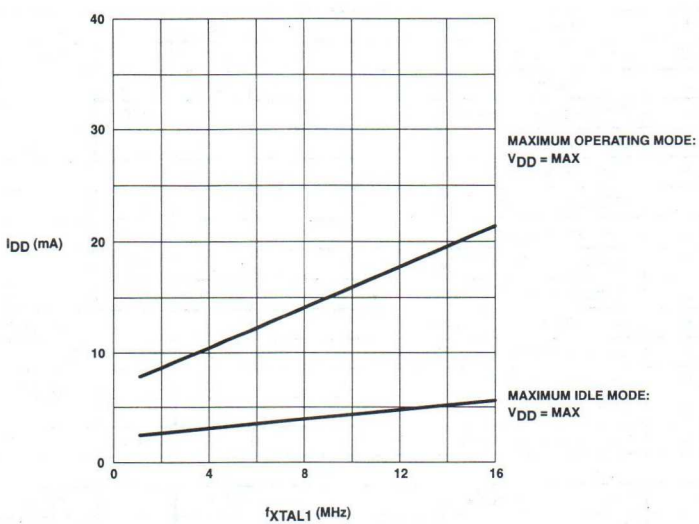



**Figure 2. $I_{DD}$ vs. Frequency**

## AC ELECTRICAL CHARACTERISTICS[1, 2]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 3 | Oscillator frequency | | | 1.2 | 16 | MHz |
| $t_{LHLL}$ | 3 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 3 | Address valid to ALE low | 8 | | $t_{CLCL}-55$ | | ns |
| $t_{LLAX}$ | 3 | Address hold after ALE low | 28 | | $t_{CLCL}-35$ | | ns |
| $t_{LLIV}$ | 3 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 3 | ALE low to $\overline{PSEN}$ low | 23 | | $t_{CLCL}-40$ | | ns |
| $t_{PLPH}$ | 3 | $\overline{PSEN}$ pulse width | 143 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 3 | $\overline{PSEN}$ low to valid instruction in | | 83 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 3 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 3 | Input instruction float after $\overline{PSEN}$ | | 38 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 3 | Address to valid instruction in | | 208 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 3 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{AVLL}$ | 4, 5 | Address valid to ALE low | 8 | | $t_{CLCL}-55$ | | ns |
| $t_{RLRH}$ | 4, 5 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 4, 5 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 4, 5 | $\overline{RD}$ low to valid data in | | 148 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 4, 5 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 4, 5 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-70$ | ns |
| $t_{LLDV}$ | 4, 5 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 4, 5 | Address to valid data in | | 398 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 4, 5 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 138 | 238 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AVWL}$ | 4, 5 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 120 | | $4t_{CLCL}-130$ | | ns |
| $t_{QVWX}$ | 4, 5 | Data valid to $\overline{WR}$ transition | 3 | | $t_{CLCL}-60$ | | ns |
| $t_{DW}$ | 4, 5 | Data setup time before $\overline{WR}$ | 288 | | $7t_{CLCL}-150$ | | ns |
| $t_{WHQX}$ | 4, 5 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{RLAZ}$ | 4, 5 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 4, 5 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **Shift Register[3]** | | | | | | | |
| $t_{XLXL}$ | 6 | Serial port clock cycle time | 0.75 | | $12t_{CLCL}$ | | µs |
| $t_{QVXH}$ | 6 | Output data setup to clock rising edge | 492 | | $10t_{CLCL}-133$ | | ns |
| $t_{XHQX}$ | 6 | Output data hold after clock rising edge | 80 | | $2t_{CLCL}-117$ | | ns |
| $t_{XHDX}$ | 6 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 6 | Clock rising edge to input data valid | | 492 | | $10t_{CLCL}-133$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 7 | High time | 20 | | 20 | $t_{CLCL}-t_{LOW}$ | ns |
| $t_{CLCX}$ | 7 | Low time | 20 | | 20 | $t_{CLCL}-t_{HIGH}$ | ns |
| $t_{CLCH}$ | 7 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 7 | Fall time | | 20 | | 20 | ns |

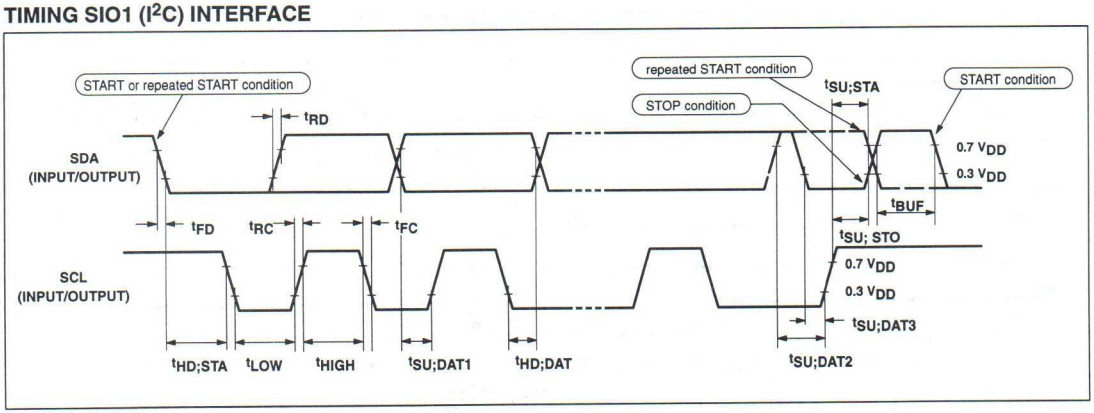
**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. Test condition: $T_{amb}$ = 0°C to +70C; $V_{DD}$ = 5V + 10%; $V_{SS}$ = 0V; load capacitance = 80pF.

## AC ELECTRICAL CHARACTERISTICS – I²C INTERFACE

| SYMBOL | PARAMETER | INPUT | OUTPUT |
|---|---|---|---|
| **SCL TIMING CHARACTERISTICS** | | | |
| $t_{HD;STA}$ | START condition hold time | ≥ 14 $t_{CLCL}$ | > 4.0µs[1] |
| $t_{LOW}$ | SCL LOW time | ≥ 16 $t_{CLCL}$ | > 4.7µs[1] |
| $t_{HIGH}$ | SCL HIGH time | ≥ 14 $t_{CLCL}$ | > 4.0µs[1] |
| $t_{RC}$ | SCL rise time | ≤ 1µs | –[2] |
| $t_{FC}$ | SCL fall time | ≤ 0.3µs | < 0.3µs[3] |
| **SDA TIMING CHARACTERISTICS** | | | |
| $t_{SU;DAT1}$ | Data set-up time | ≥ 250ns | > 20 $t_{CLCL}$ – $t_{RD}$ |
| $t_{SU;DAT2}$ | SDA set-up time (before rep. START cond.) | ≥ 250ns | > 1µs[1] |
| $t_{SU;DAT3}$ | SDA set-up time (before STOP cond.) | ≥ 250ns | > 8 $t_{CLCL}$ |
| $t_{HD;DAT}$ | Data hold time | ≥ 0ns | > 8 $t_{CLCL}$ – $t_{FC}$ |
| $t_{SU;STA}$ | Repeated START set-up time | ≥ 14 $t_{CLCL}$ | > 4.7µs[1] |
| $t_{SU;STO}$ | STOP condition set-up time | ≥ 14 $t_{CLCL}$ | > 4.0µs[1] |
| $t_{BUF}$ | Bus free time | ≥ 14 $t_{CLCL}$ | > 4.7µs[1] |
| $t_{RD}$ | SDA rise time | ≤ 1µs | –[2] |
| $t_{FD}$ | SDA fall time | ≤ 0.3µs | < 0.3µs[3] |

**NOTES:**
1. At 100 kbit/s. At other bit rates this value is inversely proportional to the bit-rate of 100 kbit/s.
2. Determined by the external bus-line capacitance and the external bus-line pull-resistor, this must be < 1µs.
3. Spikes on the SDA and SCL lines with a duration of less than 3 $t_{CLCL}$ will be filtered out. Maximum capacitance on bus-lines SDA and SCL = 400pF.
4. $t_{CLCL} = 1/f_{OSC}$ = one oscillator clock period at pin XTAL1. For 62ns < $t_{CLCL}$ < 285ns (16MHz > $f_{OSC}$ > 3.5MHz) the SIO1 interface meets the I²C-bus specification for bit-rates up to 100 kbit/s.

## TIMING SIO1 (I²C) INTERFACE




### Oscillator Circuitry

The capacitors connected to the crystal should be: C1 = C2 = 20pF.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{PSEN}$ low.

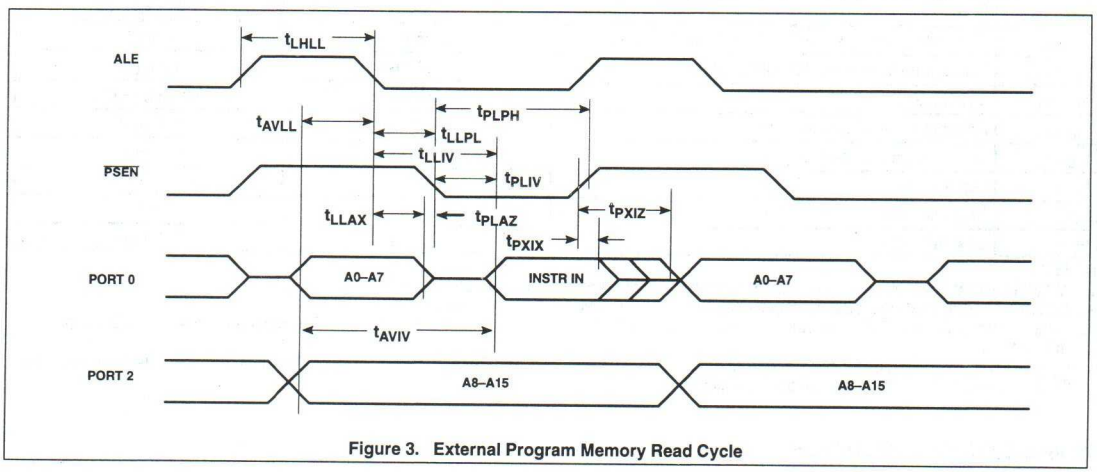



**Figure 3. External Program Memory Read Cycle**

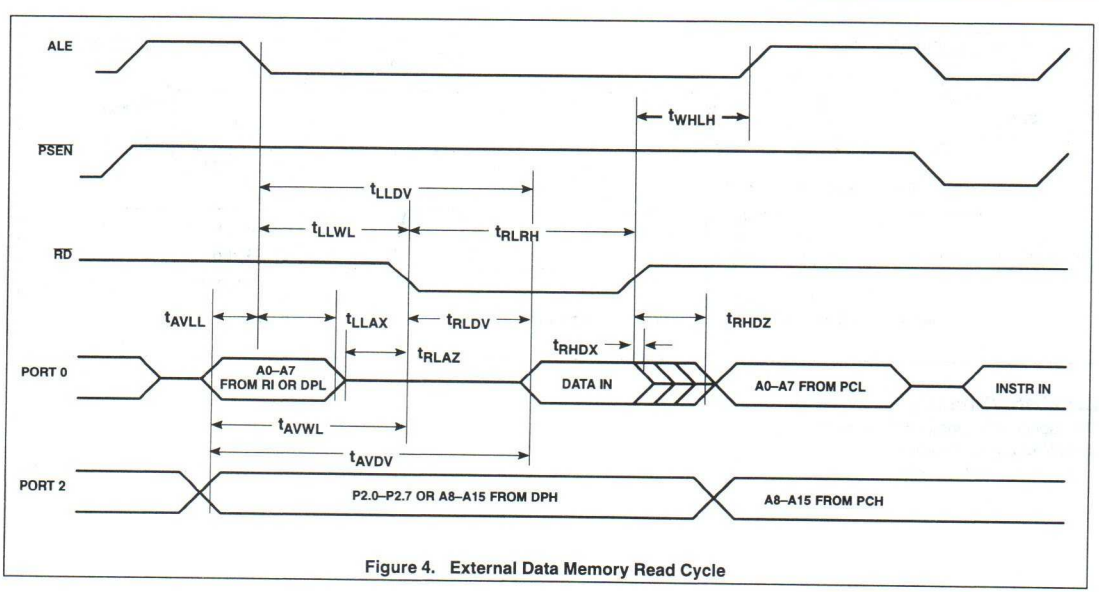



**Figure 4. External Data Memory Read Cycle**

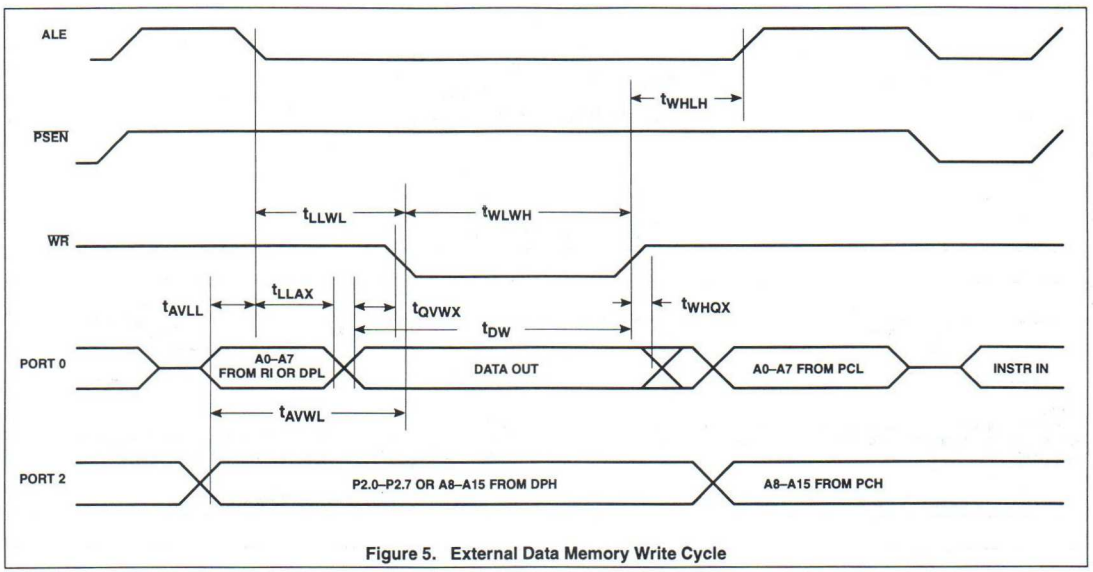



**Figure 5. External Data Memory Write Cycle**

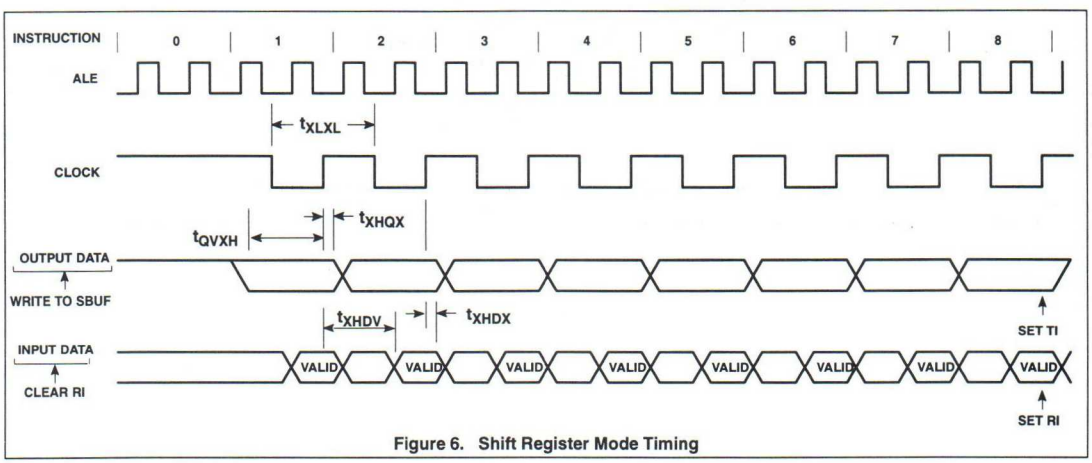



**Figure 6. Shift Register Mode Timing**

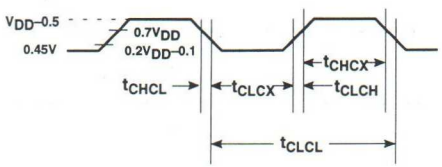
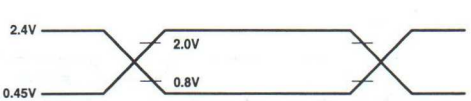
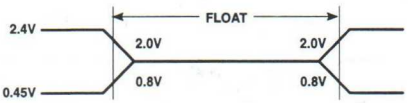



**Figure 7. External Clock Drive**




NOTE:
AC INPUTS DURING TESTING ARE DRIVEN AT 2.4V FOR A LOGIC '1' AND 0.45V FOR A LOGIC '0'. TIMING MEASUREMENTS ARE MADE AT 2.0V FOR A LOGIC '1' AND 0.8V FOR A LOGIC '0'.

**Figure 8. AC Testing Input/Output**




NOTE:
THE FLOAT STATE IS DEFINED AS THE POINT AT WHICH A PORT 0 PIN SINKS 3.2MA OR SOURCES 400µS AT THE VOLTAGE TEST LEVELS.

**Figure 9. Float Waveform**

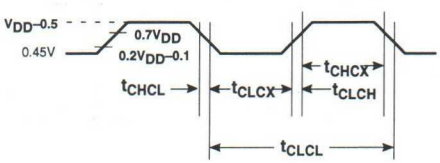



**Figure 10. Clock Signal Waveform for $I_{DD}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# CMOS single-chip 8-bit microcontrollers — 83C748/87C748

## DESCRIPTION

The Philips 83C748/87C748 offers the advantages of the 80C51 architecture in a small package and at low cost.

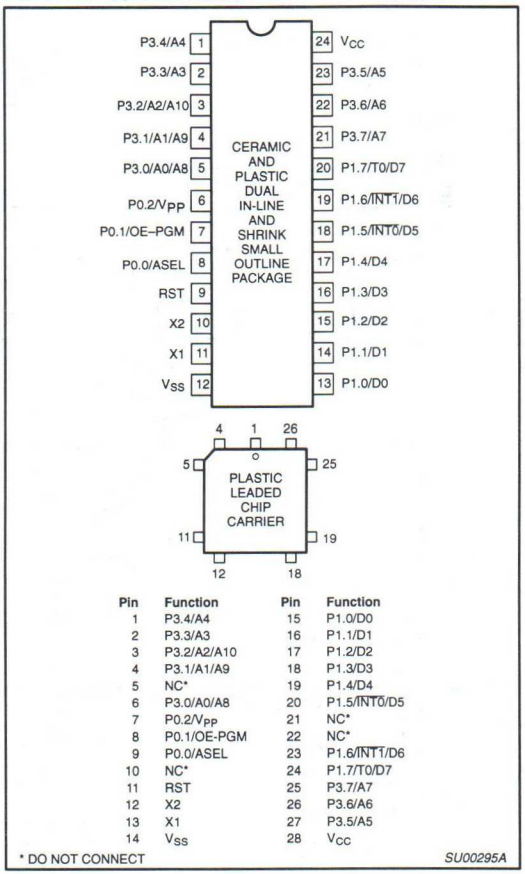
The 8XC748 Microcontroller is fabricated with Philips high-density CMOS technology. Philips epitaxial substrate minimizes CMOS latch-up sensitivity.

The 8XC748 contains a 2k × 8 ROM (83C748) EPROM (87C748), a 64 × 8 RAM, 19 I/O lines, a 16-bit auto-reload counter/timer, a four-source, fixed-priority level interrupt structure, and an on-chip oscillator.

## FEATURES

- 80C51 based architecture
- Small package sizes
  - 24-pin DIP (300 mil "skinny DIP")
  - 24-pin Shrink Small Outline Package (SSOP)
  - 28-pin PLCC
- 87C748 available in erasable quartz lid or one-time programmable plastic packages
- Wide oscillator frequency range: –3.5 to 16MHz
- Low power consumption:
  - Normal operation: less than 11mA @ 5V, 12MHz
  - Idle mode
  - Power-down mode
- 2k × 8 ROM (83C748)
  2k × 8 EPROM (87C748)
- 64 × 8 RAM
- 16-bit auto reloadable counter/timer
- 10-bit fixed-rate timer
- Boolean processor
- CMOS and TTL compatible
- Well suited for logic replacement, consumer and industrial applications
- LED drive outputs

## PIN CONFIGURATIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P3.4/A4 | 15 | P1.0/D0 |
| 2 | P3.3/A3 | 16 | P1.1/D1 |
| 3 | P3.2/A2/A10 | 17 | P1.2/D2 |
| 4 | P3.1/A1/A9 | 18 | P1.3/D3 |
| 5 | NC* | 19 | P1.4/D4 |
| 6 | P3.0/A0/A8 | 20 | P1.5/INT0/D5 |
| 7 | P0.2/VPP | 21 | NC* |
| 8 | P0.1/OE-PGM | 22 | NC* |
| 9 | P0.0/ASEL | 23 | P1.6/INT1/D6 |
| 10 | NC* | 24 | P1.7/T0/D7 |
| 11 | RST | 25 | P3.7/A7 |
| 12 | X2 | 26 | P3.6/A6 |
| 13 | X1 | 27 | P3.5/A5 |
| 14 | VSS | 28 | VCC |

* DO NOT CONNECT

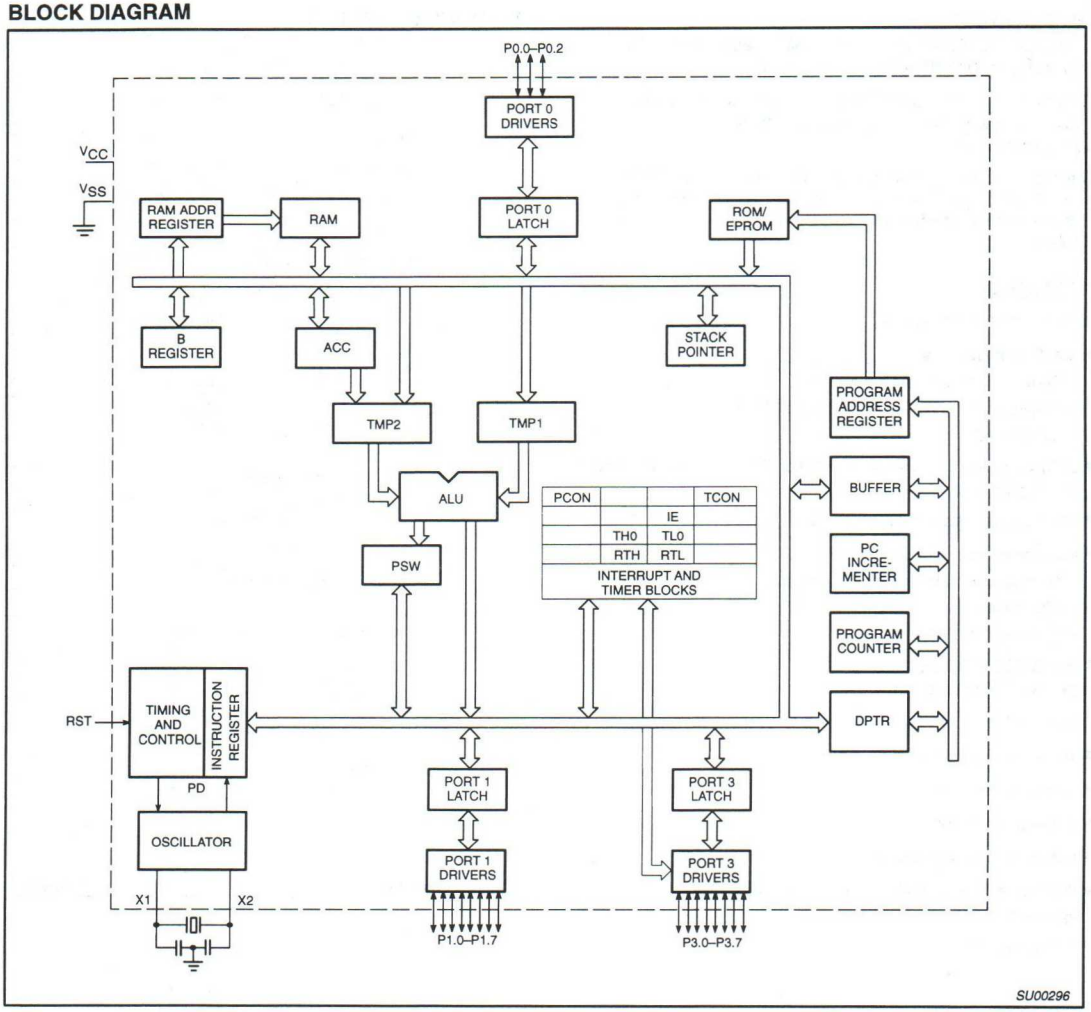
SU00295A

## ORDERING INFORMATION

| ROM | EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQUENCY MHz | DRAWING NUMBER |
|---|---|---|---|---|---|
| | P87C748EBF FA | UV | 0 to +70, Ceramic Dual In-line Package | 3.5 to 16 | 0586B |
| | P87C748EFF FA | UV | –40 to +85, Ceramic Dual In-line Package | 3.5 to 16 | 0586B |
| P83C748EBP N | P87C748EBP N | OTP | 0 to +70, Plastic Dual In-line Package | 3.5 to 16 | SOT222-1 |
| P83C748EFP N | P87C748EFP N | OTP | –40 to +85, Plastic Dual In-line Package | 3.5 to 16 | SOT222-1 |
| P83C748EBA A | P87C748EBA A | OTP | 0 to +70, Plastic Leaded Chip Carrier | 3.5 to 16 | SOT261-3 |
| P83C748EFA A | P87C748EFA A | OTP | –40 to +85, Plastic Leaded Chip Carrier | 3.5 to 16 | SOT261-3 |
| P83C748EBD DB | P87C748EBD DB | OTP | 0 to +70, Shrink Small Outline Package | 3.5 to 16 | SOT340-1 |

**NOTE:**
1. OTP = One Time Programmable EPROM. UV = UV Erasable EPROM.

## BLOCK DIAGRAM



## PIN DESCRIPTIONS

| MNEMONIC | PIN NO. DIP/ SSOP | PIN NO. LCC | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|
| $V_{SS}$ | 12 | 14 | I | **Circuit Ground Potential** |
| $V_{CC}$ | 24 | 28 | I | **Supply voltage during normal, idle, and power-down operation.** |
| P0.0–P0.2 | 8–6 | 9–7 | I/O | **Port 0:** Port 0 is a 3-bit open-drain, bidirectional port. Port 0 pins that have 1s written to them float, and in that state can be used as high-impedance inputs. These pins are driven low if the port register bit is written with a 0. The state of the pin can always be read from the port register by the program.<br>P0.0 and P0.1 are open drain bidirectional I/O pins. While these differ from "standard TTL" characteristics, they are close enough for the pins to still be used as general-purpose I/O. Port 0 also provides alternate functions for programming the EPROM memory as follows: |
| | 6 | 7 | N/A | **$V_{PP}$ (P0.2)** – Programming voltage input. |
| | 7 | 8 | I | **OE/PGM (P0.1)** – Input which specifies verify mode (output enable) or the program mode.<br>OE/PGM = 1 output enabled (verify mode).<br>OE/PGM = 0 program mode. |
| | 8 | 9 | I | **ASEL (P0.0)** – Input which indicates which bits of the EPROM address are applied to port 3.<br>ASEL = 0 low address byte available on port 3.<br>ASEL = 1 high address byte available on port 3 (only the three least significant bits are used). |
| P1.0–P1.7 | 13–20 | 15–20, 23, 24 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 serves to output the addressed EPROM contents in the verify mode and accepts as inputs the value to program into the selected address during the program mode. Port 1 also serves the special function features of the 80C51 family as listed below: |
| | 18 | 20 | I | **$\overline{INT0}$ (P1.5):** External interrupt. |
| | 19 | 23 | I | **$\overline{INT1}$ (P1.6):** External interrupt. |
| | 20 | 24 | I | **T0 (P1.7):** Timer 0 external input. |
| P3.0–P3.7 | 5–1, 23–21 | 6, 4–1, 27–25 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also functions as the address input for the EPROM memory location to be programmed (or verified). The 11-bit address is multiplexed into this port as specified by P0.0/ASEL. |
| RST | 9 | 11 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on RESET using only an external capacitor to $V_{CC}$. After the device is reset, a 10-bit serial sequence, sent LSB first, applied to RESET, places the device in the programming state allowing programming address, data and $V_{PP}$ to be applied for programming or verification purposes. The RESET serial sequence must be synchronized with the X1 input. |
| X1 | 11 | 13 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. X1 also serves as the clock to strobe in a serial bit stream into RESET to place the device in the programming state. |
| X2 | 10 | 12 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

## ABSOLUTE MAXIMUM RATINGS[1, 2]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage from $V_{CC}$ to $V_{SS}$ | –0.5 to +6.5 | V |
| Voltage from any pin to $V_{SS}$ (except $V_{PP}$) | –0.5 to $V_{CC}$ + 0.5 | V |
| Power dissipation | 1.0 | W |
| Voltage on $V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Maximum $I_{OL}$ per I/O pin | 10 | mA |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1]

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $V_{IL}$ | Input low voltage | | –0.5 | $0.2V_{DD}$–0.1 | V |
| $V_{IH}$ | Input high voltage, except X1, RST | | $0.2V_{CC}$+0.9 | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, X1, RST | | $0.7V_{CC}$ | $V_{CC}$+0.5 | V |
| | P0.2 | | | | |
| $V_{IL1}$ | Input low voltage | | –0.5 | $0.3V_{CC}$ | V |
| $V_{IH2}$ | Input high voltage | | $0.7V_{CC}$ | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1 and 3 | $I_{OL}$ = 1.6mA[2] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0.2 | $I_{OL}$ = 3.2mA[2] | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1 and 3 | $I_{OH}$ = –60µA | 2.4 | | V |
| | | $I_{OH}$ = –25µA | $0.75V_{CC}$ | | V |
| | | $I_{OH}$ = –10µA | $0.9V_{CC}$ | | V |
| | Port 0.0 and 0.1 – Drivers | | | | |
| $V_{OL2}$ | Output low voltage | $I_{OL}$ = 3mA | | 0.4 | V |
| | Driver, receiver combined: | (over $V_{CC}$ range) | | | |
| C | Capacitance | | | 10 | pF |
| $I_{IL}$ | Logical 0 input current, ports 1 and 3 | $V_{IN}$ = 0.45V | | –50 | µA |
| $I_{TL}$ | Logical 1 to 0 transition current, ports 1 and 3[3] | $V_{IN}$ = 2V (0 to 70°C) | | –650 | µA |
| | | $V_{IN}$ = 2V (–40 to +85°C) | | –750 | µA |
| $I_{LI}$ | Input leakage current, port 0 | 0.45 < $V_{IN}$ < $V_{CC}$ | | ±10 | µA |
| $R_{RST}$ | Internal pull-down resistor | | 25 | 175 | kΩ |
| $C_{IO}$ | Pin capacitance | Test freq = 1MHz, $T_{amb}$ = 25°C | | 10 | pF |
| $I_{PD}$ | Power-down current[4] | $V_{CC}$ = 2 to $V_{CC}$ max | | 50 | µA |
| $V_{PP}$ | $V_{PP}$ program voltage (for 87C748 only) | $V_{SS}$ = 0V<br>$V_{CC}$ = 5V±10%<br>$T_{amb}$ = 21°C to 27°C | 12.5 | 13.0 | V |
| $I_{PP}$ | Program current (for 87C748 only) | $V_{PP}$ = 13.0V | | 50 | mA |
| $I_{CC}$ | Supply current (see Figure 2) | | | | |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.
2. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 10mA (NOTE: This is 85°C spec.)
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 67mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
3. Pins of ports 1 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
4. Power-down $I_{CC}$ is measured with all output pins disconnected; port 0 = $V_{CC}$; X2, X1 n.c.; RST = $V_{SS}$.
5. Active $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; RST = port 0 = $V_{CC}$. $I_{CC}$ will be slightly higher if a crystal oscillator is used.
6. Idle $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; port 0 = $V_{CC}$; RST = $V_{SS}$.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2]

| SYMBOL | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | Oscillator frequency: | | | 3.5<br>3.5 | 12<br>16 | MHz<br>MHz |
| **External Clock (Figure 1)** | | | | | | |
| $t_{CHCX}$ | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | Fall time | | 20 | | 20 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.
2. Load capacitance for ports = 80pF.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

C – Clock
D – Input data
H – Logic level high
L – Logic level low
Q – Output data
T – Time
V – Valid
X – No longer a valid logic level
Z – Float

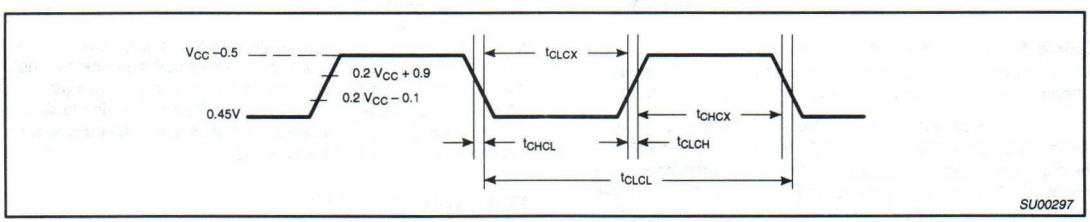



**Figure 1. External Clock Drive**

## ROM CODE SUBMISSION

When submitting ROM code for the 83C748, the following must be specified:
1. 2k byte user ROM data

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 07FFH | DATA | 7:0 | User ROM Data |

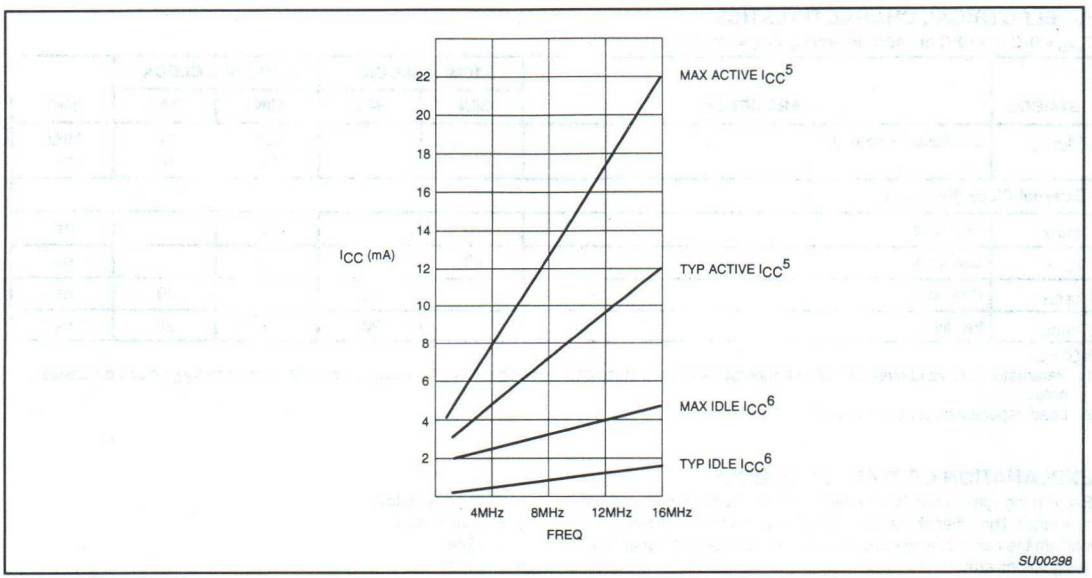



**Figure 2. $I_{CC}$ vs. FREQ**
**Maximum $I_{CC}$ values taken at $V_{CC}$ max and worst case temperature.**
**Typical $I_{CC}$ values taken at $V_{CC}$ = 5.0V and 25°C.**
**Notes 5 and 6 refer to DC Electrical Characteristics.**

## OSCILLATOR CHARACTERISTICS

X1 and X2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, X1 should be driven while X2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-up reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-up, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up.

## IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. the control bits for the reduced power modes are in the special function register PCON.

**Table 1. External Pin Status During Idle and Power-Down Modes**

| MODE | Port 0 | Port 1 | Port 2 |
|---|---|---|---|
| Idle | Data | Data | Data |
| Power-down | Data | Data | Data |

## DIFFERENCES BETWEEN THE 8XC748 AND THE 80C51

### Memory Organization

The central processing unit (CPU) manipulates operands in two address spaces as shown in Figure 3. The part's internal memory space consists of 2k bytes of program memory, and 64 bytes of data RAM overlapped with the 128-byte special function register area. The differences from the 80C51 are in RAM size (64 bytes vs. 128 bytes), in external RAM access (not available on the 83C748), in internal ROM size (2k bytes vs. 4k bytes), and in external program memory expansion (not available on the 83C748). The 128-byte special function register (SFR) space is accessed as on the 80C51 with some of the registers having been changed to reflect changes in the 83C748 peripheral functions. The stack may be located anywhere in internal RAM by loading the 8-bit stack pointer (SP). It should be noted that stack depth is limited to 64 bytes, the amount of available RAM. A reset loads the stack pointer with 07 (which is pre-incremented on a PUSH instruction).

### Program Memory

On the 8XC748, program memory is 2048 bytes long and is not externally expandable, so the 80C51 instructions MOVX, LJMP, and LCALL are not implemented. The only fixed locations in program memory are the addresses at which execution is taken up in response to reset and interrupts, which are as follows:

| Event | Program Memory Address |
|---|---|
| Reset | 000 |
| External $\overline{INT0}$ | 003 |
| Counter/timer 0 | 00B |
| External $\overline{INT1}$ | 013 |
| Timer I | 01B |

### Counter/Timer Subsystem

The 8XC748 has one counter/timer called timer/counter 0. Its operation is similar to mode 2 operation on the 80C51, but is extended to 16 bits with 16 bits of autoload. The controls for this counter are centralized in a single register called TCON.

Timer I is available for use as a fixed 10-bit time-base, or as a watchdog.

### Counter Timer – Special Function Register

The counter/timer has only one mode of operation, so the TMOD SFR is not used. There is also only one counter/timer, so there is no need for the TL1 and TH1 SFRs found on the 80C51. These have been replaced on the 8XC748 by RTL and RTH, the counter/timer reload registers. Table 2 shows the special function registers, their locations, and reset values.

### Interrupt Subsystem – Fixed Priority

The IP register and the 2-level interrupt system of the 80C51 are eliminated. Simultaneous interrupt conditions are resolved by a single-level, fixed priority as follows:

| | |
|---|---|
| Highest priority: | Pin $\overline{INT0}$ |
| | Counter/timer flag 0 |
| | Pin $\overline{INT1}$ |
| Lowest priority: | Timer I |

### Special Function Register – Interrupt Subsystem

Because the interrupt structure is single level on the 83C748, there is no need for the IP SFR, so it is not used.

### Special Function Register – Serial Communications

The 8XC748 contains many of the special function registers (SFR) that are found on the 80C51. Due to the different peripheral features on the 8XC748, there are several additional SFRs. Since the UART found on 80C51 has been removed, the UART SFRs SCON and SBUF have also been removed.

### I/O Port Latches (P0, P1, P3)

The port latches function the same as those on the 80C51. Since there is no port 2 on the 83C748, the P2 latch is not used. Port 0 on the 83C748 has only 3 bits, so only 3 bits of the P0 SFR have a useful function.

### Data Pointer (DPTR)

The data pointer (DPTR) consists of a high byte (DPH) and a low byte (DPL). In the 80C51 this register allows the access of external data memory using the MOVX instruction. Since the 83C748 does not support MOVX or external memory accesses, this register is generally used as a 16-bit offset pointer of the accumulator in a MOVC instruction. DPTR may also be manipulated as two independent 8-bit registers.

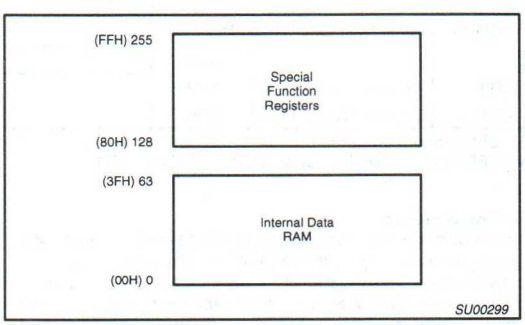



**Figure 3. Memory Map**

**Table 2. 8XC748 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes) | | | | | | | | | | |
| DPH | High byte | 83H | | | | | | | | | 00H |
| DPL | Low byte | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | – | – | – | ETI | EX1 | ET0 | EX0 | 00H |
| | | | | | | | | 82 | 81 | 80 | |
| P0*# | Port 0 | 80H | – | – | – | – | – | – | – | – | xxxxx111B |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | T0 | $\overline{INT1}$ | $\overline{INT0}$ | – | – | – | – | – | FFH |
| P3* | Port 3 | B0H | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | FFH |
| PCON# | Power control | 87H | – | – | – | – | – | – | PD | IDL | xxxxxx00B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON*# | Timer/counter control | 88H | GATE | C/T | TF | TR | IE0 | IT0 | IE1 | IT1 | 00H |
| TL# | Timer low byte | 8AH | | | | | | | | | 00H |
| TH# | Timer high byte | 8CH | | | | | | | | | 00H |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| TICON*# | Timer I control | D8H/RD | – | – | 0 | TIRUN | – | – | – | – | 0000xx00B |
| | | WR | – | – | CLRTI | TIRUN | – | – | – | – | |
| RTL# | Timer low reload | 8BH | | | | | | | | | 00H |
| RTH# | Timer high reload | 8DH | | | | | | | | | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

**I/O Port Structure**

The 8XC748 has two 8-bit ports (ports 1 and 3) and one 3-bit port (port 0). All three ports on the 8XC748 are bidirectional. Each consists of a latch (special function register P0, P1, P3), an output driver, and an input buffer. Three port 1 pins and two port 0 pins are multifunctional. In addition to being port pins, these pins serve the function of special features as follows:

| Port Pin | Alternate Function |
|---|---|
| P1.5 | INT0 (external interrupt 0 input) |
| P1.6 | INT1 (external interrupt 1 input) |
| P1.7 | T0 (timer 0 external input) |

Ports 1 and 3 are identical in structure to the same ports on the 80C51. The structure of port 0 on the 8XC748 is similar to that of the 80C51 but does not include address/data input and output circuitry. As on the 80C51, ports 1 and 3 are quasi-bidirectional while port 0 is bidirectional with no internal pullups.

**Timer/Counter**

The 8XC748 has two timers: a 16-bit timer/counter and a 10-bit fixed-rate timer. The 16-bit timer/counter's operation is similar to mode 2 operation on the 80C51, but is extended to 16 bits. The timer/counter is clocked by either 1/12 the oscillator frequency or by transitions on the T0 pin. The C/T pin in special function register TCON selects between these two modes. When the TCON TR bit is set, the timer/counter is enabled. Register pair TH and TL are incremented by the clock source. When the register pair overflows, the register pair is reloaded with the values in registers RTH and RTL. The value in the reload registers is left unchanged. See the 83C748 counter/timer block diagram in Figure 4. The TF bit in special function register TCON is set on counter overflow and, if the interrupt is enabled, will generate an interrupt.

### TCON Register

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| GATE | C/T | TF | TR | IE0 | IT0 | IE1 | IT1 |

GATE 1 – Timer/counter is enabled only when INT0 pin is high, and TR is 1.
0 – Timer/counter is enabled when TR is 1.
C/T 1 – Counter/timer operation from T0 pin.
0 – Timer operation from internal clock.
TF 1 – Set on overflow of TH.
0 – Cleared when processor vectors to interrupt routine and by reset.
TR 1 – Timer/counter enabled.
0 – Timer/counter disabled.
IE0 1 – Edge detected in $\overline{INT0}$.
IT0 1 – $\overline{INT0}$ is edge triggered.
0 – $\overline{INT0}$ is level sensitive.
IE1 1 – Edge detected on $\overline{INT1}$.
IT1 1 – $\overline{INT1}$ is edge triggered.
0 – $\overline{INT1}$ is level sensitive.

These flags are functionally identical to the corresponding 80C51 flags, except that there is only one timer on the 83C748 and the flags are therefore combined into one register.

Note that the positions of the IE0/IT0 and IE1/IT1 bits are transposed from the positions used in the standard 80C51 TCON register.

### Timer I Implementation

Timer I is clocked once per machine cycle, which is the oscillator frequency divided by 12. The timer operation is enabled by setting the TIRUN bit (bit 4) in the I2CFG register. Writing a 0 into the TIRUN bit will stop and clear the timer. The timer is 10 bits wide, and when it reaches the terminal count of 1024, it carries out and sets the Timer I interrupt flag. An interrupt will occur if the Timer I interrupt is enabled by bit ETI (bit 4) of the Interrupt Enable (IE) register, and global interrupts are enabled by bit EA (bit 7) of the same IE register.

The vector address for the Timer I interrupt is 1Bhex, and the interrupt service routine must start at this address. As with all 8051 family microcontrollers, only the Program Counter is pushed onto the stack upon interrupt (other registers that are used both by the interrupt service routine and elsewhere must be explicitly saved). The Timer I interrupt flag is cleared by setting the CKRTI bit (bit 5 of the I1CFG register. For more information, see application note AN427.

### Interrupts

The interrupt structure is a four-source, one-level interrupt system. Interrupt sources common to the 80C51 are the external interrupts ($\overline{INT0}$, $\overline{INT1}$) and the timer/counter interrupt (ET0). Timer I interrupt (ETI) is the other interrupt source. The interrupt sources are listed below in their order of polling sequence priority.

Upon interrupt or reset the program counter is loaded with specific values for the appropriate interrupt service routine in program memory. These values are:

| Event | Program Memory Address | Priority |
|---|---|---|
| Reset | 000 | Highest |
| INT0 | 003 | |
| Counter/Timer 0 | 00B | |
| INT1 | 013 | |
| Timer I | 01B | Lowest |

The interrupt enable register (IE) is used to individually enable or disable the four sources. Bit $\overline{EA}$ in the interrupt enable register can be used to globally enable or disable all interrupt sources. The interrupt enable register is described below. All other interrupt details are based on the 80C51 interrupt architecture.

### Interrupt Enable Register

| $\overline{EA}$ | X | X | – | ETI | EX1 | ET0 | EX0 |
|---|---|---|---|---|---|---|---|

| Symbol | Position | Function |
|---|---|---|
| $\overline{EA}$ | IE.7 | Disables all interrupts. If EA = 0, no interrupt will be acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit |
| – | IE.6 | Reserved |
| – | IE.5 | Reserved |
| – | IE.4 | Reserved |
| ETI | IE.3 | Enables or disables the Timer I overflow interrupt. If ET1 = 0, the Timer I interrupt is disabled. |
| EX1 | IE.2 | Enables or disables external interrupt 1. If EX1 = 0, external interrupt 1 is disabled. |
| ET0 | IE.1 | Enables or disables the Timer 0 overflow interrupt. If ET0 = 0, theTimer 0 interrupt is disabled. |
| EX0 | IE.0 | Enables or disables external interrupt 0. If EX0 = 0, external interrupt 0 is disabled. |

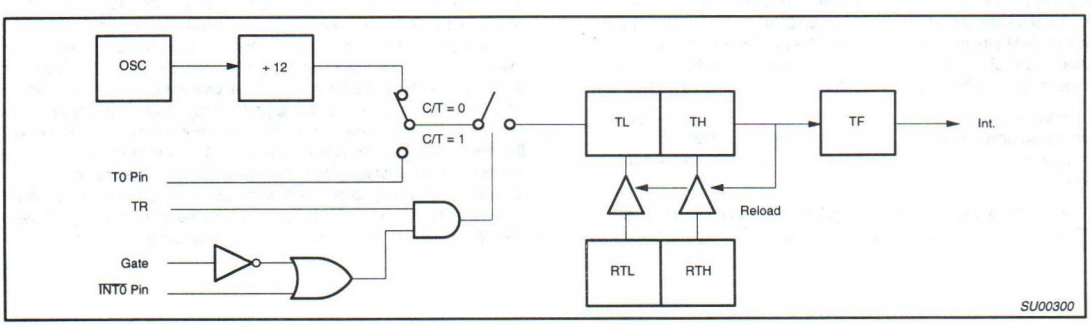



**Figure 4. 83C748 Counter/Timer Block Diagram**

## 87C748 PROGRAMMING CONSIDERATIONS

### EPROM Characteristics

The 87C748 is programmed by using a modified Quick-Pulse Programming algorithm similar to that used for devices such as the 87C451 and 87C51. It differs from these devices in that a serial data stream is used to place the 87C748 in the programming mode.

Figure 5 shows a block diagram of the programming configuration for the 87C748. Port pin P0.2 is used as the programming voltage supply input ($V_{PP}$ signal). Port pin P0.1 is used as the program (PGM/) signal. This pin is used for the 25 programming pulses.

Port 3 is used as the address input for the byte to be programmed and accepts both the high and low components of the eleven bit address. Multiplexing of these address components is performed using the ASEL input. The user should drive the ASEL input high and then drive port 3 with the high order bits of the address. ASEL should remain high for at least 13 clock cycles. ASEL may then be driven low which latches the high order bits of the address internally. the high address should remain on port 3 for at least two clock cycles after ASEL is driven low. Port 3 may then be driven with the low byte of the address. The low address will be internally stable 13 clock cycles later. The address will remain stable provided that the low byte placed on port 3 is held stable and ASEL is kept low. **Note:** ASEL needs to be pulsed high only to change the high byte of the address.

Port 1 is used as a bidirectional data bus during programming and verify operations. During programming mode, it accepts the byte to be programmed. During verify mode, it provides the contents of the EPROM location specified by the address which has been supplied to Port 3.

The XTAL1 pin is the oscillator input and receives the master system clock. This clock should be between 1.2 and 6MHz.

The RESET pin is used to accept the serial data stream that places the 87C748 into various programming modes. This pattern consists of a 10-bit code with the LSB sent first. Each bit is synchronized to the clock input, X1.

### Programming Operation

Figures 6 and 7 show the timing diagrams for the program/verify cycle. RESET should initially be held high for at least two machine cycles. P0.1 (PGM/) and P0.2 ($V_{PP}$) will be at $V_{OH}$ as a result of the RESET operation. At this point, these pins function as normal quasi-bidirectional I/O ports and the programming equipment may pull these lines low. However, prior to sending the 10-bit code on the RESET pin, the programming equipment should drive these pins high ($V_{IH}$). The RESET pin may now be used as the serial data input for the data stream which places the 87C748 in the programming mode. Data bits are sampled during the clock high time and thus should only change during the time that the clock is low. Following transmission of the last data bit, the RESET pin should be held low.

Next the address information for the location to be programmed is placed on port 3 and ASEL is used to perform the address multiplexing, as previously described. At this time, port 1 functions as an output.

A high voltage $V_{PP}$ level is then applied to the $V_{PP}$ input (P0.2). (This sets Port 1 as an input port). The data to be programmed into the EPROM array is then placed on Port 1. This is followed by a series of programming pulses applied to the PGM/ pin (P0.1). These pulses are created by driving P0.1 low and then high. This pulse is repeated until a total of 25 programming pulses have occurred. At the conclusion of the last pulse, the PGM/ signal should remain high.

The $V_{PP}$ signal may now be driven to the $V_{OH}$ level, placing the 87C748 in the verify mode. (Port 1 is now used as an output port). After four machine cycles (48 clock periods), the contents of the addressed location in the EPROM array will appear on Port 1.

The next programming cycle may now be initiated by placing the address information at the inputs of the multiplexed buffers, driving the $V_{PP}$ pin to the $V_{PP}$ voltage level, providing the byte to be programmed to Port1 and issuing the 26 programming pulses on the PGM/ pin, bringing $V_{PP}$ back down to the $V_C$ level and verifying the byte.

### Programming Modes

The 87C748 has four programming features incorporated within its EPROM array. These include the USER EPROM for storage of the application's code, a 16-byte encryption key array and two security bits. Programming and verification of these four elements are selected by a combination of the serial data stream applied to the RESET pin and the voltage levels applied to port pins P0.1 and P0.2. The various combinations are shown in Table 3.

**Table 3. Implementing Program/Verify Modes**

| OPERATION | SERIAL CODE | P0.1 (PGM/) | P0.2 ($V_{PP}$) |
|---|---|---|---|
| Program user EPROM | 296H | –* | $V_{PP}$ |
| Verify user EPROM | 296H | $V_{IH}$ | $V_{IH}$ |
| Program key EPROM | 292H | –* | $V_{PP}$ |
| Verify key EPROM | 292H | $V_{IH}$ | $V_{IH}$ |
| Program security bit 1 | 29AH | –* | $V_{PP}$ |
| Program security bit 2 | 298H | –* | $V_{PP}$ |
| Verify security bits | 29AH | $V_{IH}$ | $V_{IH}$ |

**NOTE:**
* Pulsed from $V_{IH}$ to $V_{IL}$ and returned to $V_{IH}$.

### Encryption Key Table

The 87C748 includes a 16-byte EPROM array that is programmable by the end user. The contents of this array can then be used to encrypt the program memory contents during a program memory verify operation. When a program memory verify operation is performed, the contents of the program memory location is XNOR'ed with one of the bytes in the 16-byte encryption table. The resulting data pattern is then provided to port 1 as the verify data. The encryption mechanism can be disable, in essence, by leaving the bytes in the encryption table in their erased state (FFH) since the XNOR product of a bit with a logical one will result in the original bit. The encryption bytes are mapped with the code memory in 16-byte groups. the first byte in code memory will be encrypted with the first byte in the encryption table; the second byte in code memory will be encrypted with the second byte in the encryption table and so forth up to and including the 16the byte. The encryption repeats in 16-byte groups; the 17th byte in the code memory will be encrypted with the first byte in the encryption table, and so forth.

## Security Bits

Two security bits, security bit 1 and security bit 2, are provided to limit access to the USER EPROM and encryption key arrays. Security bit 1 is the program inhibit bit, and once programmed performs the following functions:

1. Additional programming of the USER EPROM is inhibited.
2. Additional programming of the encryption key is inhibited.
3. Verification of the encryption key is inhibited.
4. Verification of the USER EPROM and the security bit levels may still be performed.

(If the encryption key array is being used, this security bit should be programmed by the user to prevent unauthorized parties from reprogramming the encryption key to all logical zero bits. Such programming would provide data during a verify cycle that is the logical complement of the USER EPROM contents).

Security bit 2, the verify inhibit bit, prevents verification of both the USER EPROM array and the encryption key arrays. The security bit levels may still be verified.

## Programming and Verifying Security Bits

Security bits are programmed employing the same techniques used to program the USER EPROM and KEY arrays using serial data streams and logic levels on port pins indicated in Table 3. When programming either security bit, it is not necessary to provide address or data information to the 87C748 on ports 1 and 3.

Verification occurs in a similar manner using the RESET serial stream shown in Table 3. Port 3 is not required to be driven and the results of the verify operation will appear on ports 1.6 and 1.7.

Ports 1.7 contains the security bit 1 data and is a logical one if programmed and a logical zero if erased. Likewise, P1.6 contains the security bit 2 data and is a logical one if programmed and a logical zero if erased.

## Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Flourless part number 2345–5 or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-s/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

## EPROM PROGRAMMING AND VERIFICATION

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $1/t_{CLCL}$ | Oscillator/clock frequency | 1.2 | 6 | MHz |
| $t_{AVGL}$[1] | Address setup to P0.1 (PROG–) low | 10µs + 24$t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after P0.1 (PROG–) high | 48$t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to P0.1 (PROG–) low | 38$t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after P0.1 (PROG–) high | 36$t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to P0.1 (PROG–) low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after P0.1 (PROG–) | 10 | | µs |
| $t_{GLGH}$ | P0.1 (PROG–) width | 90 | 110 | µs |
| $t_{AVQV}$[2] | $V_{PP}$ low ($V_{CC}$) to data valid | | 48$t_{CLCL}$ | |
| $t_{GHGL}$ | P0.1 (PROG–) high to P0.1 (PROG–) low | 10 | | µs |
| $t_{MASEL}$ | ASEL high time | 13$t_{CLCL}$ | | |
| $t_{HAHLD}$ | Address hold time | 2$t_{CLCL}$ | | |
| $t_{HASET}$ | Address setup to ASEL | 13$t_{CLCL}$ | | |
| $t_{ADSTA}$ | Low address to valid data | | 48$t_{CLCL}$ | |

**NOTES:**
1. Address should be valid at least 24$t_{CLCL}$ before the rising edge of P0.2 ($V_{PP}$).
2. For a pure verify mode, i.e., no program mode in between, $t_{AVQV}$ is 14$t_{CLCL}$ maximum.

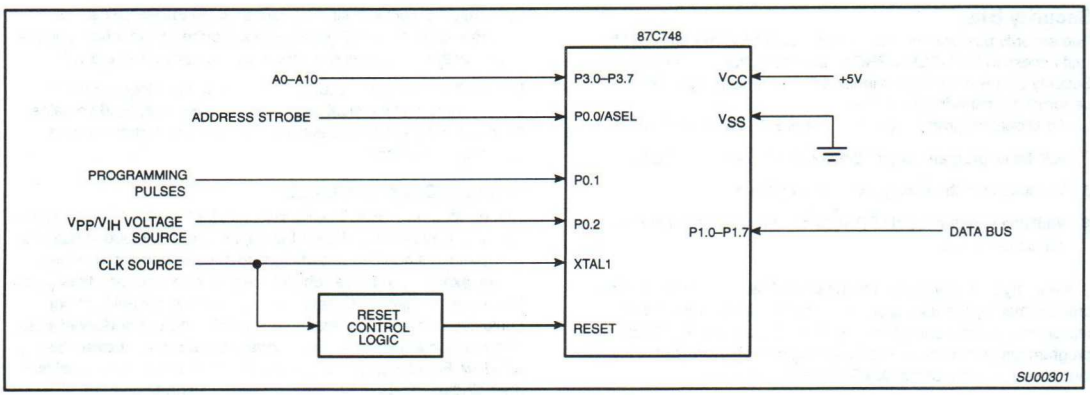



**Figure 5. Programming Configuration**

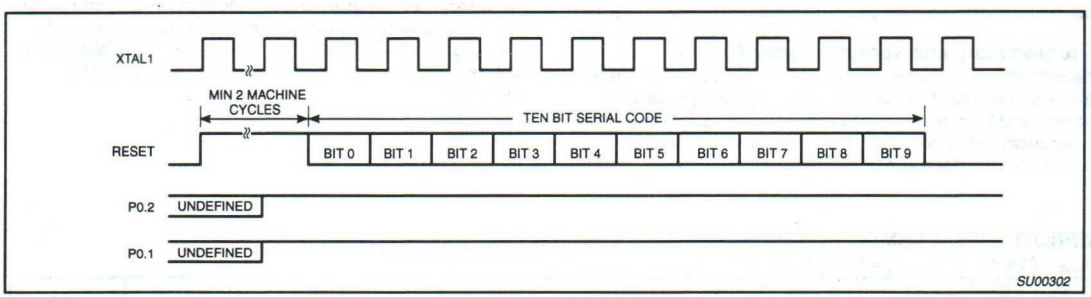



**Figure 6. Entry into Program/Verify Modes**

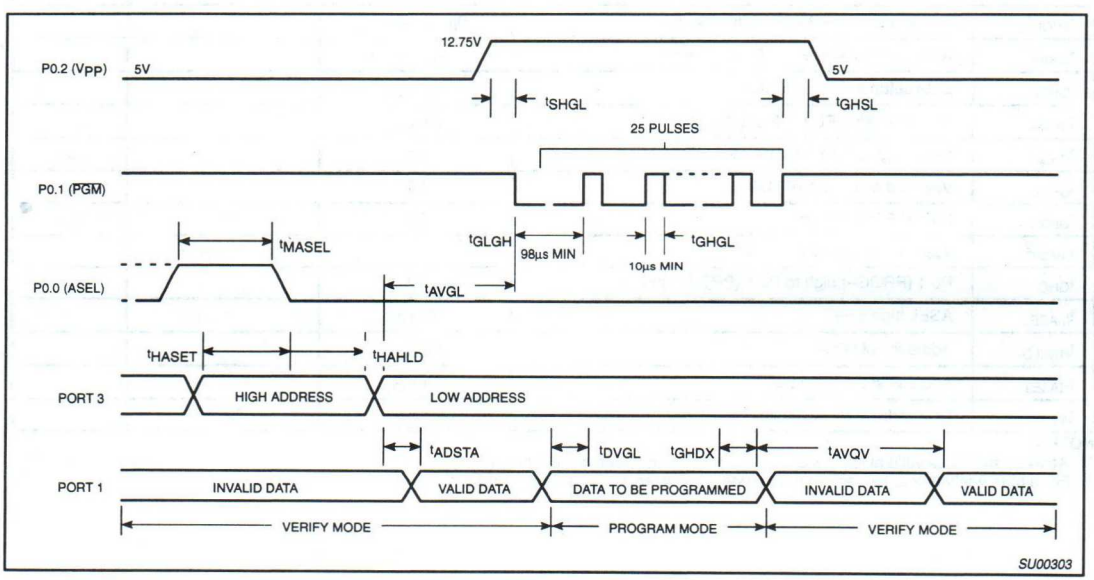



**Figure 7. Program/Verify Cycle**

# CMOS single-chip 8-bit microcontrollers 83C749/87C749

## DESCRIPTION

The Philips 83C749/87C749 offers many of the advantages of the 80C51 architecture in a small package and at low cost.

The 8XC749 Microcontroller is fabricated with Philips high-density CMOS technology. Philips epitaxial substrate minimizes CMOS latch-up sensitivity.

The 8XC749 contains a 2k × 8 ROM (83C749) EPROM (87C749), a 64 × 8 RAM, 21 I/O lines, a 16-bit auto-reload counter/timer, a fixed-priority level interrupt structure, an on-chip oscillator, a five channel multiplexed 8-bit A/D converter, and an 8-bit PWM output.

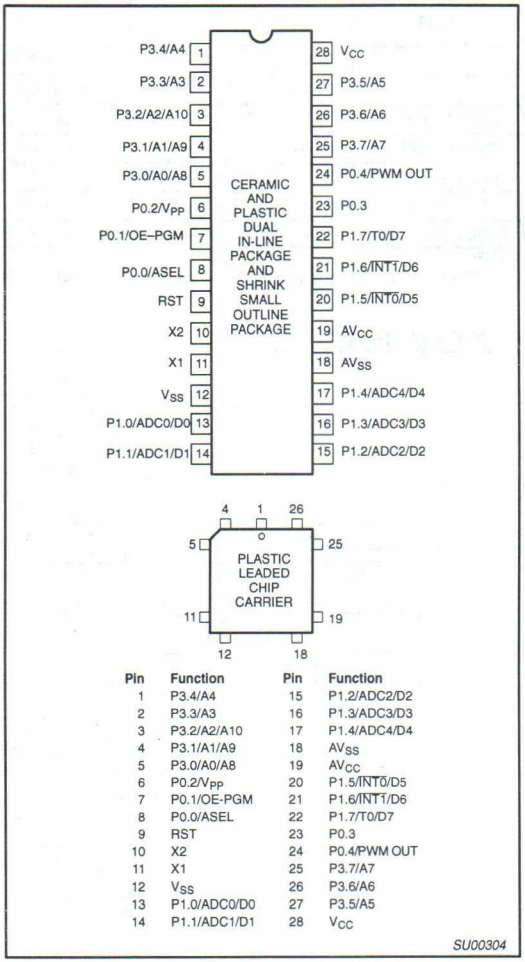
The EPROM version of this device, the 87C749, is also available in both quartz-lid erasable and plastic one-time programmable (OTP) packages. Once the array has been programmed, it is functionally equivalent to the masked ROM 83C749. Thus, unless explicitly stated otherwise, all references made to the 83C749 apply equally to the 87C749.

The 83C749 supports two power reduction modes of operation referred to as the idle mode and the power-down mode.

## FEATURES

- Available in erasable quartz lid or One-Time Programmable plastic packages
- 80C51 based architecture
- Small package sizes
  - 28-pin DIP
  - 28-pin Shrink Small Outline Package (SSOP)
  - 28-pin PLCC
- Wide oscillator frequency range: 3.5MHz to 16MHz
- Low power consumption:
  - Normal operation: less than 11mA @ 5V, 12MHz
  - Idle mode
  - Power-down mode
- 2k × 8 ROM (83C749)
  EPROM (87C749)
- 64 × 8 RAM
- 16-bit auto reloadable counter/timer
- 5-channel 8-bit A/D converter
- 8-bit PWM output/timer
- 10-bit fixed-rate timer
- Boolean processor
- CMOS and TTL compatible
- Well suited for logic replacement, consumer and industrial applications

## PIN CONFIGURATION




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P3.4/A4 | 15 | P1.2/ADC2/D2 |
| 2 | P3.3/A3 | 16 | P1.3/ADC3/D3 |
| 3 | P3.2/A2/A10 | 17 | P1.4/ADC4/D4 |
| 4 | P3.1/A1/A9 | 18 | $AV_{SS}$ |
| 5 | P3.0/A0/A8 | 19 | $AV_{CC}$ |
| 6 | $P0.2/V_{PP}$ | 20 | P1.5/$\overline{INT0}$/D5 |
| 7 | P0.1/OE-PGM | 21 | P1.6/$\overline{INT1}$/D6 |
| 8 | P0.0/ASEL | 22 | P1.7/T0/D7 |
| 9 | RST | 23 | P0.3 |
| 10 | X2 | 24 | P0.4/PWM OUT |
| 11 | X1 | 25 | P3.7/A7 |
| 12 | $V_{SS}$ | 26 | P3.6/A6 |
| 13 | P1.0/ADC0/D0 | 27 | P3.5/A5 |
| 14 | P1.1/ADC1/D1 | 28 | $V_{CC}$ |

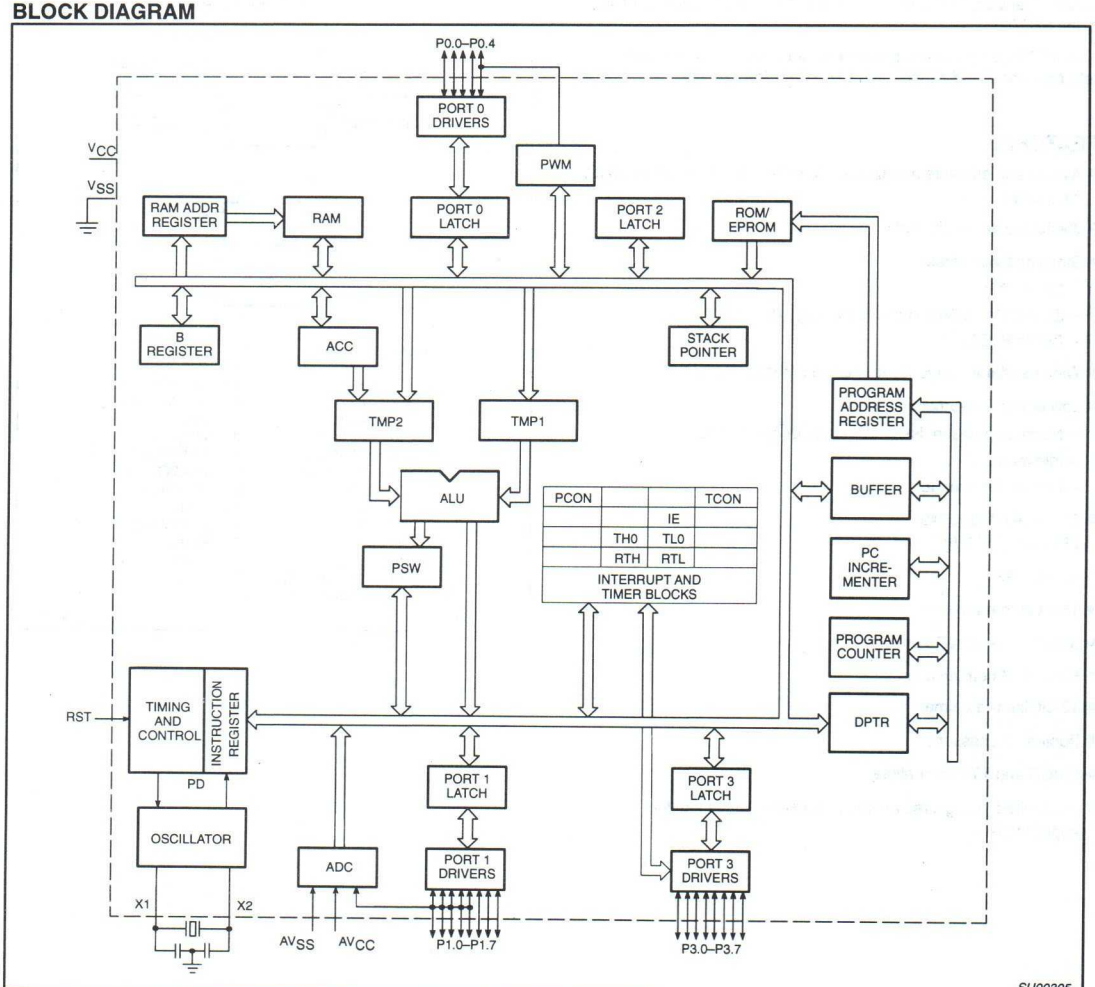
SU00304

## PART NUMBER SELECTION

| ROM | EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQUENCY | DRAWING NUMBER |
|---|---|---|---|---|---|
| | P87C749EBF FA | UV | 0 to +70, 28-pin Ceramic Dual In-line Package | 3.5 to 16MHz | 0589B |
| | P87C749EFF FA | UV | –40 to +85, 28-pin Ceramic Dual In-line Package | 3.5 to 16MHz | 0589B |
| P83C749EBP N | P87C749EBP N | OTP | 0 to +70, 28-pin Plastic Dual In-line Package | 3.5 to 16MHz | SOT117-2 |
| P83C749EFP N | P87C749EFP N | OTP | –40 to +85, 28-pin Plastic Dual In-line Package | 3.5 to 16MHz | SOT117-2 |
| P83C749EBA A | P87C749EBA A | OTP | 0 to +70, 28-pin Plastic Leaded Chip Carrier | 3.5 to 16MHz | SOT216-3 |
| P83C749EFA A | P87C749EFA A | OTP | –40 to +85, 28-pin Plastic Leaded Chip Carrier | 3.5 to 16MHz | SOT216-3 |
| P83C749EBD DB | P87C749EBD DB | OTP | 0 to +70, 28-pin Shrink Small Outline Package | 3.5 to 16MHz | SOT341-1 |

**NOTE:**
1. OTP = One Time Programmable EPROM. UV = UV Erasable EPROM.

## BLOCK DIAGRAM



## PIN DESCRIPTION

| MNEMONIC | PIN NO. | TYPE | NAME AND FUNCTION |
|---|---|---|---|
| $V_{SS}$ | 12 | I | **Circuit Ground Potential.** |
| $V_{CC}$ | 28 | I | **Supply voltage during normal, idle, and power-down operation.** |
| P0.0–P0.4 | 8–6<br>23, 24 | I/O | **Port 0:** Port 0 is a 5-bit bidirectional port. Port 0.0–P0.2 are open drain. Port 0.0–P0.2 pins that have 1s written to them float, and in that state can be used as high-impedance inputs. P0.3–P0.4 are bidirectional I/O port pins with internal pull-ups. These pins are driven low if the port register bit is written with a 0. The state of the pin can always be read from the port register by the program. Port 0.3 and 0.4 have internal pull-ups that function identically to port 3. Pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs.<br><br>While P0.0 anbd P0.1 differ from "standard TTL" characteristics, they are close enough for the pins to still be used as general-purpose I/O. |
| | 6 | I | **$V_{PP}$ (P0.2)** – Programming voltage input. (See Note 2.) |
| | 7 | I | **OE/PGM (P0.1)** – Input which specifies verify mode (output enable) or the program mode.<br>OE/PGM = 1 output enabled (verify mode).<br>OE/PGM = 0 program mode. |
| | 8 | I | **ASEL (P0.0)** – Input which indicates which bits of the EPROM address are applied to port 3.<br>ASEL = 0 low address byte available on port 3.<br>ASEL = 1 high address byte available on port 3 (only the three least significant bits are used). |
| P1.0–P1.7 | 13–17,<br>20–22 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. P0.3–P0.4 pins are bidirectional I/O port pins with internal pull-ups. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also serves the special function features of the SC80C51 family as listed below: |
| | 20 | I | **$\overline{INT0}$ (P1.5):** External interrupt. |
| | 21 | I | **$\overline{INT1}$ (P1.6):** External interrupt. |
| | 22 | I | **T0 (P1.7):** Timer 0 external input. |
| | 13–17 | I | **ADC0 (P1.0)–ADC4 (P1.4):** Port 1 also functions as the inputs to the five channel multiplexed A/D converter. These pins can be used as outputs only if the A/D function has been disabled. These pins can be used as digital inputs while the A/D converter is enabled.<br><br>Port 1 serves to output the addressed EPROM contents in the verify mode and accepts as inputs the value to program into the selected address during the program mode. |
| P3.0–P3.7 | 5–1,<br>27–25 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also functions as the address input for the EPROM memory location to be programmed (or verified). The 11-bit address is multiplexed into this port as specified by P0.0/ASEL. |
| RST | 9 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on RESET using only an external capacitor to $V_{CC}$. After the device is reset, a 10-bit serial sequence, sent LSB first, applied to RESET, places the device in the programming state allowing programming address, data and $V_{PP}$ to be applied for programming or verification purposes. The RESET serial sequence must be synchronized with the X1 input. |
| X1 | 11 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. X1 also serves as the clock to strobe in a serial bit stream into RESET to place the device in the programming state. |
| X2 | 10 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |
| $AV_{CC}$ [1] | 19 | I | **Analog supply voltage and reference input.** |
| $AV_{SS}$ [1] | 18 | I | **Analog supply and reference ground.** |

**NOTE:**

1. $AV_{SS}$ (reference ground) must be connected to 0V (ground). $AV_{CC}$ (reference input) cannot differ from $V_{CC}$ by more than ±0.2V, and must be in the range 4.5V to 5.5V.
2. When P0.2 is at or close to 0 volt, it may affect the internal ROM operation. We recommend that P0.2 be tied to $V_{CC}$ via a small pullup (e.g., 2kΩ).

## OSCILLATOR CHARACTERISTICS

X1 and X2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, X1 should be driven while X2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## IDLE MODE

The 8XC749 includes the 80C51 power-down and idle mode features. In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals except the A/D and PWM stay active. The functions that continue to run while in the idle mode are Timer 0, Timer I, and the interrupts. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset. Upon powering-up the circuit, or exiting from idle mode, sufficient time must be allowed for stabilization of the internal analog reference voltages before an A/D conversion is started.

### Special Function Registers

The special function registers (directly addressable only) contain all of the 8XC751 registers except the program counter and the four register banks. Most of the 21 special function registers are used to control the on-chip peripheral hardware. Other registers include arithmetic registers (ACC, B, PSW), stack pointer (SP) and data pointer registers (DPH, DPL). Nine of the SFRs are bit addressable.

### Data Pointer

The data pointer (DPTR) consists of a high byte (DPH) and a low byte (DPL). In the 80C51 this register allows the access of external data memory using the MOVX instruction. Since the 83C749 does not support MOVX or external memory accesses, this register is generally used as a 16-bit offset pointer of the accumulator in a MOVC instruction. DPTR may also be manipulated as two independent 8-bit registers.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. The control bits for the reduced power modes are in the special function register PCON.

**Table 1. External Pin Status During Idle and Power-Down Modes**

| MODE | Port 0* | Port 1 | Port 2 |
|---|---|---|---|
| Idle | Data | Data | Data |
| Power-down | Data | Data | Data |

* Except for PWM output (P0.4).

## DIFFERENCES BETWEEN THE 8XC749 AND THE 80C51

### Program Memory

On the 8XC749, program memory is 2048 bytes long and is not externally expandable, so the 80C51 instructions MOVX, LJMP, and LCALL are not implemented. If these instructions are executed, the appropriate number of instruction cycles will take place along with external fetches; however, no operation will take place. The LJMP may not respond to all program address bits. The only fixed locations in program memory are the addresses at which execution is taken up in response to reset and interrupts, which are as follows:

| | Program Memory |
|---|---|
| **Event** | **Address** |
| Reset | 000 |
| External $\overline{INT0}$ | 003 |
| Counter/timer 0 | 00B |
| External $\overline{INT1}$ | 013 |
| Timer I | 01B |
| ADC | 02B |
| PWM | 033 |

### Memory Organization

The 8XC749 manipulates operands in three memory address spaces. The first is the program memory space which contains program instructions as well as constants such as look-up tables. The program memory space contains 2k bytes in the 8XC749.

The second memory space is the data memory array which has a logical address space of 128 bytes. However, only the first 64 (0 to 3FH) are implemented in the 8XC749.

The third memory space is the special function register array having a 128-byte address space (80H to FFH). Only selected locations in this memory space are used (see Table 2). Note that the architecture of these memory spaces (internal program memory, internal data memory, and special function registers) is identical to the 80C51, and the 8XC749 varies only in the amount of memory physically implemented.

The 8XC749 does not directly address any external data or program memory spaces. For this reason, the MOVX instructions in the 80C51 instruction set are not implemented in the 83C749, nor are the alternate I/O pin functions $\overline{RD}$ and $\overline{WR}$.

### I/O Ports

The I/O pins provided by the 83C749 consist of port 0, port 1, and port 3.

#### Port 0

Port 0 is a 5-bit bidirectional I/O port and includes alternate functions on some pins of this port. Pins P0.3 and P0.4 are provided with internal pullups while the remaining pins (P0.0, P0.1, and P0.2) have open drain output structures. The alternate function for port P0.4 is PWM output.

If the alternate function PWM is not being used, then this pin may be used as an I/O port.

**Port 1**
Port 1 is an 8-bit bidirectional I/O port whose structure is identical to the 80C51, but also includes alternate input functions on all pins. The alternate pin functions for port 1 are:

P1.0-P1.4 - ADC0-ADC4 - A/D converter analog inputs
P1.5 $\overline{INT0}$ - external interrupt 0 input
P1.6 $\overline{INT1}$ - external interrupt 1 input
P1.7 - T0 - timer 0 external input

If the alternate functions $\overline{INT0}$, $\overline{INT1}$, or T0 are not being used, these pins may be used as standard I/O ports. It is necessary to connect $AV_{CC}$ and $AV_{SS}$ to $V_{CC}$ and $V_{SS}$, respectively, in order to use P1.5, P1.6, and P1.7 pins as standard I/O pins. When the A/D converter is enabled, the analog channel connected to the A/D may not be used as a digital input; however, the remaining analog inputs may be used as digital inputs. They may not be used as digital outputs. While the A/D is enabled, the analog inputs are floating.

**Port 3**
Port 3 is an 8-bit bidirectional I/O port whose structure is identical to the 80C51. Note that the alternate functions associated with port 3 of the 80C51 have been moved to port 1 of the 83C749 (as applicable). See Figure 1 for port bit configurations.

## Counter/Timer Subsystem

The 8XC749 has one counter/timer called timer/counter 0. Its operation is similar to mode 2 operation on the 80C51, but is extended to 16 bits with 16 bits of autoload. The controls for this counter are centralized in a single register called TCON.

**Timer I Implementation**
Timer I is clocked once per machine cycle, which is the oscillator frequency divided by 12. The timer operation is enabled by setting the TIRUN bit (bit 4) in the I2CFG register. Writing a 0 into the TIRUN bit will stop and clear the timer. The timer is 10 bits wide, and when it reaches the terminal count of 1024, it carries out and sets the Timer I interrupt flag. An interrupt will occur if the Timer I interrupt is enabled by bit ETI (bit 4) of the Interrupt Enable (IE) register, and global interrupts are enabled by bit EA (bit 7) of the same IE register.

The vector address for the Timer I interrupt is 1Bhex, and the interrupt service routine must start at this address. As with all 8051 family microcontrollers, only the Program Counter is pushed onto the stack upon interrupt (other registers that are used both by the interrupt service routine and elsewhere must be explicitly saved). The Timer I interrupt flag is cleared by setting the CKRTI bit (bit 5 of the I1CFG register. For more information, see application note AN427.

## Interrupt Subsystem—Fixed Priority

The IP register and the 2-level interrupt system of the 80C51 are eliminated. The interrupt structure is a seven-source, one-level interrupt system similar to the 8XC751. Simultaneous interrupt conditions are resolved by a single-level, fixed priority as follows:

| | |
|---|---|
| Highest priority: | Pin $\overline{INT0}$ |
| | Counter/timer flag 0 |
| | Pin $\overline{INT1}$ |
| | PWM |
| | Timer I |
| Lowest priority: | ADC |

The vector addresses are as follows:

| Source | Vector Address |
|---|---|
| INT0 | 0003H |
| TF0 | 000BH |
| INT1 | 0013H |
| TIMER I | 001BH |
| ADC | 002BH |
| PWM | 0033H |

**Interrupt Control Registers**
The 80C51 interrupt enable register is modified to take into account the different interrupt sources of the 8XC749.

**Interrupt Enable Register**

MSB LSB

| EA | EAD | ETI | — | EPWM | EX1 | ET0 | EX0 |
|---|---|---|---|---|---|---|---|

| Position | Symbol | Function |
|---|---|---|
| IE.7 | EA | Global interrupt disable when EA = 0 |
| IE.6 | EAD | A/D conversion complete |
| IE.5 | ETI | Timer I |
| IE.4 | — | |
| IE.3 | EPWM | PWM counter overflow |
| IE.2 | EX1 | External interrupt 1 |
| IE.1 | ET0 | Timer 0 overflow |
| IE.0 | EX0 | External interrupt 0 |

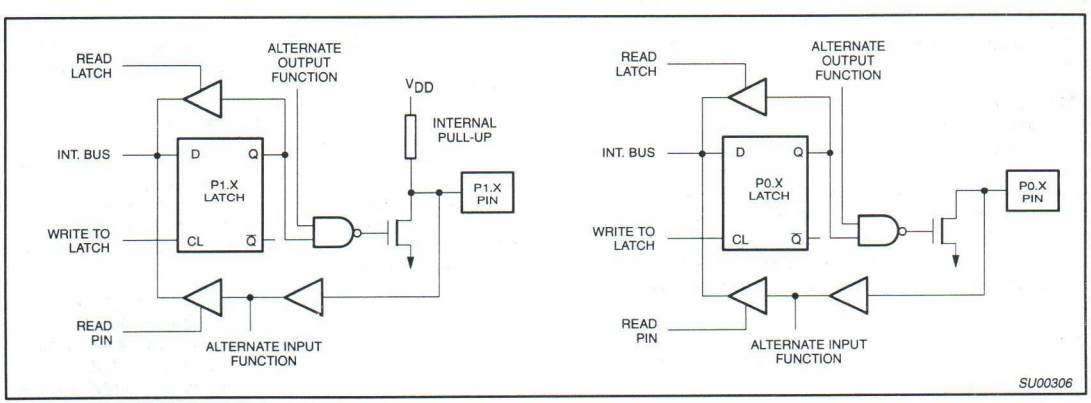



**Figure 1. Port Bit Latches and I/O Buffers**

## Pulse Width Modulation Output (P0.4)

The PWM outputs pulses of programmable length and interval. The repetition frequency is defined by an 8-bit prescaler which generates the clock for the counter. The prescaler register is PWMP. The prescaler and counter are not associated with any other timer. The 8-bit counter counts modulo 255, that is from 0 to 254 inclusive. The value of the 8-bit counter is compared to the contents of a compare register, PWM. When the counter value matches the contents of this register, the output of the PWM is set high. When the counter reaches zero, the output of the PWM is set low. The pulse width ratio (duty cycle) is defined by the contents of the compare register and is in the range of 0 to 1 programmed in increments of 1/255. The PWM output can be set to be continuously high by loading the compare register with 0 and the output can be set to be continuously low by loading the compare register with 255. The PWM output is enabled by a bit in a special function register, PWENA. When enabled, the pin output is driven with a fully active pull-up. That is, when the output is high, a strong pull-up is continuously applied. When disabled, the pin functions as a normal bidirectional I/O pin, however, the counter remains active.

The PWM function is disabled during RESET and remains disabled after reset is removed until re-enabled by software. The PWM output is high during power down and idle. The counter is disabled during idle. The repetition frequency of the PWM is given by:

$$f_{PWM} = f_{OSC} / 2 (1 + PWMP) 255$$

The low/high ratio of the PWM signal is PWM / (255 – PWM) for PWM not equal to 255. For PWM = 255, the output is always low.

The repetition frequency range is 92Hz to 23.5kHz for an oscillator frequency of 12MHz.

An interrupt will be asserted upon PWM counter overflow if the interrupt is not masked off.

The PWM output is an alternative function of P0.4. In order to use this port as a bidirectional I/O port, the PWM output must be disabled by clearing the enable/disable bit in PWENA. In this case, the PWM subsystem can be used as an interval timer by enabling the PWM interrupt.

## Special Function Register Addresses

Special function registers for the 8XC749 are identical to those of the 80C51, except for the changes listed below:

80C51 special function registers not present in the 8XC749 are TMOD (89), P2 (A0) and IP (B8). Additional special function registers are ADCON (A0), ADAT (84), PWM (8E), PWMP (8F), and PWENA (FE).

### A/D Converter

The analog input circuitry consists of a 5-input analog multiplexer and an A to D converter with 8-bit resolution. The conversion takes 40 machine cycles, i.e., 40µs at 12MHz oscillator frequency. The A/D converter is controlled using the ADCON control register. Input channels are selected by the analog multiplexer through ADCON register bits 0–2.

The 83C749 contains a five-channel multiplexed 8-bit A/D converter. The conversion requires 40 machine cycles (40µs at 12MHz oscillator frequency).

The A/D converter is controlled by the A/D control register, ADCON. Input channels are selected by the analog multiplexer by bits ADCON.0 through ADCON.2. The ADCON register is not bit addressable.

**ADCON Register**

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| X | X | ENADC | ADCI | ADCS | AADR2 | AADR1 | AADR0 |

| ADCI | ADCS | Operation |
|---|---|---|
| 0 | 0 | ADC not busy, a conversion can be started. |
| 0 | 1 | ADC busy, start of a new conversion is blocked. |
| 1 | 0 | Conversion completed, start of a new conversion is blocked. |
| 1 | 1 | Not possible. |

| INPUT CHANNEL SELECTION | | | |
|---|---|---|---|
| ADDR2 | ADDR1 | ADDR0 | INPUT PIN |
| 0 | 0 | 0 | P1.0 |
| 0 | 0 | 1 | P1.1 |
| 0 | 1 | 0 | P1.2 |
| 0 | 1 | 1 | P1.3 |
| 1 | 0 | 0 | P1.4 |

| Position | Symbol | Function |
|---|---|---|
| ADCON.5 | ENADC | Enable A/D function when ENADC = 1. Reset forces ENADC = 0. |
| ADCON.4 | ADCI | ADC interrupt flag. This flag is set when an ADC conversion is complete. If IE.6 = 1, an interrupt is requested when ADCI = 1. The ADCI flag is cleared when conversion data is read. This flag is read only. |
| ADCON.3 | ADCS | ADC start. Setting this bit starts an A/D conversion. Once set, ADCS remains high throughout the conversion cycle. On completion of the conversion, it is reset just before the ADCI interrupt flag is cleared. ADCS cannot be reset by software. ADCS should not be used to monitor the A/D converter status. ADCI should be used for this purpose. |
| ADCON.2 | AADR2 | Analog input select. |
| ADCON.1 | AADR1 | Analog input select. |
| ADCON.0 | AADR0 | Analog input select. This binary coded address selects one of the five analog input port pins of P1 to be input to the converter. It can only be changed when ADCI and ADCS are both low. AADR2 is the most significant bit. |

The completion of the 8-bit ADC conversion is flagged by ADCI in the ADCON register, and the result is stored in the special function register ADAT.

An ADC conversion in progress is unaffected by an ADC start. The result of a completed conversion remains unaffected provided ADCI remains at a logic 1. While ADCS is a logic 1 or ADCI is a logic 1, a new ADC START will be blocked and consequently lost. An ADC conversion in progress is aborted when the idle or power-down mode is entered. The result of a completed conversion (ADCI = logic 1) remains unaffected when entering the idle mode. See Figure 2 for an A/D input equivalent circuit.

The analog input pins ADC0-ADC4 may be used as digital inputs and outputs when the A/D converter is disabled by a 0 in the ENADC bit in ADCON. When the A/D is enabled, the analog input channel that is selected by the ADDR2-ADDR0 bits in ADCON cannot be used as a digital input. Reading the selected A/D channel as a digital input will always return a 1. The unselected A/D inputs

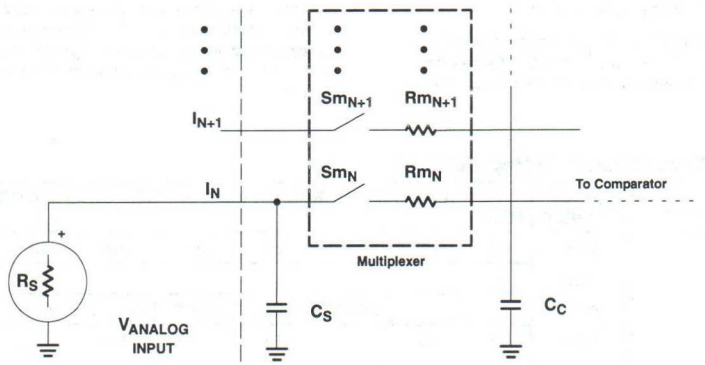
may always be used as digital inputs. Unselected analog inputs will be floating and may not be used as digital outputs.

The A/D reference inputs on the 8XC749 are tied together with the analog supply pins $AV_{CC}$ and $AV_{SS}$. This means that the reference voltage on the A/D cannot be varied separately from the analog supply pins. $AV_{SS}$ must be connected to 0V and $AV_{CC}$ must be connected to a supply voltage between 4.5V and 5.5V. A/D measurements may be made in the range of 4.5V to 5.5V. Increasing the voltage on the A/D ground reference above 0V or reducing the voltage on the positive A/D reference below 4.5V is not permitted.

**Table 2. 8XC749 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| ADAT# | A/D result | 84H | | | | | | | | | 00H |
| ADCON# | A/D control | A0H | – | – | ENADC | ADCI | ADCS | AADR2 | AADR1 | AADR0 | C0H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes) | | | | | | | | | | |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | EAD | ETI | — | EPWM | EX1 | ET0 | EX0 | 00H |
| | | | – | – | – | 84 | 83 | 82 | 81 | 80 | |
| P0*# | Port 0 | 80H | – | – | – | PWM0 | – | – | – | – | xxx11111B |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | FFH |
| P1*# | Port 1 | 90H | T0 | $\overline{INT1}$ | INT0 | ADC4 | ADC3 | ADC2 | ADC1 | ADC0 | |
| P3* | Port 3 | B0H | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | FFH |
| PCON# | Power control | 87H | – | – | – | – | – | – | PD | IDL | xxxx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| PWCM# | PWM compare | 8EH | | | | | | | | | xxxxxxxxB |
| PWENA# | PWM enable | FEH | – | – | – | – | – | – | – | PWE | FEH |
| PWMP# | PWM prescaler | 8FH | | | | | | | | | 00H |
| RTL# | Timer low reload | 8BH | | | | | | | | | 00H |
| RTH# | Timer high reload | 8DH | | | | | | | | | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| TL# | Timer low | 8AH | | | | | | | | | 00H |
| TH# | Timer high | 8CH | | | | | | | | | 00H |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| TICON*# | Timer I control | D8H/RD | – | – | 0 | TIRUN | – | – | – | – | 0000xx00B |
| | | WR | – | – | CLRTI | TIRUN | – | – | – | – | |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON*# | Timer control | 88H | GATE | C/T | TF | TR | IE0 | IT0 | IE1 | IT1 | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.




Rm = 0.5 - 3 kΩ
CS + CC = 15pF maximum
RS = Recommended < 9.6 kΩ for 1 LSB @ 12MHz

**NOTE:**
Because the analog to digital converter has a sampled-data comparator, the input looks capacitive to a source. When a conversion is initiated, switch Sm closes for 8tcy (8µs @ 12MHz crystal frequency) during which time capacitance Cs + Cc is charged. It should be noted that the sampling causes the analog input to present a varying load to an analog source.

SU00199

**Figure 2. A/D Input: Equivalent Circuit**

## A/D CONVERTER PARAMETER DEFINITIONS

The following definitions are included to clarify some specifications given and do not represent a complete set of A/D parameter definitions.

### Absolute Accuracy Error

Absolute accuracy error of a given output is the difference between the theoretical analog input voltage to produce a given output and the actual analog input voltage required to produce the same code. Since the same output code is produced by a band of input voltages, the "required input voltage" is defined as the midpoint of the band of input voltage that will produce that code. Absolute accuracy error not specified with a code is the maximum over all codes.

### Nonlinearity

If a straight line is drawn between the end points of the actual converter characteristics such that zero offset and full scale errors are removed, then non-linearity is the maximum deviation of the code transitions of the actual characteristics from that of the straight line so constructed. This is also referred to as relative accuracy and also integral non-linearity.

### Differential Non-Linearity

Differential non-linearity is the maximum difference between the actual and ideal code widths of the converter. The code widths are the differences expressed in LSB between the code transition points, as the input voltage is varied through the range for the complete set of codes.

### Gain Error

Gain error is the deviation between the ideal and actual analog input voltage required to cause the final code transition to a full-scale output code after the offset error has been removed. This may sometimes be referred to as full scale error.

### Offset Error

Offset error is the difference between the actual input voltage that causes the first code transition and the ideal value to cause the first code transition. This ideal value is 1/2 LSB above $V_{ref-}$.

### Channel to Channel Matching

Channel to channel matching is the maximum difference between the corresponding code transitions of the actual characteristics taken from different channels under the same temperature, voltage and frequency conditions.

### Crosstalk

Crosstalk is the measured level of a signal at the output of the converter resulting from a signal applied to one deselected channel.

### Total Error

Maximum deviation of any step point from a line connecting the ideal first transition point to the ideal last transition point.

### Relative Accuracy

Relative accuracy error is the deviation of the ADC's actual code transition points from the ideal code transition points on a straight line which connects the ideal first code transition point and the final code transition point, after nullifying offset error and gain error. It is generally expressed in LSBs or in percent of FSR.

## COUNTER/TIMER

The 8XC749 counter/timer is designated Timer 0 and is separate from Timer I and from the PWM. Its operation is similar to mode 2 of the 80C51 counter/timer, extended to 16 bits. When Timer 0 is used in the external counter mode, the T0 input (P1.7) is sampled every S4P1. The counter/timer function is controlled using the timer control register (TCON).

### TCON Register

MSB LSB

| GATE | C/T | TF | TR | IE0 | IT0 | IE1 | IT1 |
|---|---|---|---|---|---|---|---|

| Position | Symbol | Function |
|---|---|---|
| TCON.7 | GATE | 1 – Timer 0 is enabled only when INT0 pin is high and TR is 1. |
| | | 0 – Timer 0 is enabled only when TR is 1. |
| TCON.6 | C/T | 1 – Counter operation from T0 pin. |
| | | 0 – Timer operation from internal clock. |
| TCON.5 | TF | 1 – Set on overflow of T0. |
| | | 0 – Cleared when processor vectors to interrupt routine and by reset. |
| TCON.4 | TR | 1 – Enable timer 0 |
| | | 0 – Disable timer 0 |
| TCON.3 | IE0 | 1 – Edge detected on INT0 |
| TCON.2 | IT0 | 1 – INT0 is edge triggered. |
| | | 0 – INT0 is level sensitive. |
| TCON.1 | IE1 | 1 – Edge detected on INT1 |
| TCON.0 | IT1 | 1 – INT1 is edge triggered. |
| | | 0 – INT1 is level sensitive. |

These flags are functionally identical to the corresponding 80C51 flags except that there is only one of the 80C51 style timers, and the flags are combined into one register.

Note that the positions of the IE0/IT0 and IE1/IT1 bits are transposed from the positions used in the standard 80C51 TCON register.

Timer I may be used as a fixed time base timer or watchdog timer.

Timer T0 is a 16-bit autoreloadable timer/counter, that operates similar to mode 2 operation on the 80C51, but is extended to 16 bits. The timer/counter is clocked by either 1/12 the oscillator frequency or by transitions on the T0 pin. The C/T bit in special function register TCON selects between these two modes. When the TCON TR bit is set, the timer/counter is enabled. Register pair TH and TL are incremented by the clock source. When the register pair overflows, the register pair is reloaded with the values in registers RTH and RTL. The value in the reload registers is left unchanged. The TF bit in special function register TCON is set on counter overflow and, if the interrupt is enabled, will generate an interrupt (see Figure 3).

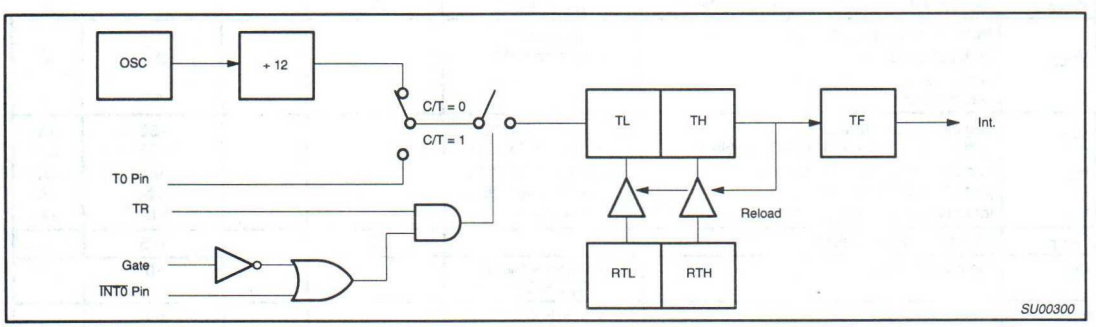



**Figure 3. 83C749 Counter/Timer Block Diagram**

## ABSOLUTE MAXIMUM RATINGS[1, 3, 4]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage from $V_{CC}$ to $V_{SS}$ | –0.5 to +6.5 | V |
| Voltage from any pin to $V_{SS}$ (except $V_{PP}$) | –0.5 to $V_{CC}$ + 0.5 | V |
| Power dissipation | 1.0 | W |
| Voltage from $V_{PP}$ pin to $V_{SS}$ | –0.5 to + 13.0 | V |

**NOTES ON PAGE 3-1105.**

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $AV_{CC}$ = 5V ±5, $AV_{SS}$ = 0V[4]
$V_{CC}$ = 5V ± 10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS[4] MIN | TYP[1] | MAX | UNIT |
|---|---|---|---|---|---|---|
| $I_{CC}$ | Supply current (see Figure 6) | | | | | |
| **Inputs** | | | | | | |
| $V_{IL}$ | Input low voltage | (0 to 70°C)<br>(–40 to +85°C) | –0.5<br>–0.5 | | $0.2V_{CC}$–0.1<br>$0.2V_{CC}$–0.15 | V<br>V |
| $V_{IH}$ | Input high voltage, except X1, RST | (0 to 70°C)<br>($0.2V_{CC}$+1) | $0.2V_{CC}$+0.9<br>(–40 to +85°C) | | $V_{CC}$+0.5<br>$V_{CC}$+0.5 | V<br>V |
| $V_{IH1}$ | Input high voltage, X1, RST | (0 to 70°C)<br>(–40 to +85°C) | $0.7V_{CC}$<br>$0.7V_{CC}$ to 0.1 | | $V_{CC}$+0.5<br>$V_{CC}$+0.5 | V<br>V |
| | P0.2 | | | | | |
| $V_{IL1}$ | Input low voltage | (0 to 70°C)<br>(–40 to +85°C) | –0.5<br>–0.5 | | $0.3V_{CC}$<br>$0.3V_{CC}$–0.1 | V<br>V |
| $V_{IH2}$ | Input high voltage | (0 to 70°C)<br>(–40 to –85°C) | $0.7V_{CC}$<br>$0.7V_{CC}$+0.1 | | $V_{CC}$+0.5<br>$V_{CC}$+0.5 | V |
| **Outputs** | | | | | | |
| $V_{OL}$ | Output low voltage, ports 1, 3, 0.3, and 0.4 (PWM disabled) | $I_{OL}$ = 1.6mA[2] | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0.2 | $I_{OL}$ = 3.2mA[2] | | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1, 3, 0.3, and 0.4 (PWM disabled) | $I_{OH}$ = –60µA,<br>$I_{OH}$ = –25µA<br>$I_{OH}$ = –10µA<br>$I_{OH}$ = –400µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$<br>2.4 | | | V<br>V<br>V<br>V |
| $V_{OH2}$ | Output high voltage, P0.4 (PWM enabled) | $I_{OH}$ = –40µA | $0.9V_{CC}$ | | | V |
| $V_{OL2}$ | Port 0.0 and 0.1 – Drivers<br>Output low voltage | $I_{OL}$ = 3mA<br>(over $V_{CC}$ range) | | | 0.4 | V |
| C | Driver, receiver combined:<br>Capacitance | | | | 10 | pF |
| $I_{IL}$ | Logical 0 input current, ports 1, 3, 0.3, and 0.4 (PWM disabled)[11] | $V_{IN}$ = 0.45V (0 to 70°C)<br>$V_{IN}$ = 0.45V (0 to +85°C) | | | –50<br>–75 | µA<br>µA |
| $I_{TL}$ | Logical 1 to 0 transition current, ports 1, 3, 0.3 and 0.4[11] | $V_{IN}$ = 2V (0 to 70°C)<br>$V_{IN}$ = 2V (–40 to +85°C) | | | –650<br>–750 | µA<br>µA |
| $I_{LI}$ | Input leakage current, port 0.0, 0.1 and 0.2 | 0.45 < $V_{IN}$ < $V_{CC}$ | | | ±10 | µA |
| $R_{RST}$ | Reset pull-down resistor | | 25 | | 175 | kΩ |
| $C_{IO}$ | Pin capacitance | Test freq = 1MHz,<br>$T_{amb}$ = 25°C | | | 10 | pF |
| $I_{PD}$ | Power-down current[5] | $V_{CC}$ = 2 to 5.5V<br>$V_{CC}$ = 2 to 6.0V (83C749) | | | 50 | µA |
| $V_{PP}$ | $V_{PP}$ program voltage (87C749 only) | $V_{SS}$ = 0V<br>$V_{CC}$ = 5V±10%<br>$T_{amb}$ = 21°C to 27°C | 12.5 | | 13.0 | V |
| $I_{PP}$ | Program current (87C749 only) | $V_{PP}$ = 13.0V | | | 50 | mA |
| **Analog Inputs** (A/D guaranteed only with quartz window covered.) | | | | | | |
| $AV_{CC}$ | Analog supply voltage[10] | $AV_{CC}$ = $V_{CC}$±0.2V | 4.5 | | 5.5 | V |
| $AI_{CC}$ | Analog operating supply current | $AV_{CC}$ = 5.12V | | | 3[9] | mA |
| $AV_{IN}$ | Analog input voltage | | $AV_{SS}$–0.2 | | $AV_{CC}$+0.2 | V |
| $C_{IA}$ | Analog input capacitance | | | | 15 | pF |
| $t_{ADS}$ | Sampling time | | | | $8t_{CY}$ | s |
| $t_{ADC}$ | Conversion time | | | | $40t_{CY}$ | s |

**NOTES ON FOLLOWING PAGE.**

## DC ELECTRICAL CHARACTERISTICS (Continued)

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $AV_{CC}$ = 5V ±5, $AV_{SS}$ = 0V[4]
$V_{CC}$ = 5V ± 10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS[4] MIN | TYP[1] | MAX | UNIT |
|---|---|---|---|---|---|---|
| **Analog Inputs** (A/D guaranteed only with quartz window covered.) (Continued) | | | | | | |
| R | Resolution | | | | 8 | bits |
| $E_{RA}$ | Relative accuracy | | | | ±1 | LSB |
| $OS_e$ | Zero scale offset | | | | ±1 | LSB |
| $G_e$ | Full scale gain error | | | | 0.4 | % |
| $M_{CTC}$ | Channel to channel matching | | | | ±1 | LSB |
| $C_t$ | Crosstalk | 0–100kHz | | | –60 | dB |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 10mA (NOTE: This is 85°C spec.)
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 67mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
3. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
4. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.
5. Power-down $I_{CC}$ is measured with all output pins disconnected; port 0 = $V_{CC}$; X2, X1 n.c.; RST = $V_{SS}$.
6. $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; RST = port 0 = $V_{CC}$. $I_{CC}$ will be slightly higher if a crystal oscillator is used.
7. Idle $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; port 0 = $V_{CC}$; RST = $V_{SS}$.
8. Load capacitance for ports = 80pF.
9. The resistor ladder network is not disconnected in the power down or idle modes. Thus, to conserve power, the user may remove $AV_{CC}$.
10. If the A/D function is not required, or if the A/D function is only needed periodically, $AV_{CC}$ may be removed without affecting the operation of the digital circuitry. Contents of ADCON and ADAT are not guaranteed to be valid. If $AV_{CC}$ is removed, the A/D inputs must be lowered to less than 0.5V. Digital inputs on P1.0–P1.4 will not function normally.
11. These parameters do not apply to P1.0–P1.4 if the A/D function is enabled.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[4, 8]

| SYMBOL | PARAMETER | 16MHz CLOCK MIN | 16MHz CLOCK MAX | VARIABLE CLOCK MIN | VARIABLE CLOCK MAX | UNIT |
|---|---|---|---|---|---|---|
| $1/t_{CLCL}$ | Oscillator frequency: | | | 3.5 | 16 | MHz |
| **External Clock (Figure 4)** | | | | | | |
| $t_{CHCX}$ | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | Fall time | | 20 | | 20 | ns |

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

C – Clock
D – Input data
H – Logic level high
L – Logic level low
Q – Output data
T – Time
V – Valid
X – No longer a valid logic level
Z – Float

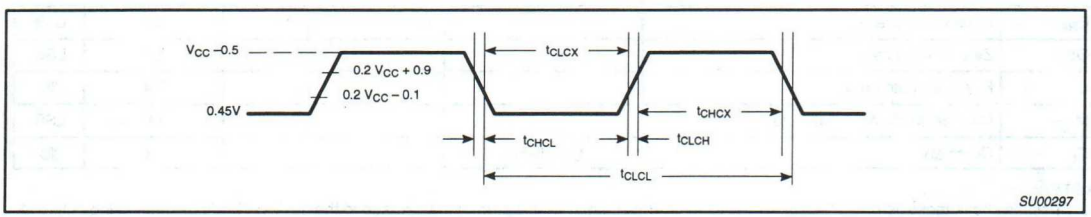



**Figure 4. External Clock Drive**

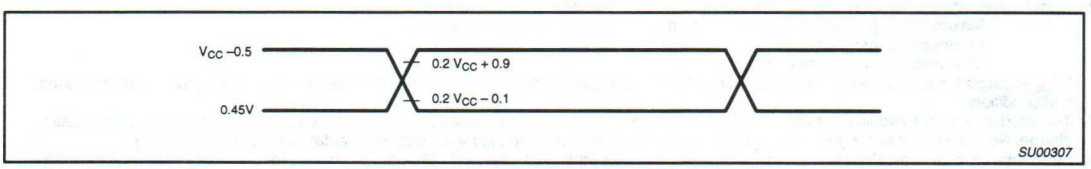



**Figure 5. AC Testing Input/Output**

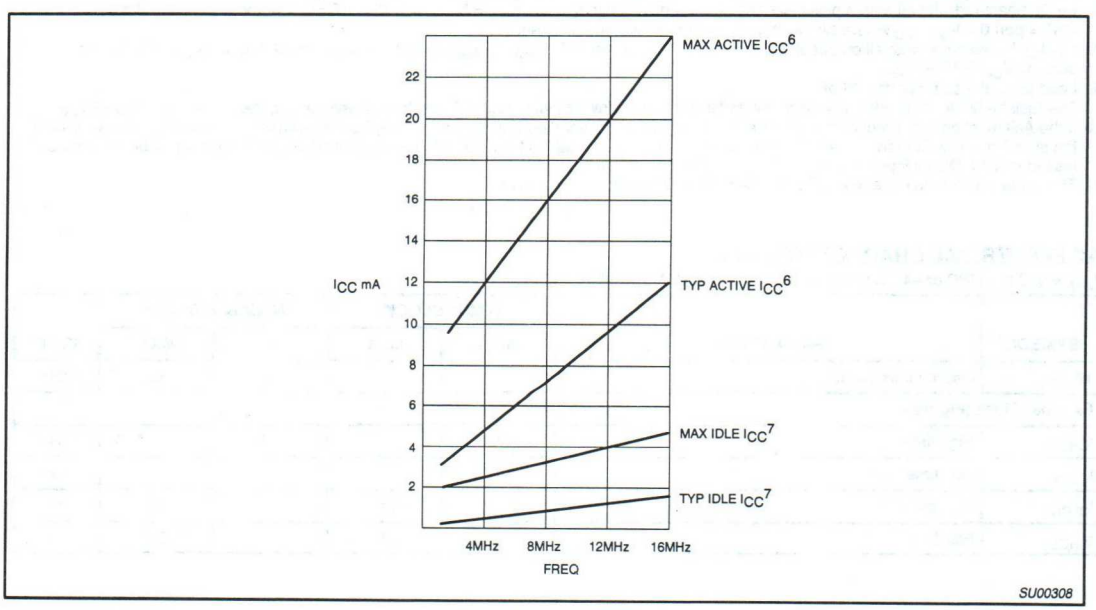



**Figure 6. $I_{CC}$ vs. FREQ**
**Maximum $I_{CC}$ values taken at $V_{CC}$ = 5.5V and worst case temperature.**
**Typical $I_{CC}$ values taken at $V_{CC}$ = 5.0V and 25°C.**
**Notes 6 and 7 refer to AC Electrical Characteristics.**

## PROGRAMMING CONSIDERATIONS

### EPROM Characteristics

The 87C749 is programmed by using a modified Quick-Pulse Programming algorithm similar to that used for devices such as the 87C451 and 87C51. It differs from these devices in that a serial data stream is used to place the 87C749 in the programming mode.

Figure 7 shows a block diagram of the programming configuration for the 87C749. Port pin P0.2 is used as the programming voltage supply input ($V_{PP}$ signal). Port pin P0.1 is used as the program (PGM/) signal. This pin is used for the 25 programming pulses.

Port 3 is used as the address input for the byte to be programmed and accepts both the high and low components of the eleven bit address. Multiplexing of these address components is performed using the ASEL input. The user should drive the ASEL input high and then drive port 3 with the high order bits of the address. ASEL should remain high for at least 13 clock cycles. ASEL may then be driven low which latches the high order bits of the address internally. The high address should remain on port 3 for at least two clock cycles after ASEL is driven low. Port 3 may then be driven with the low byte of the address. The low address will be internally stable 13 clock cycles later. The address will remain stable provided that the low byte placed on port 3 is held stable and ASEL is kept low. **Note:** ASEL needs to be pulsed high only to change the high byte of the address.

Port 1 is used as a bidirectional data bus during programming and verify operations. During programming mode, it accepts the byte to be programmed. During verify mode, it provides the contents of the EPROM location specified by the address which has been supplied to Port 3.

The XTAL1 pin is the oscillator input and receives the master system clock. This clock should be between 1.2 and 6MHz.

The RESET pin is used to accept the serial data stream that places the 87C749 into various programming modes. This pattern consists of a 10-bit code with the LSB sent first. Each bit is synchronized to the clock input, X1.

### Programming Operation

Figures 8 and 9 show the timing diagrams for the program/verify cycle. RESET should initially be held high for at least two machine cycles. P0.1 (PGM/) and P0.2 ($V_{PP}$) will be at $V_{OH}$ as a result of the RESET operation. At this point, these pins function as normal quasi-bidirectional I/O ports and the programming equipment may pull these lines low. However, prior to sending the 10-bit code on the RESET pin, the programming equipment should drive these pins high ($V_{IH}$). The RESET pin may now be used as the serial data input for the data stream which places the 87C749 in the programming mode. Data bits are sampled during the clock high time and thus should only change during the time that the clock is low. Following transmission of the last data bit, the RESET pin should be held low.

Next the address information for the location to be programmed is placed on port 3 and ASEL is used to perform the address multiplexing, as previously described. At this time, port 1 functions as an output.

A high voltage $V_{PP}$ level is then applied to the $V_{PP}$ input (P0.2). (This sets Port 1 as an input port). The data to be programmed into the EPROM array is then placed on Port 1. This is followed by a series of programming pulses applied to the PGM/ pin (P0.1). These pulses are created by driving P0.1 low and then high. This pulse is repeated until a total of 25 programming pulses have occurred. At the conclusion of the last pulse, the PGM/ signal should remain high.

The $V_{PP}$ signal may now be driven to the $V_{OH}$ level, placing the 87C749 in the verify mode. (Port 1 is now used as an output port). After four machine cycles (48 clock periods), the contents of the addressed location in the EPROM array will appear on Port 1.

The next programming cycle may now be initiated by placing the address information at the inputs of the multiplexed buffers, driving the $V_{PP}$ pin to the $V_{PP}$ voltage level, providing the byte to be programmed to Port1 and issuing the 26 programming pulses on the PGM/ pin, bringing $V_{PP}$ back down to the $V_C$ level and verifying the byte.

### Programming Modes

The 87C749 has four programming features incorporated within its EPROM array. These include the USER EPROM for storage of the application's code, a 16-byte encryption key array and two security bits. Programming and verification of these four elements are selected by a combination of the serial data stream applied to the RESET pin and the voltage levels applied to port pins P0.1 and P0.2. The various combinations are shown in Table 3.

### Encryption Key Table

The 87C749 includes a 16-byte EPROM array that is programmable by the end user. The contents of this array can then be used to encrypt the program memory contents during a program memory verify operation. When a program memory verify operation is performed, the contents of the program memory location is XNOR'ed with one of the bytes in the 16-byte encryption table. The resulting data pattern is then provided to port 1 as the verify data. The encryption mechanism can be disable, in essence, by leaving the bytes in the encryption table in their erased state (FFH) since the XNOR product of a bit with a logical one will result in the original bit. The encryption bytes are mapped with the code memory in 16-byte groups. the first byte in code memory will be encrypted with the first byte in the encryption table; the second byte in code memory will be encrypted with the second byte in the encryption table and so forth up to and including the 16the byte. The encryption repeats in 16-byte groups; the 17th byte in the code memory will be encrypted with the first byte in the encryption table, and so forth.

### Security Bits

Two security bits, security bit 1 and security bit 2, are provided to limit access to the USER EPROM and encryption key arrays. Security bit 1 is the program inhibit bit, and once programmed performs the following functions:

1. Additional programming of the USER EPROM is inhibited.
2. Additional programming of the encryption key is inhibited.
3. Verification of the encryption key is inhibited.
4. Verification of the USER EPROM and the security bit levels may still be performed.

(If the encryption key array is being used, this security bit should be programmed by the user to prevent unauthorized parties from reprogramming the encryption key to all logical zero bits. Such programming would provide data during a verify cycle that is the logical complement of the USER EPROM contents).

Security bit 2, the verify inhibit bit, prevents verification of both the USER EPROM array and the encryption key arrays. The security bit levels may still be verified.

### Programming and Verifying Security Bits

Security bits are programmed employing the same techniques used to program the USER EPROM and KEY arrays using serial data streams and logic levels on port pins indicated in Table 3. When programming either security bit, it is not necessary to provide address or data information to the 87C749 on ports 1 and 3.

Verification occurs in a similar manner using the RESET serial stream shown in Table 3. Port 3 is not required to be driven and the results of the verify operation will appear on ports 1.6 and 1.7.

Ports 1.7 contains the security bit 1 data and is a logical one if programmed and a logical zero if erased. Likewise, P1.6 contains the security bit 2 data and is a logical one if programmed and a logical zero if erased.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Flourless part number 2345–5 or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-sec/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000uW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

### Table 3. Implementing Program/Verify Modes

| OPERATION | SERIAL CODE | P0.1 (PGM/) | P0.2 ($V_{PP}$) |
|---|---|---|---|
| Program user EPROM | 296H | –* | $V_{PP}$ |
| Verify user EPROM | 296H | $V_{IH}$ | $V_{IH}$ |
| Program key EPROM | 292H | –* | $V_{PP}$ |
| Verify key EPROM | 292H | $V_{IH}$ | $V_{IH}$ |
| Program security bit 1 | 29AH | –* | $V_{PP}$ |
| Program security bit 2 | 298H | –* | $V_{PP}$ |
| Verify security bits | 29AH | $V_{IH}$ | $V_{IH}$ |

**NOTE:**
* Pulsed from $V_{IH}$ to $V_{IL}$ and returned to $V_{IH}$.

## EPROM PROGRAMMING AND VERIFICATION

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $1/t_{CLCL}$ | Oscillator/clock frequency | 1.2 | 6 | MHz |
| $t_{AVGL}$[1] | Address setup to P0.1 (PROG–) low | $10\mu s + 24t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after P0.1 (PROG–) high | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to P0.1 (PROG–) low | $38t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to P0.1 (PROG–) low | $38t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after P0.1 (PROG–) high | $36t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to P0.1 (PROG–) low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after P0.1 (PROG–) | 10 | | µs |
| $t_{GLGH}$ | P0.1 (PROG–) width | 90 | 110 | µs |
| $t_{AVQV}$[2] | $V_{PP}$ low ($V_{CC}$) to data valid | | $48t_{CLCL}$ | |
| $t_{GHGL}$ | P0.1 (PROG–) high to P0.1 (PROG–) low | 10 | | µs |
| $t_{SYNL}$ | P0.0 (sync pulse) low | $4t_{CLCL}$ | | |
| $t_{SYNH}$ | P0.0 (sync pulse) high | $8t_{CLCL}$ | | |
| $t_{MASEL}$ | ASEL high time | $13t_{CLCL}$ | | |
| $t_{MAHLD}$ | Address hold time | $2t_{CLCL}$ | | |
| $t_{HASET}$ | Address setup to ASEL | $13t_{CLCL}$ | | |
| $t_{ADSTA}$ | Low address to address stable | $13t_{CLCL}$ | | |

**NOTES:**
1. Address should be valid at least $24t_{CLCL}$ before the rising edge of P0.2 ($V_{PP}$).
2. For a pure verify mode, i.e., no program mode in between, $t_{AVQV}$ is $14t_{CLCL}$ maximum.

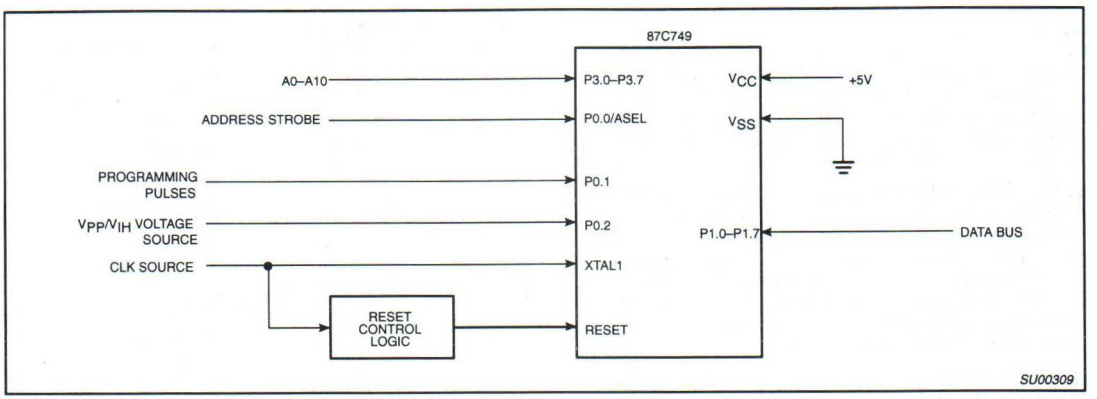



**Figure 7. Programming Configuration**

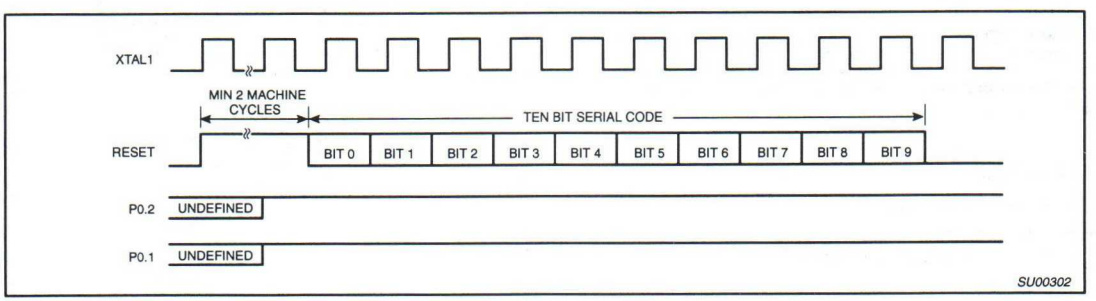



**Figure 8. Entry into Program/Verify Modes**

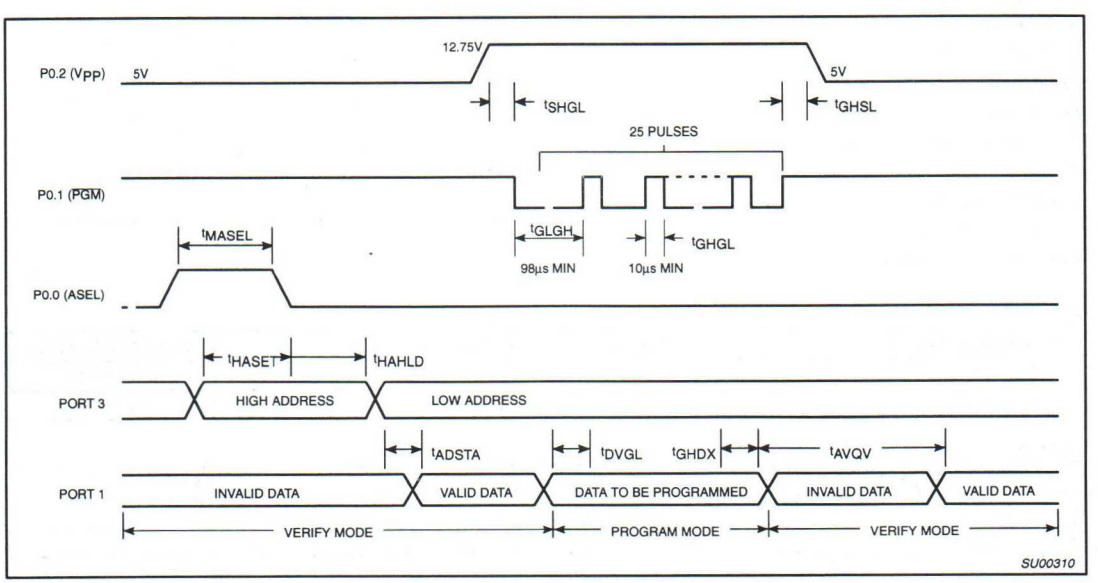



**Figure 9. Program/Verify Cycle**

# Microcontroller with TrackPoint™ microcode from IBM TPM749

## DESCRIPTION

The Philips Semiconductors TPM749 is a small package, low cost, ROM-coded 80C51 with IBM®'s TrackPoint™ pointing algorithms and control code. TrackPoint is the result of years of human factors research and innovation at IBM. The result is a "velocity sensitive" pointing solution more efficient and easier to use than "position sensitive" devices such as the mouse, the trackball, or the touchpad.

IBM has licensed Philips Semiconductors to sell microcontrollers with TrackPoint code. By purchasing a TPM from Philips, the purchaser becomes a sub-licensee of Philips. The selling price of Philips' TPM includes the royalties for IBM's intellectual property, which Philips in turn pays to IBM. Customers for TPMs do not need to sign any licensing agreement with either IBM or Philips. This code is the intellectual property of IBM, which is covered by numerous patents, and must be treated accordingly.

The TPM is fabricated with Philips high-density CMOS technology. Philips epitaxial substrate minimizes CMOS latch-up sensitivity.

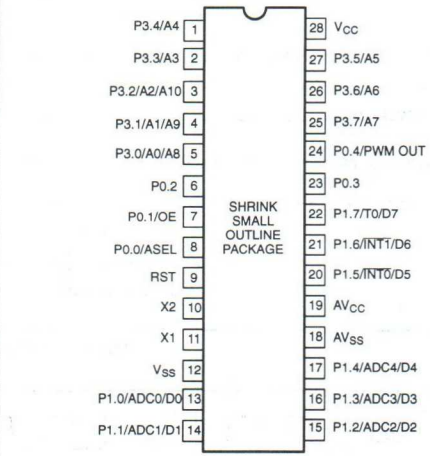
The TPM contains a 2k × 8 ROM, a 64 × 8 RAM, 21 I/O lines, a 16-bit auto-reload counter/timer, a fixed-priority level interrupt structure, an on-chip oscillator, a five channel multiplexed 8-bit A/D converter, and an 8-bit PWM output.

The TPM supports two power reduction modes of operation referred to as the idle mode and the power-down mode.

## FEATURES

- 80C51 based architecture
- Small package sizes
  - 28-pin Shrink Small Outline Package (SSOP)
  - 28-pin PLCC
- Low power consumption:
  - Normal operation: less than 11mA @ 5V, 12MHz
  - Idle mode
  - Power-down mode
- 2k × 8 ROM
- 64 × 8 RAM
- 16-bit auto reloadable counter/timer
- 5-channel 8-bit A/D converter
- 8-bit PWM output/timer
- 10-bit fixed-rate timer
- CMOS and TTL compatible

## PIN CONFIGURATION




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P3.4/A4 | 15 | P1.2/ADC2/D2 |
| 2 | P3.3/A3 | 16 | P1.3/ADC3/D3 |
| 3 | P3.2/A2/A10 | 17 | P1.4/ADC4/D4 |
| 4 | P3.1/A1/A9 | 18 | $AV_{SS}$ |
| 5 | P3.0/A0/A8 | 19 | $AV_{CC}$ |
| 6 | P0.2 | 20 | P1.5/$\overline{INT0}$/D5 |
| 7 | P0.1/OE | 21 | P1.6/$\overline{INT1}$/D6 |
| 8 | P0.0/ASEL | 22 | P1.7/T0/D7 |
| 9 | RST | 23 | P0.3 |
| 10 | X2 | 24 | P0.4/PWM OUT |
| 11 | X1 | 25 | P3.7/A7 |
| 12 | $V_{SS}$ | 26 | P3.6/A6 |
| 13 | P1.0/ADC0/D0 | 27 | P3.5/A5 |
| 14 | P1.1/ADC1/D1 | 28 | $V_{CC}$ |

SU00692A

## ORDERING INFORMATION

| ORDERING CODE | TEMPERATURE RANGE AND PACKAGE | DRAWING NUMBER |
|---|---|---|
| PTPM749 A | 0 to +70°C, Plastic Leaded Chip Carrier | SOT261-3 |
| PTPM749 DB | 0 to +70°C, Shrink Small Ouline Package | SOT341-1 |

For compatible pointing device, contact:

| COMPANY | CONTACT | TELEPHONE |
|---|---|---|
| Bokam Engineering | Ms. Jane Kamenster | (714)513-2200 |
| CTS Corporation | Mr. Dave Poole | (219)589-7169 |

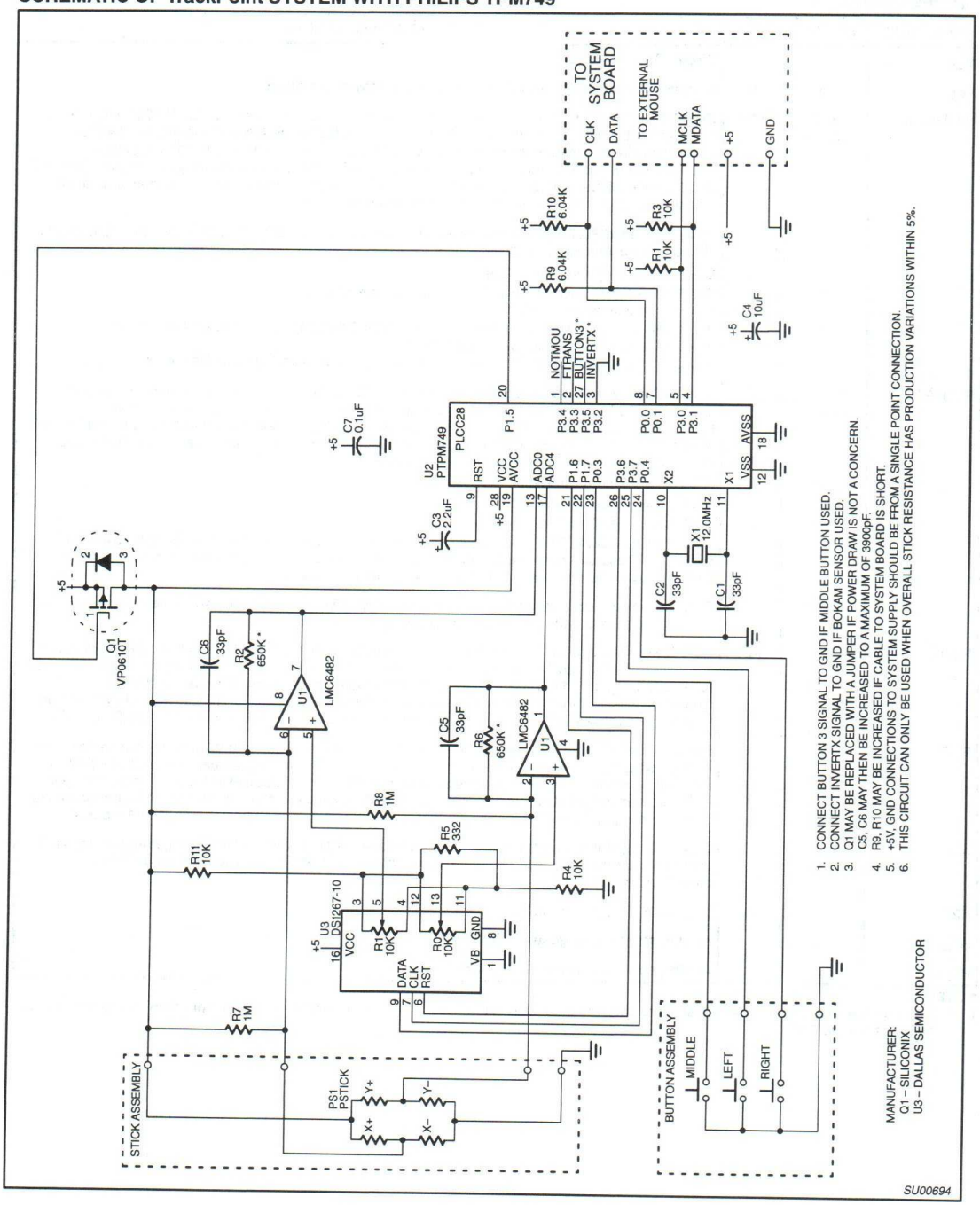
IBM is a registered trademark, and TrackPoint is a trademark of IBM Corporation.

## PIN DESCRIPTION

| MNEMONIC | PIN NO. | TYPE | NAME AND FUNCTION |
|---|---|---|---|
| $V_{SS}$ | 12 | I | **Circuit Ground Potential.** |
| $V_{CC}$ | 28 | I | **Supply voltage during normal, idle, and power-down operation.** |
| P0.0–P0.4 | 8–6<br>23, 24 | I/O | **Port 0:** Port 0 is a 5-bit bidirectional port. Port 0.0–P0.2 are open drain. Port 0.0–P0.2 pins that have 1s written to them float, and in that state can be used as high-impedance inputs. P0.3–P0.4 are bidirectional I/O port pins with internal pull-ups. These pins are driven low if the port register bit is written with a 0. The state of the pin can always be read from the port register by the program. Port 0.3 and 0.4 have internal pull-ups that function identically to port 3. Pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs.<br><br>While P0.0 anbd P0.1 differ from "standard TTL" characteristics, they are close enough for the pins to still be used as general-purpose I/O. |
| | 6 | I | **$V_{PP}$ (P0.2)** – Programming voltage input. |
| | 7 | I | **OE (P0.1)** – Input which specifies verify mode (output enable).<br>OE = 1 output enabled (verify mode). |
| | 8 | I | **ASEL (P0.0)** – Input which indicates which bits of the EPROM address are applied to port 3.<br>ASEL = 0 low address byte available on port 3.<br>ASEL = 1 high address byte available on port 3 (only the three least significant bits are used). |
| P1.0–P1.7 | 13–17,<br>20–22 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. P0.3–P0.4 pins are bidirectional I/O port pins with internal pull-ups. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also serves the special function features of the SC80C51 family as listed below: |
| | 20 | I | **$\overline{INT0}$ (P1.5):** External interrupt. |
| | 21 | I | **$\overline{INT1}$ (P1.6):** External interrupt. |
| | 22 | I | **T0 (P1.7):** Timer 0 external input. |
| | 13–17 | I | **ADC0 (P1.0)–ADC4 (P1.4):** Port 1 also functions as the inputs to the five channel multiplexed A/D converter. These pins can be used as outputs only if the A/D function has been disabled. These pins can be used as digital inputs while the A/D converter is enabled.<br><br>Port 1 serves to output the addressed EPROM contents in the verify mode and accepts as inputs the value to program into the selected address during the program mode. |
| P3.0–P3.7 | 5–1,<br>27–25 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also functions as the address input for the EPROM memory location to be programmed (or verified). The 11-bit address is multiplexed into this port as specified by P0.0/ASEL. |
| RST | 9 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on RESET using only an external capacitor to $V_{CC}$. After the device is reset, a 10-bit serial sequence, sent LSB first, applied to RESET, places the device in the programming state allowing programming address, data and $V_{PP}$ to be applied for programming or verification purposes. The RESET serial sequence must be synchronized with the X1 input. |
| X1 | 11 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. X1 also serves as the clock to strobe in a serial bit stream into RESET to place the device in the programming state. |
| X2 | 10 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |
| $AV_{CC}$ [1] | 19 | I | **Analog supply voltage and reference input.** |
| $AV_{SS}$ [1] | 18 | I | **Analog supply and reference ground.** |

**NOTE:**
1. $AV_{SS}$ (reference ground) must be connected to 0V (ground). $AV_{CC}$ (reference input) cannot differ from $V_{CC}$ by more than ±0.2V, and must be in the range 4.5V to 5.5V.

## SCHEMATIC OF TrackPoint SYSTEM WITH PHILIPS TPM749



## OSCILLATOR CHARACTERISTICS

X1 and X2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, X1 should be driven while X2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## IDLE MODE

The TPM includes the 80C51 power-down and idle mode features. In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals except the A/D and PWM stay active. The functions that continue to run while in the idle mode are Timer 0, Timer I, and the interrupts. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset. Upon powering-up the circuit, or exiting from idle mode, sufficient time must be allowed for stabilization of the internal analog reference voltages before an A/D conversion is started.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. The control bits for the reduced power modes are in the special function register PCON.

**Table 1. External Pin Status During Idle and Power-Down Modes**

| MODE | Port 0* | Port 1 | Port 2 |
|---|---|---|---|
| Idle | Data | Data | Data |
| Power-down | Data | Data | Data |

* Except for PWM output (P0.4).

## I/O Ports

The I/O pins provided by the TPM consist of port 0, port 1, and port 3.

### Port 0

Port 0 is a 5-bit bidirectional I/O port and includes alternate functions on some pins of this port. Pins P0.3 and P0.4 are provided with internal pullups while the remaining pins (P0.0, P0.1, and P0.2) have open drain output structures. The alternate function for port P0.4 is PWM output.

If the alternate function PWM is not being used, then this pin may be used as an I/O port.

### Port 1

Port 1 is an 8-bit bidirectional I/O port whose structure is identical to the 80C51, but also includes alternate input functions on all pins. The alternate pin functions for port 1 are:

P1.0-P1.4 - ADC0-ADC4 - A/D converter analog inputs
P1.5 $\overline{INT0}$ - external interrupt 0 input
P1.6 $\overline{INT1}$ - external interrupt 1 input
P1.7 - T0 - timer 0 external input

If the alternate functions $\overline{INT0}$, $\overline{INT1}$, or T0 are not being used, these pins may be used as standard I/O ports. It is necessary to connect $AV_{CC}$ and $AV_{SS}$ to $V_{CC}$ and $V_{SS}$, respectively, in order to use P1.5, P1.6, and P1.7 pins as standard I/O pins. When the A/D converter is enabled, the analog channel connected to the A/D may not be used as a digital input; however, the remaining analog inputs may be used as digital inputs. They may not be used as digital outputs. While the A/D is enabled, the analog inputs are floating.

### Port 3

Port 3 is an 8-bit bidirectional I/O port whose structure is identical to the 80C51. Note that the alternate functions associated with port 3 of the 80C51 have been moved to port 1 of the TPM (as applicable). See Figure 1 for port bit configurations.

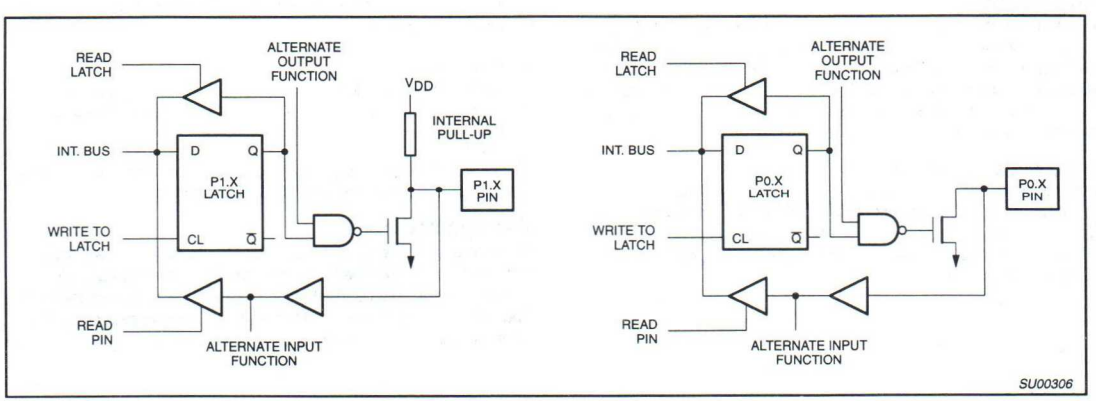
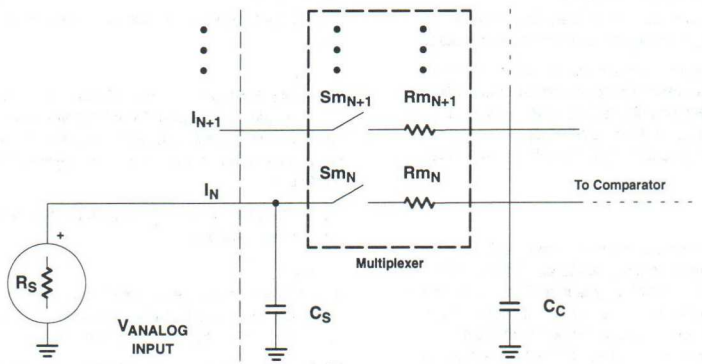



**Figure 1. Port Bit Latches and I/O Buffers**




Rm = 0.5 - 3 kΩ
CS + CC = 15pF maximum
RS = Recommended < 9.6 kΩ for 1 LSB @ 12MHz

**NOTE:**
Because the analog to digital converter has a sampled-data comparator, the input looks capacitive to a source. When a conversion is initiated, switch Sm closes for 8tcy (8µs @ 12MHz crystal frequency) during which time capacitance Cs + Cc is charged. It should be noted that the sampling causes the analog input to present a varying load to an analog source.

SU00199

**Figure 2. A/D Input: Equivalent Circuit**

## A/D CONVERTER PARAMETER DEFINITIONS

The following definitions are included to clarify some specifications given and do not represent a complete set of A/D parameter definitions.

### Absolute Accuracy Error

Absolute accuracy error of a given output is the difference between the theoretical analog input voltage to produce a given output and the actual analog input voltage required to produce the same code. Since the same output code is produced by a band of input voltages, the "required input voltage" is defined as the midpoint of the band of input voltage that will produce that code. Absolute accuracy error not specified with a code is the maximum over all codes.

### Nonlinearity

If a straight line is drawn between the end points of the actual converter characteristics such that zero offset and full scale errors are removed, then non-linearity is the maximum deviation of the code transitions of the actual characteristics from that of the straight line so constructed. This is also referred to as relative accuracy and also integral non-linearity.

### Differential Non-Linearity

Differential non-linearity is the maximum difference between the actual and ideal code widths of the converter. The code widths are the differences expressed in LSB between the code transition points, as the input voltage is varied through the range for the complete set of codes.

### Gain Error

Gain error is the deviation between the ideal and actual analog input voltage required to cause the final code transition to a full-scale output code after the offset error has been removed. This may sometimes be referred to as full scale error.

### Offset Error

Offset error is the difference between the actual input voltage that causes the first code transition and the ideal value to cause the first code transition. This ideal value is 1/2 LSB above $V_{ref-}$.

### Channel to Channel Matching

Channel to channel matching is the maximum difference between the corresponding code transitions of the actual characteristics taken from different channels under the same temperature, voltage and frequency conditions.

### Crosstalk

Crosstalk is the measured level of a signal at the output of the converter resulting from a signal applied to one deselected channel.

### Total Error

Maximum deviation of any step point from a line connecting the ideal first transition point to the ideal last transition point.

### Relative Accuracy

Relative accuracy error is the deviation of the ADC's actual code transition points from the ideal code transition points on a straight line which connects the ideal first code transition point and the final code transition point, after nullifying offset error and gain error. It is generally expressed in LSBs or in percent of FSR.

## ABSOLUTE MAXIMUM RATINGS[1, 3, 4]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage from $V_{CC}$ to $V_{SS}$ | –0.5 to +6.5 | V |
| Voltage from any pin to $V_{SS}$ (except $V_{PP}$) | –0.5 to $V_{CC}$ + 0.5 | V |
| Power dissipation | 1.0 | W |
| Voltage from $V_{PP}$ pin to $V_{SS}$ | –0.5 to + 13.0 | V |

**NOTES ON PAGE 3-1116.**

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, $AV_{CC}$ = 5V ±5, $AV_{SS}$ = 0V[4]
$V_{CC}$ = 5V ± 10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS[4] MIN | TYP[1] | MAX | UNIT |
|---|---|---|---|---|---|---|
| $I_{CC}$ | Supply current (see Figure 5) | | | | | |
| **Inputs** | | | | | | |
| $V_{IL}$ | Input low voltage | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IH}$ | Input high voltage, except X1, RST | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, X1, RST | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| | P0.2 | | | | | |
| $V_{IL1}$ | Input low voltage | | –0.5 | | $0.3V_{CC}$ | V |
| $V_{IH2}$ | Input high voltage | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| **Outputs** | | | | | | |
| $V_{OL}$ | Output low voltage, ports 1, 3, 0.3, and 0.4 (PWM disabled) | $I_{OL}$ = 1.6mA[2] | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0.2 | $I_{OL}$ = 3.2mA[2] | | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1, 3, 0.3, and 0.4 (PWM disabled) | $I_{OH}$ = –60µA, | 2.4 | | | V |
| | | $I_{OH}$ = –25µA | $0.75V_{CC}$ | | | V |
| | | $I_{OH}$ = –10µA | $0.9V_{CC}$ | | | V |
| | | $I_{OH}$ = –400µA | 2.4 | | | V |
| $V_{OH2}$ | Output high voltage, P0.4 (PWM enabled) | $I_{OH}$ = –40µA | $0.9V_{CC}$ | | | V |
| | Port 0.0 and 0.1 – Drivers | $I_{OL}$ = 3mA | | | | |
| $V_{OL2}$ | Output low voltage | (over $V_{CC}$ range) | | | 0.4 | V |
| | Driver, receiver combined: | | | | | |
| C | Capacitance | | | | 10 | pF |
| $I_{IL}$ | Logical 0 input current, ports 1, 3, 0.3, and 0.4 (PWM disabled)[11] | $V_{IN}$ = 0.45V | | | –50 | µA |
| $I_{TL}$ | Logical 1 to 0 transition current, ports 1, 3, 0.3 and 0.4[11] | $V_{IN}$ = 2V | | | –650 | µA |
| $I_{LI}$ | Input leakage current, port 0.0, 0.1 and 0.2 | 0.45 < $V_{IN}$ < $V_{CC}$ | | | ±10 | µA |
| $R_{RST}$ | Reset pull-down resistor | | 25 | | 175 | kΩ |
| $C_{IO}$ | Pin capacitance | Test freq = 1MHz, $T_{amb}$ = 25°C | | | 10 | pF |
| $I_{PD}$ | Power-down current[5] | $V_{CC}$ = 2 to 5.5V<br>$V_{CC}$ = 2 to 6.0V | | | 50 | µA |

**NOTES ON FOLLOWING PAGE.**

## DC ELECTRICAL CHARACTERISTICS (Continued)

$T_{amb}$ = 0°C to +70°C, $AV_{CC}$ = 5V ±5, $AV_{SS}$ = 0V[4]
$V_{CC}$ = 5V ± 10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS[4] MIN | TYP[1] | MAX | UNIT |
|---|---|---|---|---|---|---|
| **Analog Inputs** (A/D guaranteed only with quartz window covered.) | | | | | | |
| $AV_{CC}$ | Analog supply voltage[10] | $AV_{CC} = V_{CC}$±0.2V | 4.5 | | 5.5 | V |
| $AI_{CC}$ | Analog operating supply current | $AV_{CC}$ = 5.12V | | | 3[9] | mA |
| $AV_{IN}$ | Analog input voltage | | $AV_{SS}$–0.2 | | $AV_{CC}$+0.2 | V |
| $C_{IA}$ | Analog input capacitance | | | | 15 | pF |
| $t_{ADS}$ | Sampling time | | | | $8t_{CY}$ | s |
| $t_{ADC}$ | Conversion time | | | | $40t_{CY}$ | s |
| **Analog Inputs** (A/D guaranteed only with quartz window covered.) (Continued) | | | | | | |
| R | Resolution | | | | 8 | bits |
| $E_{RA}$ | Relative accuracy | | | | ±1 | LSB |
| $OS_e$ | Zero scale offset | | | | ±1 | LSB |
| $G_e$ | Full scale gain error | | | | 0.4 | % |
| $M_{CTC}$ | Channel to channel matching | | | | ±1 | LSB |
| $C_t$ | Crosstalk | 0–100kHz | | | –60 | dB |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 10mA
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 67mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
3. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
4. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.
5. Power-down $I_{CC}$ is measured with all output pins disconnected; port 0 = $V_{CC}$; X2, X1 n.c.; RST = $V_{SS}$.
6. $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL} = V_{SS} + 0.5V$, $V_{IH} = V_{CC} - 0.5V$; X2 n.c.; RST = port 0 = $V_{CC}$. $I_{CC}$ will be slightly higher if a crystal oscillator is used.
7. Idle $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL} = V_{SS} + 0.5V$, $V_{IH} = V_{CC} - 0.5V$; X2 n.c.; port 0 = $V_{CC}$; RST = $V_{SS}$.
8. Load capacitance for ports = 80pF.
9. The resistor ladder network is not disconnected in the power down or idle modes. Thus, to conserve power, the user may remove $AV_{CC}$.
10. If the A/D function is not required, or if the A/D function is only needed periodically, $AV_{CC}$ may be removed without affecting the operation of the digital circuitry. Contents of ADCON and ADAT are not guaranteed to be valid. If $AV_{CC}$ is removed, the A/D inputs must be lowered to less than 0.5V. Digital inputs on P1.0–P1.4 will not function normally.
11. These parameters do not apply to P1.0–P1.4 if the A/D function is enabled.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[4, 8]

| SYMBOL | PARAMETER | 12MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | Oscillator frequency: | | | 3.5 | 12 | MHz |
| **External Clock (Figure 3)** | | | | | | |
| $t_{CHCX}$ | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | Fall time | | 20 | | 20 | ns |

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

C – Clock
D – Input data
H – Logic level high
L – Logic level low
Q – Output data
T – Time
V – Valid
X – No longer a valid logic level
Z – Float

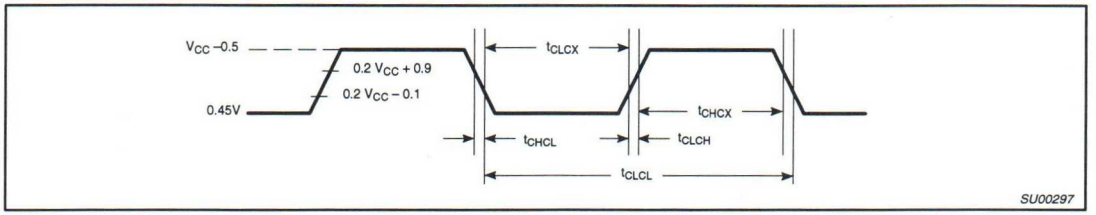



**Figure 3. External Clock Drive**

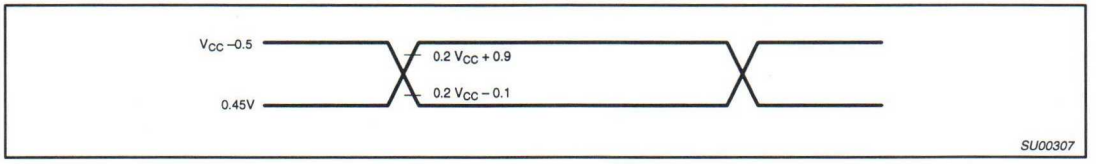



**Figure 4. AC Testing Input/Output**

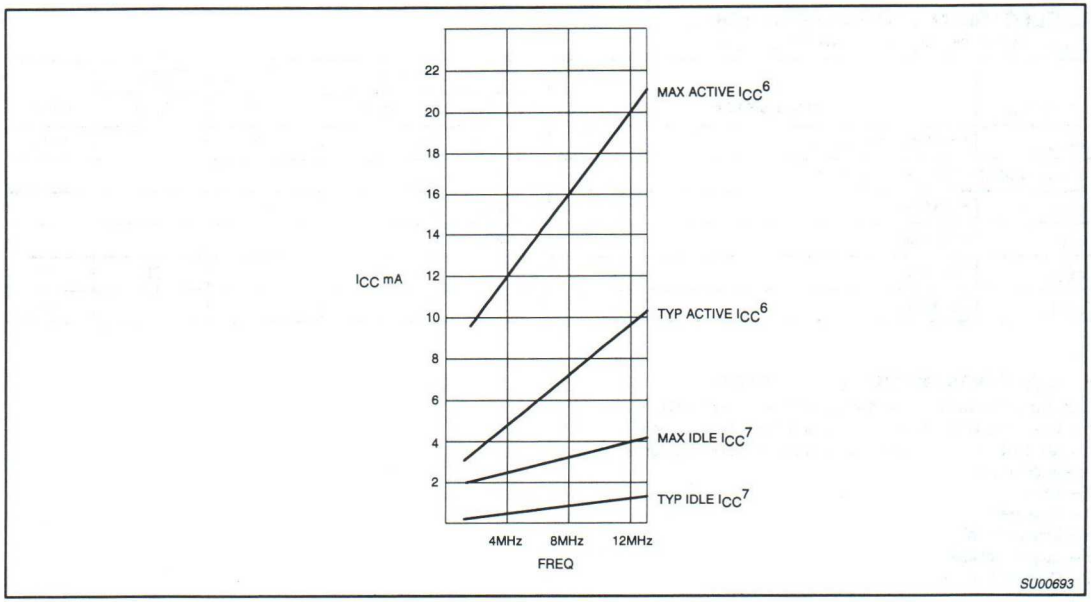



**Figure 5. $I_{CC}$ vs. FREQ**
**Maximum $I_{CC}$ values taken at $V_{CC}$ = 5.5V and worst case temperature.**
**Typical $I_{CC}$ values taken at $V_{CC}$ = 5.0V and 25°C.**
**Notes 6 and 7 refer to AC Electrical Characteristics.**

# CMOS single-chip 8-bit microcontrollers 83C750/87C750

## DESCRIPTION

The Philips 8XC750 offers the advantages of the 80C51 architecture in a small package and at low cost.

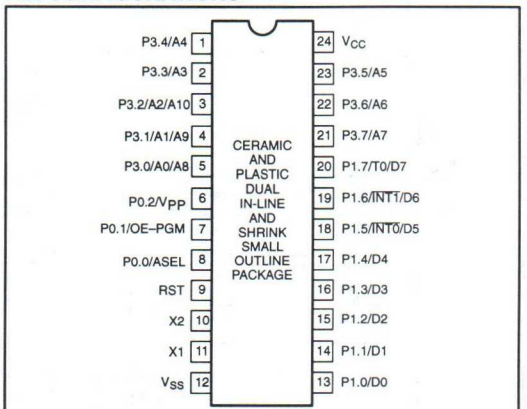
The 8XC750 Microcontroller is fabricated with Philips high-density CMOS technology. Philips epitaxial substrate minimizes CMOS latch-up sensitivity.

The 87C750 contains a 1k × 8 EPROM, a 64 × 8 RAM, 19 I/O lines, a 16-bit auto-reload counter/timer, a five-source, fixed-priority level interrupt structure and an on-chip oscillator.

## FEATURES

- 80C51 based architecture
- Wide oscillator frequency range—up to 40MHz
- Small package sizes
  - 24-pin DIP (300 mil "skinny DIP")
  - 24-pin Shrink Small Outline Package
  - 28-pin PLCC
- 87C750 available in erasable quartz lid or one-time programmable plastic packages
- Low power consumption:
  - Normal operation: less than 11mA @ 5V, 12MHz
  - Idle mode
  - Power-down mode
- 1k × 8 EPROM (87C750)
- 64 × 8 RAM
- 16-bit auto reloadable counter/timer
- Boolean processor
- CMOS and TTL compatible
- Well suited for logic replacement, consumer and industrial applications
- LED drive outputs

## PIN CONFIGURATIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P3.4/A4 | 15 | P1.0/D0 |
| 2 | P3.3/A3 | 16 | P1.1/D1 |
| 3 | P3.2/A2/A10 | 17 | P1.2/D2 |
| 4 | P3.1/A1/A9 | 18 | P1.3/D3 |
| 5 | NC* | 19 | P1.4/D4 |
| 6 | P3.0/A0/A8 | 20 | P1.5/INT0/D5 |
| 7 | P0.2/VPP | 21 | NC* |
| 8 | P0.1/OE-PGM | 22 | NC* |
| 9 | P0.0/ASEL | 23 | P1.6/INT1/D6 |
| 10 | NC* | 24 | P1.7/T0/D7 |
| 11 | RST | 25 | P3.7/A7 |
| 12 | X2 | 26 | P3.6/A6 |
| 13 | X1 | 27 | P3.5/A5 |
| 14 | VSS | 28 | VCC |

* DO NOT CONNECT

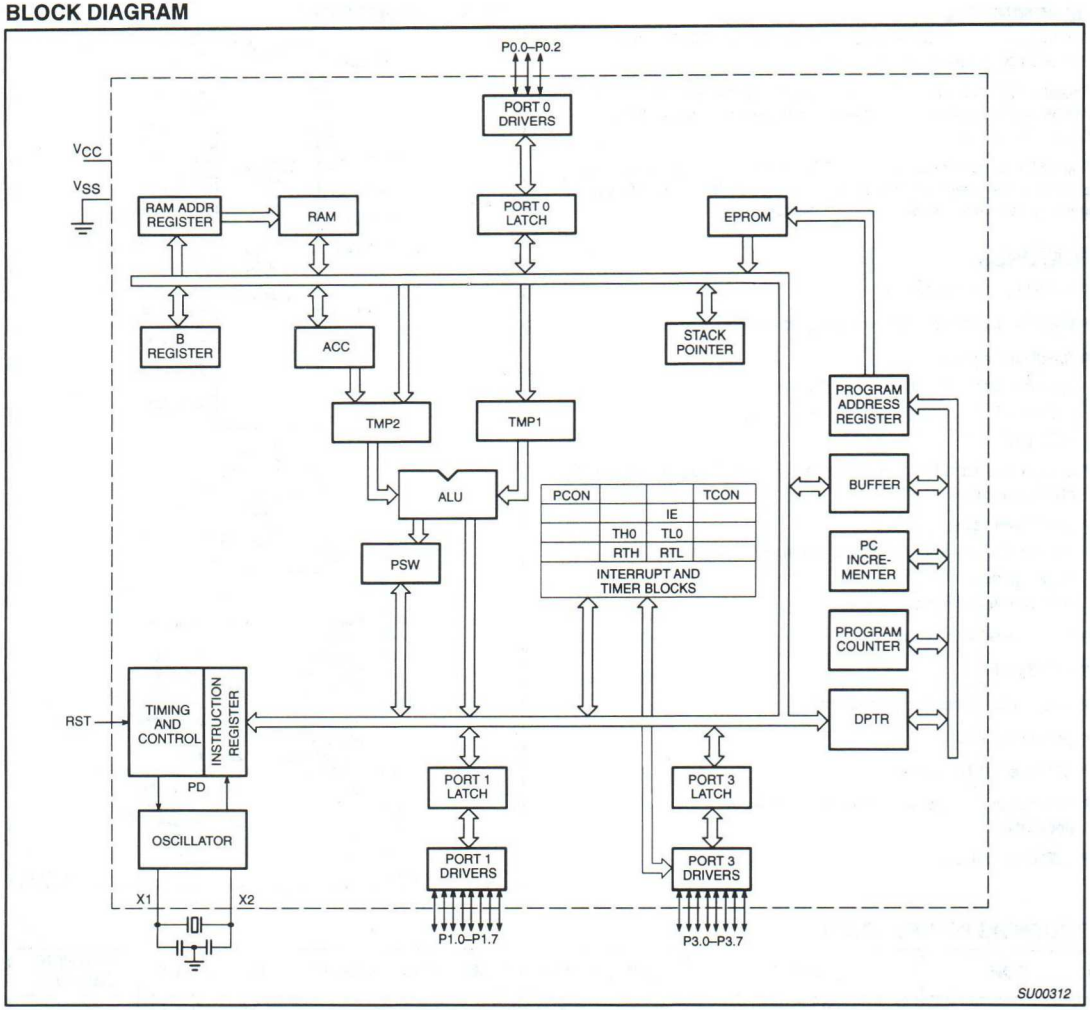
SU00295A

## ORDERING INFORMATION

| ROM | EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQUENCY | DRAWING NUMBER |
|---|---|---|---|---|---|
| | P87C750EBF FA | UV | 0 to +70, Ceramic Dual In-line Package | 3.5 to 16MHz | 0586B |
| | P87C750EFF FA | UV | –40 to +85, Ceramic Dual In-line Package | 3.5 to 16MHz | 0586B |
| P83C750EBP N | P87C750EBP N | OTP | 0 to +70, Plastic Dual In-line Package | 3.5 to 16MHz | SOT222-1 |
| P83C750EFP N | P87C750EFP N | OTP | –40 to +85, Plastic Dual In-line Package | 3.5 to 16MHz | SOT222-1 |
| P83C750EBA A | P87C750EBA A | OTP | 0 to +70, Plastic Lead Chip Carrier | 3.5 to 16MHz | SOT261-3 |
| P83C750EFA A | P87C750EFA A | OTP | –40 to +85, Plastic Lead Chip Carrier | 3.5 to 16MHz | SOT261-3 |
| P83C750EBD DB | P87C750EBD DB | OTP | 0 to +70, Shrink Small Outline Package | 3.5 to 16MHz | SOT340-1 |
| P83C750PBP N | P87C750PBP N | OTP | 0 to +70, Plastic Dual In-line Package | 3.5 to 40MHz | SOT222-1 |
| P83C750PFP N | P87C750PFP N | OTP | –40 to +85, Plastic Dual In-line Package | 3.5 to 40MHz | SOT222-1 |
| P83C750PBA A | P87C750PBA A | OTP | 0 to +70, Plastic Lead Chip Carrier | 3.5 to 40MHz | SOT261-3 |
| P83C750PFA A | P87C750PFA A | OTP | –40 to +85, Plastic Lead Chip Carrier | 3.5 to 40MHz | SOT261-3 |
| | P87C750PBF FA | UV | 0 to +70, Ceramic Dual In-line Package | 3.5 to 40MHz | 0586B |
| | P87C750PFF FA | UV | –40 to +85, Ceramic Dual In-line Package | 3.5 to 40MHz | 0586B |

**NOTE:**
1. OTP = One Time Programmable EPROM. UV = UV Erasable EPROM.

## BLOCK DIAGRAM



## PIN DESCRIPTIONS

| MNEMONIC | PIN NO. DIP/ SSOP | PIN NO. LCC | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|
| $V_{SS}$ | 12 | 14 | I | **Circuit Ground Potential** |
| $V_{CC}$ | 24 | 28 | I | **Supply voltage during normal, idle, and power-down operation.** |
| P0.0-P0.2 | 8-6 | 9-7 | I/O | **Port 0:** Port 0 is a 3-bit open-drain, bidirectional port. Port 0 pins that have 1s written to them float, and in that state can be used as high-impedance inputs. These pins are driven low if the port register bit is written with a 0. The state of the pin can always be read from the port register by the program.<br>P0.0, P0.1, and P0.2 are open drain bidirectional I/O pins with the electrical characteristics listed in the tables that follow. While these differ from "standard TTL" characteristics, they are close enough for the pins to still be used as general-purpose I/O. Port 0 also provides alternate functions for programming the EPROM memory as follows: |
| | 6 | 7 | N/A | **$V_{PP}$ (P0.2)** – Programming voltage input. (See Note 1.) |
| | 7 | 8 | I | **OE/PGM (P0.1)** – Input which specifies verify mode (output enable) or the program mode.<br>OE/PGM = 1 output enabled (verify mode).<br>OE/PGM = 0 program mode. |
| | 8 | 9 | I | **ASEL (P0.0)** – Input which indicates which bits of the EPROM address are applied to port 3.<br>ASEL = 0 low address byte available on port 3.<br>ASEL = 1 high address byte available on port 3 (only the three least significant bits are used). |
| P1.0-P1.7 | 13-20 | 15-20, 23, 24 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 serves to output the addressed EPROM contents in the verify mode and accepts as inputs the value to program into the selected address during the program mode. Port 1 also serves the special function features of the 80C51 family as listed below: |
| | 18 | 20 | I | **$\overline{INT0}$ (P1.5):** External interrupt. |
| | 19 | 23 | I | **$\overline{INT1}$ (P1.6):** External interrupt. |
| | 20 | 24 | I | **T0 (P1.7):** Timer 0 external input. |
| P3.0-P3.7 | 5-1, 23-21 | 6, 4-1, 27-25 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also functions as the address input for the EPROM memory location to be programmed (or verified). The 10-bit address is multiplexed into this port as specified by P0.0/ASEL. |
| RST | 9 | 11 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on RESET using only an external capacitor to $V_{CC}$. After the device is reset, a 10-bit serial sequence, sent LSB first, applied to RESET, places the device in the programming state allowing programming address, data and $V_{PP}$ to be applied for programming or verification purposes. The RESET serial sequence must be synchronized with the X1 input. |
| X1 | 11 | 13 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. X1 also serves as the clock to strobe in a serial bit stream into RESET to place the device in the programming state. |
| X2 | 10 | 12 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTE:**
1. When P0.2 is at or close to 0 Volt, it may affect the internal ROM operation. We recommend that P0.2 be tied to $V_{CC}$ via a small pull-up (e.g., 2kΩ).

## OSCILLATOR CHARACTERISTICS

X1 and X2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, X1 should be driven while X2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-up reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-up, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up.

## IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. the control bits for the reduced power modes are in the special function register PCON.

**Table 1. External Pin Status During Idle and Power-Down Modes**

| MODE | Port 0 | Port 1 | Port 2 |
|---|---|---|---|
| Idle | Data | Data | Data |
| Power-down | Data | Data | Data |

## DIFFERENCES BETWEEN THE 8XC750 AND THE 80C51

### Program Memory

On the 8XC750, program memory is 1024 bytes long and is not externally expandable, so the 80C51 instructions MOVX, LJMP, and LCALL are not implemented. The only fixed locations in program memory are the addresses at which execution is taken up in response to reset and interrupts, which are as follows:

| Event | Program Memory Address |
|---|---|
| Reset | 000 |
| External $\overline{INT0}$ | 003 |
| Counter/timer 0 | 00B |
| External $\overline{INT1}$ | 013 |

### Counter/Timer Subsystem

**Timer/Counter**

The 8XC750 has one timers: a 16-bit timer/counter. The 16-bit timer/counter's operation is similar to mode 2 operation on the 80C51, but is extended to 16 bits. The timer/counter is clocked by either 1/12 the oscillator frequency or by transitions on the T0 pin. The C/T pin in special function register TCON selects between these two modes. When the TCON TR bit is set, the timer/counter is enabled. Register pair TH and TL are incremented by the clock source. When the register pair overflows, the register pair is reloaded with the values in registers RTH and RTL. The value in the reload registers is left unchanged. See the 83C750 counter/timer block diagram in Figure 1. The TF bit in special function register TCON is set on counter overflow and, if the interrupt is enabled, will generate an interrupt.

**TCON Register**

MSB LSB

| GATE | C/T | TF | TR | IE0 | IT0 | IE1 | IT1 |
|---|---|---|---|---|---|---|---|

GATE 1 – Timer/counter is enabled only when INT0 pin is high, and TR is 1.
0 – Timer/counter is enabled when TR is 1.
C/T 1 – Counter/timer operation from T0 pin.
0 – Timer operation from internal clock.
TF 1 – Set on overflow of TH.
0 – Cleared when processor vectors to interrupt routine and by reset.
TR 1 – Timer/counter enabled.
0 – Timer/counter disabled.
IE0 1 – Edge detected in $\overline{INT0}$.
IT0 1 – $\overline{INT0}$ is edge triggered.
0 – $\overline{INT0}$ is level sensitive.
IE1 1 – Edge detected on $\overline{INT1}$.
IT1 1 – $\overline{INT1}$ is edge triggered.
0 – $\overline{INT1}$ is level sensitive.

These flags are functionally identical to the corresponding 80C51 flags, except that there is only one timer on the 83C750 and the flags are therefore combined into one register.

Note that the positions of the IE0/IT0 and IE1/IT1 bits are transposed from the positions used in the standard 80C51 TCON register.

### Interrupt Subsystem – Fixed Priority

The IP register and the 2-level interrupt system of the 80C51 are eliminated. Simultaneous interrupt conditions are resolved by a single-level, fixed priority as follows:

Highest priority: Pin $\overline{INT0}$
Counter/timer flag 0
Pin $\overline{INT1}$

### Special Function Register Addresses

Special function registers for the 8XC750 are identical to those of the 80C51, except for the changes listed below:

80C51 special function registers not present in the 8XC750 are TMOD (89), P2 (A0) and IP (B8). The 80C51 registers TH1 and TL1 are replaced with the 87C750 registers RTH and RTL respectively (refer to Table 2).

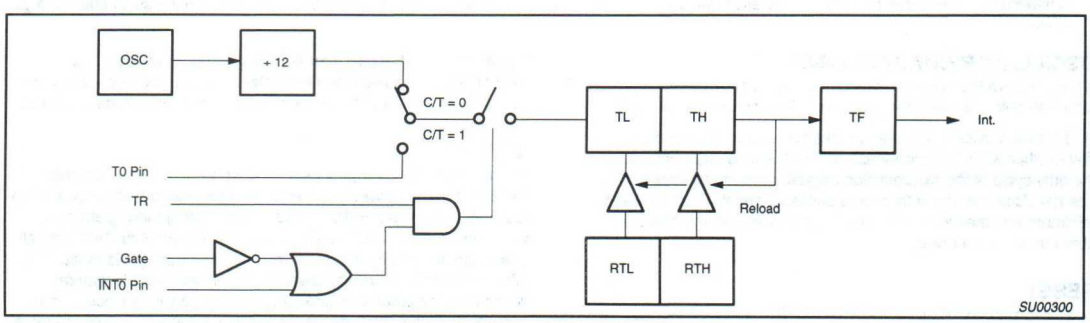



**Figure 1. 83C751 Counter/Timer Block Diagram**

**Table 2. 87C750 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes) | | | | | | | | | | |
| DPH | High byte | 83H | | | | | | | | | 00H |
| DPL | Low byte | 82H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | – | – | – | – | EX1 | ET0 | EX0 | 00H |
| | | | | | | | | 82 | 81 | 80 | |
| P0*# | Port 0 | 80H | – | – | – | – | – | – | – | – | xxxxx111B |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | T0 | $\overline{INT1}$ | $\overline{INT0}$ | – | – | – | – | – | FFH |
| P3* | Port 3 | B0H | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | FFH |
| PCON# | Power control | 87H | – | – | – | – | – | – | PD | IDL | xxxxxx00B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON*# | Timer/counter control | 88H | GATE | C/T | TF | TR | IE0 | IT0 | IE1 | IT1 | 00H |
| TL# | Timer low byte | 8AH | | | | | | | | | 00H |
| TH# | Timer high byte | 8CH | | | | | | | | | 00H |
| RTL# | Timer low reload | 8BH | | | | | | | | | 00H |
| RTH# | Timer high reload | 8DH | | | | | | | | | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

## ABSOLUTE MAXIMUM RATINGS[1, 2]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage from $V_{CC}$ to $V_{SS}$ | –0.5 to +6.5 | V |
| Voltage from any pin to $V_{SS}$ (except $V_{PP}$) | –0.5 to $V_{CC}$ + 0.5 | V |
| Power dissipation | 1.0 | W |
| Voltage on $V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Maximum $I_{OL}$ per I/O pin | 10 | mA |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1]

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $V_{IL}$ | Input low voltage | | –0.5 | $0.2V_{DD}$–0.1 | V |
| $V_{IH}$ | Input high voltage, except X1, RST | | $0.2V_{CC}$+0.9 | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, X1, RST | | $0.7V_{CC}$ | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1 and 3 | $I_{OL}$ = 1.6mA[2] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0 | $I_{OL}$ = 3.2mA[2] | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1 and 3 | $I_{OH}$ = –60µA | 2.4 | | V |
| | | $I_{OH}$ = –25µA | $0.75V_{CC}$ | | V |
| | | $I_{OH}$ = –10µA | $0.9V_{CC}$ | | V |
| C | Capacitance | | | 10 | pF |
| $I_{IL}$ | Logical 0 input current, ports 1 and 3 | $V_{IN}$ = 0.45V | | –50 | µA |
| $I_{TL}$ | Logical 1 to 0 transition current, ports 1 and 3[3] | $V_{IN}$ = 2V (0 to +70°C) | | –650 | µA |
| | | $V_{IN}$ = 2V (–40 to +85°C) | | –750 | µA |
| $I_{LI}$ | Input leakage current, port 0 | 0.45 < $V_{IN}$ < $V_{CC}$ | | ±10 | µA |
| $R_{RST}$ | Internal pull-down resistor | | 25 | 175 | kΩ |
| $C_{IO}$ | Pin capacitance | Test freq = 1MHz, $T_{amb}$ = 25°C | | 10 | pF |
| $I_{PD}$ | Power-down current[4] | $V_{CC}$ = 2 to $V_{CC}$ max | | 50 | µA |
| $V_{PP}$ | $V_{PP}$ program voltage | $V_{SS}$ = 0V<br>$V_{CC}$ = 5V±10%<br>$T_{amb}$ = 21°C to 27°C | 12.5 | 13.0 | V |
| $I_{PP}$ | Program current | $V_{PP}$ = 13.0V | | 50 | mA |
| $I_{CC}$ | Supply current (see Figure 3)[5, 6] | | | | |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.
2. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 10mA (NOTE: This is 85°C spec.)
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 67mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
3. Pins of ports 1 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
4. Power-down $I_{CC}$ is measured with all output pins disconnected; port 0 = $V_{CC}$; X2, X1 n.c.; RST = $V_{SS}$.
5. Active $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; RST = port 0 = $V_{CC}$. $I_{CC}$ will be slightly higher if a crystal oscillator is used.
6. Idle $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; port 0 = $V_{CC}$; RST = $V_{SS}$.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[1, 2]

| SYMBOL | PARAMETER | VARIABLE CLOCK | | | | UNIT |
|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | Oscillator frequency: | 3.5 | 16 | 3.5 | 40 | MHz |
| **External Clock (Figure 2)** | | | | | | |
| $t_{CHCX}$ | High time | 20 | | 10 | | ns |
| $t_{CLCX}$ | Low time | 20 | | 10 | | ns |
| $t_{CLCH}$ | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | Fall time | | 20 | | 20 | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.
2. Load capacitance for ports = 80pF.

## EXPLANATION OF THE AC SYMBOLS

In defining the clock waveform, care must be taken not to exceed the MIN or MAX limits of the AC electrical characteristics table. Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

C – Clock
D – Input data
H – Logic level high
L – Logic level low
Q – Output data
T – Time
V – Valid
X – No longer a valid logic level
Z – Float

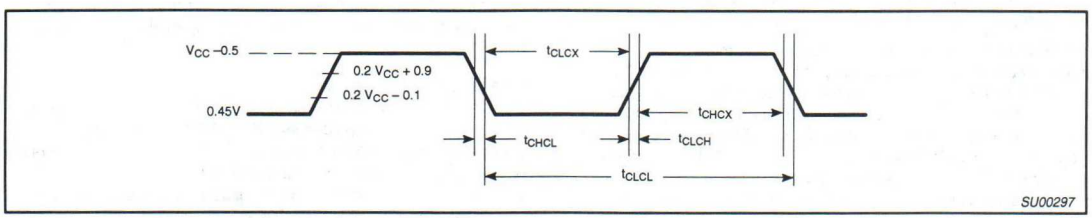



**Figure 2. External Clock Drive**

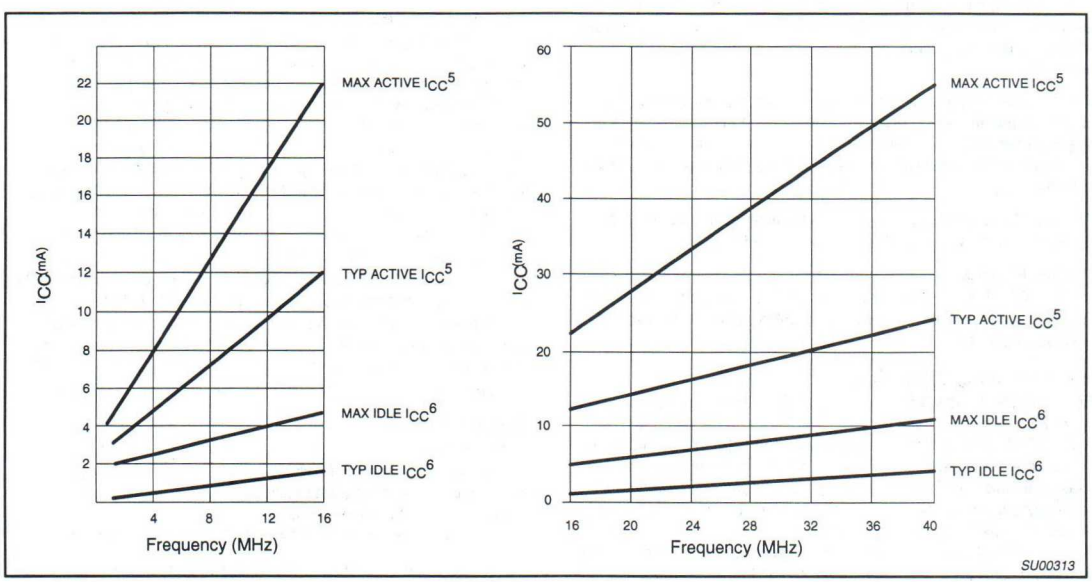



**Figure 3. $I_{CC}$ vs. Frequency**

Maximum $I_{CC}$ values taken at $V_{CC}$ max and worst case temperature.
Typical $I_{CC}$ values taken at $V_{CC}$ = 5.0V and 25°C.
Notes 5 and 6 refer to DC Electrical Characteristics.

## ROM CODE SUBMISSION

When submitting ROM code for the 80C750, the following must be specified:
1. 1k byte user ROM data

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 03FFH | DATA | 7:0 | User ROM Data |

## 87C750 PROGRAMMING CONSIDERATIONS

### EPROM Characteristics

The 87C750 is programmed by using a modified Quick-Pulse Programming algorithm similar to that used for devices such as the 87C451 and 87C51. It differs from these devices in that a serial data stream is used to place the 87C750 in the programming mode.

Figure 4 shows a block diagram of the programming configuration for the 87C750. Port pin P0.2 is used as the programming voltage supply input ($V_{PP}$ signal). Port pin P0.1 is used as the program (PGM/) signal. This pin is used for the 25 programming pulses.

Port 3 is used as the address input for the byte to be programmed and accepts both the high and low components of the eleven bit address. Multiplexing of these address components is performed using the ASEL input. The user should drive the ASEL input high and then drive port 3 with the high order bits of the address. ASEL should remain high for at least 13 clock cycles. ASEL may then be driven low which latches the high order bits of the address internally. the high address should remain on port 3 for at least two clock cycles after ASEL is driven low. Port 3 may then be driven with the low byte of the address. The low address will be internally stable 13 clock cycles later. The address will remain stable provided that the low byte placed on port 3 is held stable and ASEL is kept low. **Note:** ASEL needs to be pulsed high only to change the high byte of the address.

Port 1 is used as a bidirectional data bus during programming and verify operations. During programming mode, it accepts the byte to be programmed. During verify mode, it provides the contents of the EPROM location specified by the address which has been supplied to Port 3.

The XTAL1 pin is the oscillator input and receives the master system clock. This clock should be between 1.2 and 6MHz.

The RESET pin is used to accept the serial data stream that places the 87C750 into various programming modes. This pattern consists of a 10-bit code with the LSB sent first. Each bit is synchronized to the clock input, X1.

### Programming Operation

Figures 5 and 6 show the timing diagrams for the program/verify cycle. RESET should initially be held high for at least two machine cycles. P0.1 (PGM/) and P0.2 ($V_{PP}$) will be at $V_{OH}$ as a result of the RESET operation. At this point, these pins function as normal quasi-bidirectional I/O ports and the programming equipment may pull these lines low. However, prior to sending the 10-bit code on the RESET pin, the programming equipment should drive these pins high ($V_{IH}$). The RESET pin may now be used as the serial data input for the data stream which places the 87C750 in the programming mode. Data bits are sampled during the clock high time and thus should only change during the time that the clock is low. Following transmission of the last data bit, the RESET pin should be held low.

Next the address information for the location to be programmed is placed on port 3 and ASEL is used to perform the address multiplexing, as previously described. At this time, port 1 functions as an output.

A high voltage $V_{PP}$ level is then applied to the $V_{PP}$ input (P0.2). (This sets Port 1 as an input port). The data to be programmed into the EPROM array is then placed on Port 1. This is followed by a series of programming pulses applied to the PGM/ pin (P0.1). These pulses are created by driving P0.1 low and then high. This pulse is repeated until a total of 25 programming pulses have occurred. At the conclusion of the last pulse, the PGM/ signal should remain high.

The $V_{PP}$ signal may now be driven to the $V_{OH}$ level, placing the 87C750 in the verify mode. (Port 1 is now used as an output port). After four machine cycles (48 clock periods), the contents of the addressed location in the EPROM array will appear on Port 1.

The next programming cycle may now be initiated by placing the address information at the inputs of the multiplexed buffers, driving the $V_{PP}$ pin to the $V_{PP}$ voltage level, providing the byte to be programmed to Port1 and issuing the 26 programming pulses on the PGM/ pin, bringing $V_{PP}$ back down to the $V_C$ level and verifying the byte.

### Programming Modes

The 87C750 has four programming features incorporated within its EPROM array. These include the USER EPROM for storage of the application's code, a 16-byte encryption key array and two security bits. Programming and verification of these four elements are selected by a combination of the serial data stream applied to the RESET pin and the voltage levels applied to port pins P0.1 and P0.2. The various combinations are shown in Table 3.

### Encryption Key Table

The 87C750 includes a 16-byte EPROM array that is programmable by the end user. The contents of this array can then be used to encrypt the program memory contents during a program memory verify operation. When a program memory verify operation is performed, the contents of the program memory location is XNOR'ed with one of the bytes in the 16-byte encryption table. The resulting data pattern is then provided to port 1 as the verify data. The encryption mechanism can be disable, in essence, by leaving the bytes in the encryption table in their erased state (FFH) since the XNOR product of a bit with a logical one will result in the original bit. The encryption bytes are mapped with the code memory in 16-byte groups. the first byte in code memory will be encrypted with the first byte in the encryption table; the second byte in code memory will be encrypted with the second byte in the encryption table and so forth up to and including the 16the byte. The encryption repeats in 16-byte groups; the 17th byte in the code memory will be encrypted with the first byte in the encryption table, and so forth.

### Security Bits

Two security bits, security bit 1 and security bit 2, are provided to limit access to the USER EPROM and encryption key arrays. Security bit 1 is the program inhibit bit, and once programmed performs the following functions:

1. Additional programming of the USER EPROM is inhibited.
2. Additional programming of the encryption key is inhibited.
3. Verification of the encryption key is inhibited.
4. Verification of the USER EPROM and the security bit levels may still be performed.

(If the encryption key array is being used, this security bit should be programmed by the user to prevent unauthorized parties from reprogramming the encryption key to all logical zero bits. Such programming would provide data during a verify cycle that is the logical complement of the USER EPROM contents).

Security bit 2, the verify inhibit bit, prevents verification of both the USER EPROM array and the encryption key arrays. The security bit levels may still be verified.

### Programming and Verifying Security Bits

Security bits are programmed employing the same techniques used to program the USER EPROM and KEY arrays using serial data streams and logic levels on port pins indicated in Table 3. When programming either security bit, it is not necessary to provide address or data information to the 87C750 on ports 1 and 3.

Verification occurs in a similar manner using the RESET serial stream shown in Table 3. Port 3 is not required to be driven and the results of the verify operation will appear on ports 1.6 and 1.7.

Ports 1.7 contains the security bit 1 data and is a logical one if programmed and a logical zero if erased. Likewise, P1.6 contains the security bit 2 data and is a logical one if programmed and a logical zero if erased.

### Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Flourless part number 2345-5 or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-s/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

### Table 3. Implementing Program/Verify Modes

| OPERATION | SERIAL CODE | P0.1 (PGM/) | P0.2 ($V_{PP}$) |
|---|---|---|---|
| Program user EPROM | 296H | –[1] | $V_{PP}$ |
| Verify user EPROM | 296H | $V_{IH}$ | $V_{IH}$ |
| Program key EPROM | 292H | –[1] | $V_{PP}$ |
| Verify key EPROM | 292H | $V_{IH}$ | $V_{IH}$ |
| Program security bit 1 | 29AH | –[1] | $V_{PP}$ |
| Program security bit 2 | 298H | –[1] | $V_{PP}$ |
| Verify security bits | 29AH | $V_{IH}$ | $V_{IH}$ |

**NOTE:**
1. Pulsed from $V_{IH}$ to $V_{IL}$ and returned to $V_{IH}$.

### EPROM PROGRAMMING AND VERIFICATION

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $1/t_{CLCL}$ | Oscillator/clock frequency | 1.2 | 6 | MHz |
| $t_{AVGL}$[1] | Address setup to P0.1 (PROG–) low | 10µs + 24$t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after P0.1 (PROG–) high | 48$t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to P0.1 (PROG–) low | 38$t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to P0.1 (PROG–) low | 38$t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after P0.1 (PROG–) high | 36$t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to P0.1 (PROG–) low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after P0.1 (PROG–) | 10 | | µs |
| $t_{GLGH}$ | P0.1 (PROG–) width | 90 | 110 | µs |
| $t_{AVQV}$[2] | $V_{PP}$ low ($V_{CC}$) to data valid | | 48$t_{CLCL}$ | |
| $t_{GHGL}$ | P0.1 (PROG–) high to P0.1 (PROG–) low | 10 | | µs |
| $t_{SYNL}$ | P0.0 (sync pulse) low | 4$t_{CLCL}$ | | |
| $t_{SYNH}$ | P0.0 (sync pulse) high | 8$t_{CLCL}$ | | |
| $t_{MASEL}$ | ASEL high time | 13$t_{CLCL}$ | | |
| $t_{MAHLD}$ | Address hold time | 2$t_{CLCL}$ | | |
| $t_{HASET}$ | Address setup to ASEL | 13$t_{CLCL}$ | | |
| $t_{ADSTA}$ | Low address to valid data | | 48$t_{CLCL}$ | |

**NOTES:**
1. Address should be valid at least 24$t_{CLCL}$ before the rising edge of P0.2 ($V_{PP}$).
2. For a pure verify mode, i.e., no program mode in between, $t_{AVQV}$ is 14$t_{CLCL}$ maximum.

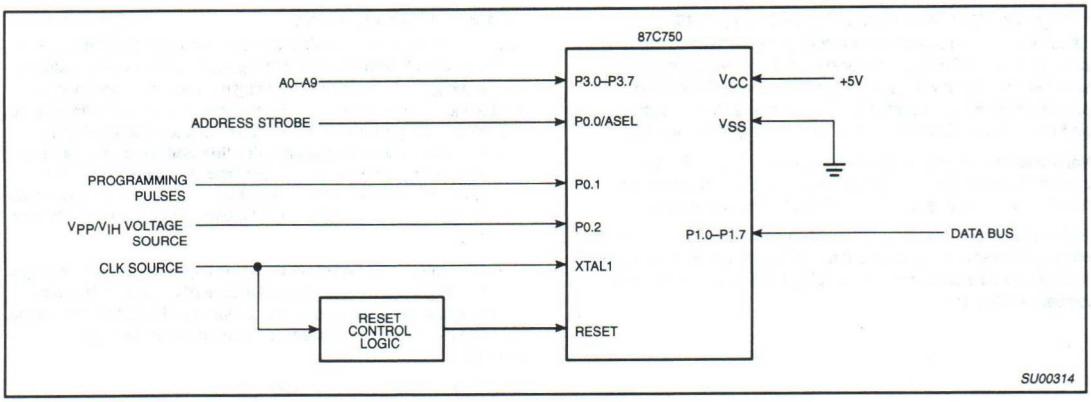



**Figure 4. Programming Configuration**

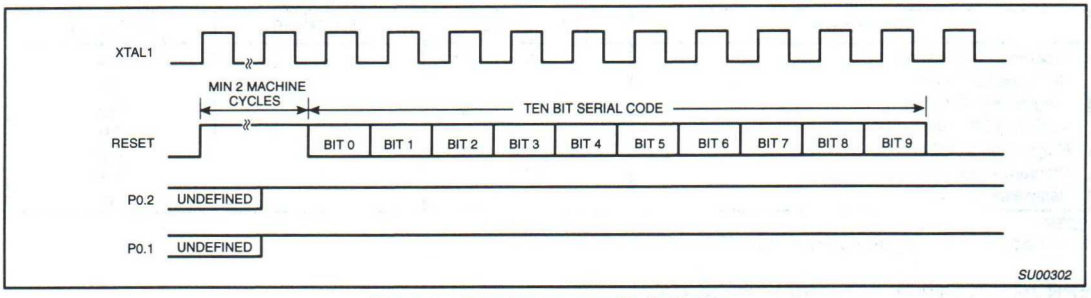



**Figure 5. Entry into Program/Verify Modes**

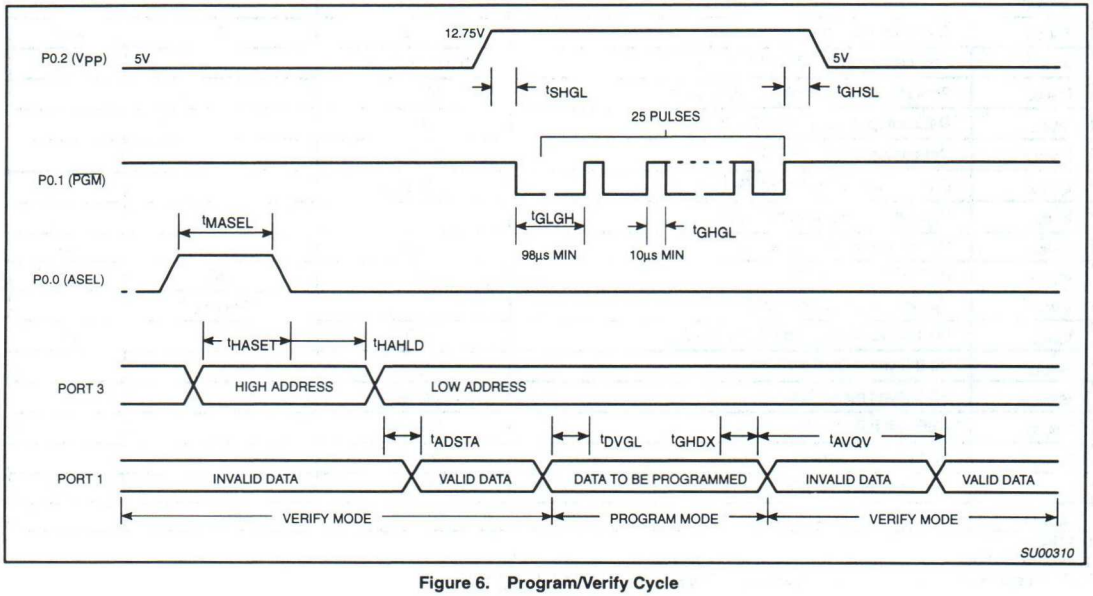



**Figure 6. Program/Verify Cycle**

# CMOS single-chip 8-bit microcontrollers 83C751/87C751

## DESCRIPTION

The Philips 83C751/87C751 offers the advantages of the 80C51 architecture in a small package and at low cost.

The 8XC751 Microcontroller is fabricated with Philips high-density CMOS technology. Philips epitaxial substrate minimizes CMOS latch-up sensitivity.

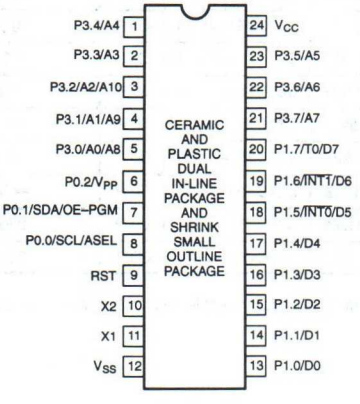
The 8XC751 contains a 2k × 8 ROM (83C751) EPROM (87C751), a 64 × 8 RAM, 19 I/O lines, a 16-bit auto-reload counter/timer, a five-source, fixed-priority level interrupt structure, a bidirectional inter-integrated circuit ($I^2C$) serial bus interface, and an on-chip oscillator.

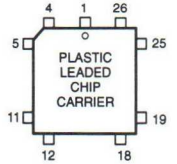
The on-board inter-integrated circuit ($I^2C$) bus interface allows the 8XC751 to operate as a master or slave device on the $I^2C$ small area network. This capability facilitates I/O and RAM expansion, access to EEPROM, processor-to-processor communication, and efficient interface to a wide variety of dedicated $I^2C$ peripherals.

## FEATURES

- 80C51 based architecture
- Inter-Integrated Circuit ($I^2C$) serial bus interface
- Small package sizes
  - 24-pin DIP (300 mil "skinny DIP")
  - 24-pin Shrink Small Outline Package
  - 28-pin PLCC
- 87C751 available in erasable quartz lid or one-time programmable plastic packages
- Wide oscillator frequency range
- Low power consumption:
  - Normal operation: less than 11mA @ 5V, 12MHz
  - Idle mode
  - Power-down mode
- 2k × 8 ROM (83C751)
  2k × 8 EPROM (87C751)
- 64 × 8 RAM
- 16-bit auto reloadable counter/timer
- Fixed-rate timer
- Boolean processor
- CMOS and TTL compatible
- Well suited for logic replacement, consumer and industrial applications
- LED drive outputs

## PIN CONFIGURATIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P3.4/A4 | 15 | P1.0/D0 |
| 2 | P3.3/A3 | 16 | P1.1/D1 |
| 3 | P3.2/A2/A10 | 17 | P1.2/D2 |
| 4 | P3.1/A1/A9 | 18 | P1.3/D3 |
| 5 | NC* | 19 | P1.4/D4 |
| 6 | P3.0/A0/A8 | 20 | P1.5/INT0/D5 |
| 7 | P0.2/VPP | 21 | NC* |
| 8 | P0.1/SDA/OE-PGM | 22 | NC* |
| 9 | P0.0//SCLASEL | 23 | P1.6/INT1/D6 |
| 10 | NC* | 24 | P1.7/T0/D7 |
| 11 | RST | 25 | P3.7/A7 |
| 12 | X2 | 26 | P3.6/A6 |
| 13 | X1 | 27 | P3.5/A5 |
| 14 | VSS | 28 | VCC |

* DO NOT CONNECT

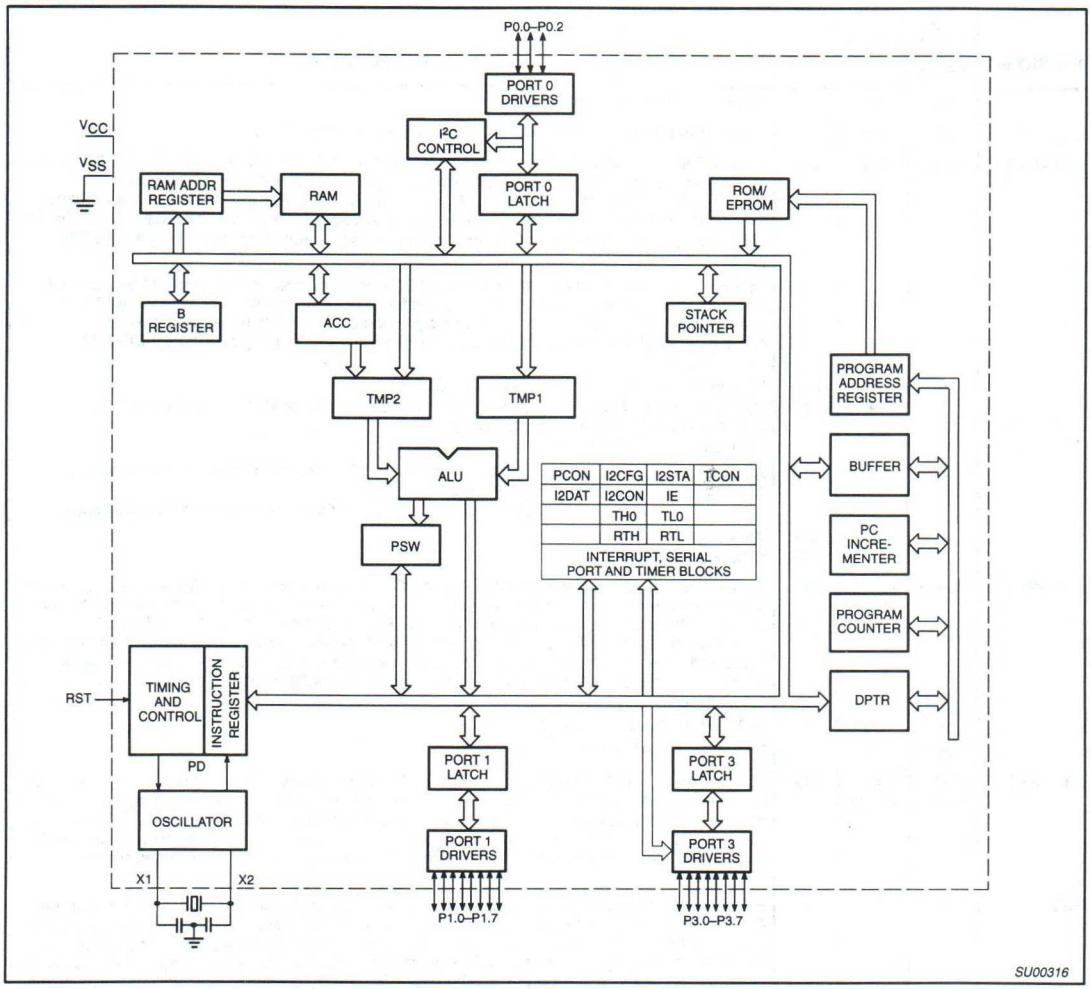
SU00315

## ORDERING INFORMATION

| ROM | EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQUENCY | DRAWING NUMBER |
|---|---|---|---|---|---|
| | S87C751–1F24 | UV | 0 to +70, Ceramic Dual In-line Package | 3.5 to 12MHz | 0586B |
| | S87C751–2F24 | UV | –40 to +85, Ceramic Dual In-line Package | 3.5 to 12MHz | 0586B |
| | S87C751–4F24 | UV | 0 to +70, Ceramic Dual In-line Package | 3.5 to 16MHz | 0586B |
| | S87C751–5F24 | UV | –40 to +85, Ceramic Dual In-line Package | 3.5 to 16MHz | 0586B |
| S83C751–1N24 | S87C751–1N24 | OTP | 0 to +70, Plastic Dual In-line Package | 3.5 to 12MHz | SOT222-1 |
| S83C751–2N24 | S87C751–2N24 | OTP | –40 to +85, Plastic Dual In-line Package | 3.5 to 12MHz | SOT222-1 |
| S83C751–4N24 | S87C751–4N24 | OTP | 0 to +70, Plastic Dual In-line Package | 3.5 to 16MHz | SOT222-1 |
| S83C751–5N24 | S87C751–5N24 | OTP | –40 to +85, Plastic Dual In-line Package | 3.5 to 16MHz | SOT222-1 |
| S83C751–1A28 | S87C751–1A28 | OTP | 0 to +70, Plastic Leaded Chip Carrier | 3.5 to 12MHz | SOT261-3 |
| S83C751–2A28 | S87C751–2A28 | OTP | –40 to +85, Plastic Leaded Chip Carrier | 3.5 to 12MHz | SOT261-3 |
| S83C751–4A28 | S87C751–4A28 | OTP | 0 to +70, Plastic Leaded Chip Carrier | 3.5 to 16MHz | SOT261-3 |
| S83C751–5A28 | S87C751–5A28 | OTP | –40 to +85, Plastic Leaded Chip Carrier | 3.5 to 16MHz | SOT261-3 |
| S83C751–1D24 | S87C751–1D24 | OTP | 0 to +70, Shrink Small Outline Package | 3.5 to 12MHz | SOT340-1 |
| S83C751–4D24 | S87C751–4D24 | OTP | 0 to +70, Shrink Small Outline Package | 3.5 to 16MHz | SOT340-1 |

**NOTE:**
1. OTP = One Time Programmable EPROM. UV = UV Erasable EPROM.

## BLOCK DIAGRAM



## PIN DESCRIPTIONS

| MNEMONIC | PIN NO. DIP/ SSOP | PIN NO. LCC | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|
| $V_{SS}$ | 12 | 14 | I | **Circuit Ground Potential** |
| $V_{CC}$ | 24 | 28 | I | **Supply voltage during normal, idle, and power-down operation.** |
| P0.0–P0.2 | 8–6 | 9–7 | I/O | **Port 0:** Port 0 is a 3-bit open-drain, bidirectional port. Port 0 pins that have 1s written to them float, and in that state can be used as high-impedance inputs. Port 0 also serves as the serial $I^2C$ interface. When this feature is activated by software, SCL and SDA are driven low in accordance with the $I^2C$ protocol. These pins are driven low if the port register bit is written with a 0 or if the $I^2C$ subsystem presents a 0. The state of the pin can always be read from the port register by the program.<br><br>To comply with the $I^2C$ specification, P0.0 and P0.1 are open drain bidirectional I/O pins with the electrical characteristics listed in the tables that follow. While these differ from "standard TTL" characteristics, they are close enough for the pins to still be used as general-purpose I/O in non-$I^2C$ applications. Port 0 also provides alternate functions for programming the EPROM memory as follows: |
| | 6 | 7 | N/A | **$V_{PP}$ (P0.2)** – Programming voltage input. (See Note 1.) |
| | 7 | 8 | I | **OE/PGM (P0.1)** – Input which specifies verify mode (output enable) or the program mode.<br>OE/PGM = 1 output enabled (verify mode).<br>OE/PGM = 0 program mode. |
| | 8 | 9 | I | **ASEL (P0.0)** – Input which indicates which bits of the EPROM address are applied to port 3.<br>ASEL = 0 low address byte available on port 3.<br>ASEL = 1 high address byte available on port 3 (only the three least significant bits are used). |
| | 7 | 8 | I/O | **SDA (P0.1)** – $I^2C$ data. |
| | 8 | 9 | I/O | **SCL (P0.0)** – $I^2C$ clock. |
| P1.0–P1.7 | 13–20 | 15–20, 23, 24 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 serves to output the addressed EPROM contents in the verify mode and accepts as inputs the value to program into the selected address during the program mode. Port 1 also serves the special function features of the 80C51 family as listed below: |
| | 18 | 20 | I | **$\overline{INT0}$ (P1.5):** External interrupt. |
| | 19 | 23 | I | **$\overline{INT1}$ (P1.6):** External interrupt. |
| | 20 | 24 | I | **T0 (P1.7):** Timer 0 external input. |
| P3.0–P3.7 | 5–1, 23–21 | 6, 4–1, 27–25 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also functions as the address input for the EPROM memory location to be programmed (or verified). The 11-bit address is multiplexed into this port as specified by P0.0/ASEL. |
| RST | 9 | 11 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on RESET using only an external capacitor to $V_{CC}$. After the device is reset, a 10-bit serial sequence, sent LSB first, applied to RESET, places the device in the programming state allowing programming address, data and $V_{PP}$ to be applied for programming or verification purposes. The RESET serial sequence must be synchronized with the X1 input. |
| X1 | 11 | 13 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. X1 also serves as the clock to strobe in a serial bit stream into RESET to place the device in the programming state. |
| X2 | 10 | 12 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**NOTES:**
1. When P0.2 is at or close to 0V it may affect the internal ROM operation. We recommend that P0.2 be tied to $V_{CC}$ via a small pullup (e.g., 2kΩ).

## ABSOLUTE MAXIMUM RATINGS[1, 2]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage from $V_{CC}$ to $V_{SS}$ | –0.5 to +6.5 | V |
| Voltage from any pin to $V_{SS}$ (except $V_{PP}$) | –0.5 to $V_{CC}$ + 0.5 | V |
| Power dissipation | 1.0 | W |
| Voltage on $V_{PP}$ pin to $V_{SS}$ | 0 to +13.0 | V |
| Maximum $I_{OL}$ per I/O pin | 10 | mA |

**NOTES:**
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10% for 87C751, $V_{CC}$ = 5V ±10% for 83C751, $V_{SS}$ = 0V[1]

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| $V_{IL}$ | Input low voltage, except SDA, SCL | | –0.5 | $0.2V_{DD}$–0.1 | V |
| $V_{IH}$ | Input high voltage, except X1, RST | | $0.2V_{CC}$+0.9 | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, X1, RST | | $0.7V_{CC}$ | $V_{CC}$+0.5 | V |
| | SDA, SCL, P0.2 | | | | |
| $V_{IL1}$ | Input low voltage | | –0.5 | $0.3V_{CC}$ | V |
| $V_{IH2}$ | Input high voltage | | $0.7V_{CC}$ | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1 and 3 | $I_{OL}$ = 1.6mA[2] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0.2 | $I_{OL}$ = 3.2mA[2] | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1 and 3 | $I_{OH}$ = –60µA | 2.4 | | V |
| | | $I_{OH}$ = –25µA | $0.75V_{CC}$ | | V |
| | | $I_{OH}$ = –10µA | $0.9V_{CC}$ | | V |
| | Port 0.0 and 0.1 ($I^2C$) – Drivers | | | | |
| $V_{OL2}$ | Output low voltage | $I_{OL}$ = 3mA | | 0.4 | V |
| | Driver, receiver combined: | (over $V_{CC}$ range) | | | |
| C | Capacitance | | | 10 | pF |
| $I_{IL}$ | Logical 0 input current, ports 1 and 3 | $V_{IN}$ = 0.45V | | –50 | µA |
| $I_{TL}$ | Logical 1 to 0 transition current, ports 1 and 3[3] | $V_{IN}$ = 2V (0 to 70°C) | | –650 | µA |
| | | $V_{IN}$ = 2V (–40 to +85°C) | | –750 | µA |
| $I_{LI}$ | Input leakage current, port 0 | 0.45 < $V_{IN}$ < $V_{CC}$ | | ±10 | µA |
| $R_{RST}$ | Internal pull-down resistor | | 25 | 175 | kΩ |
| $C_{IO}$ | Pin capacitance | Test freq = 1MHz, $T_{amb}$ = 25°C | | 10 | pF |
| $I_{PD}$ | Power-down current[4] | $V_{CC}$ = 2 to $V_{CC}$ max | | 50 | µA |
| $V_{PP}$ | $V_{PP}$ program voltage (for 87C751 only) | $V_{SS}$ = 0V<br>$V_{CC}$ = 5V±10%<br>$T_{amb}$ = 21°C to 27°C | 12.5 | 13.0 | V |
| $I_{PP}$ | Program current (for 87C751 only) | $V_{PP}$ = 13.0V | | 50 | mA |
| $I_{CC}$ | Supply current (see Figure 2) | | | | |

**NOTES TO DC ELECTRICAL CHARACTERISTICS ON NEXT PAGE.**

**NOTES TO DC ELECTRICAL CHARACTERISTICS:**

1. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.
2. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 10mA (NOTE: This is 85°C spec.)
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 67mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
3. Pins of ports 1 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when $V_{IN}$ is approximately 2V.
4. Power-down $I_{CC}$ is measured with all output pins disconnected; port 0 = $V_{CC}$; X2, X1 n.c.; RST = $V_{SS}$.
5. Active $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; RST = port 0 = $V_{CC}$. $I_{CC}$ will be slightly higher if a crystal oscillator is used.
6. Idle $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; port 0 = $V_{CC}$; RST = $V_{SS}$.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10% for 87C751, $V_{CC}$ = 5V ±10% for 83C751, $V_{SS}$ = 0V[1, 2]

| SYMBOL | PARAMETER | 12MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | |
| 1/$t_{CLCL}$ | Oscillator frequency: | | | 3.5<br>3.5 | 12<br>16 | MHz<br>MHz |
| **External Clock (Figure 1)** | | | | | | |
| $t_{CHCX}$ | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | Fall time | | 20 | | 20 | ns |

**NOTES:**

1. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.
2. Load capacitance for ports = 80pF.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

C – Clock
D – Input data
H – Logic level high
L – Logic level low
Q – Output data
T – Time
V – Valid
X – No longer a valid logic level
Z – Float

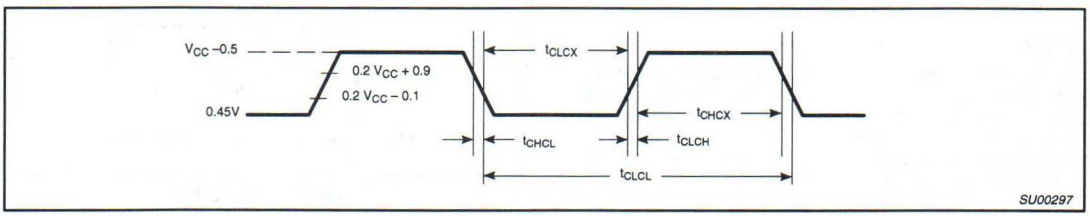



**Figure 1. External Clock Drive**

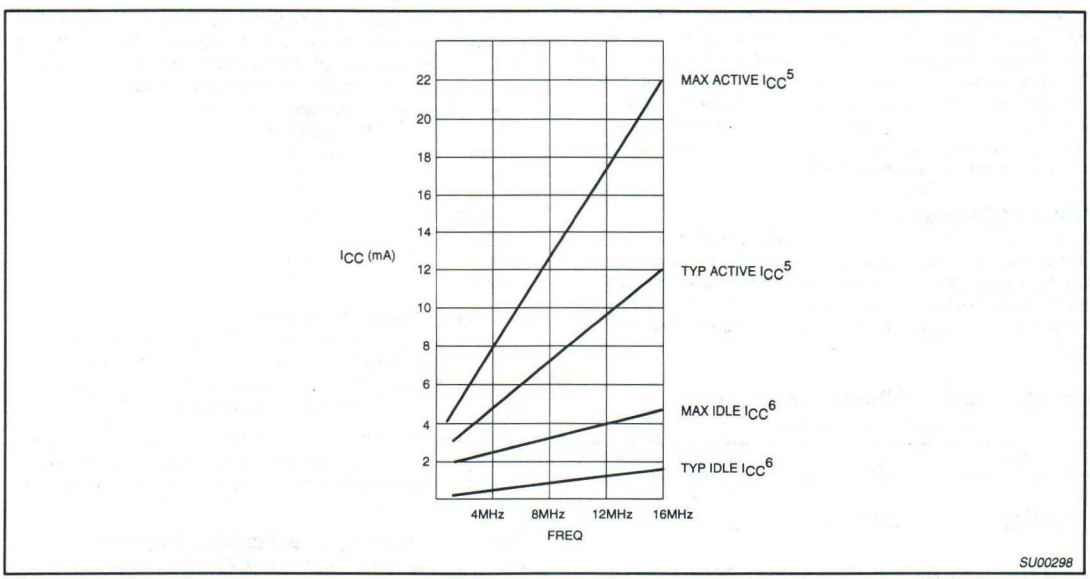



**Figure 2. $I_{CC}$ vs. FREQ**
**Maximum $I_{CC}$ values taken at $V_{CC}$ max and worst case temperature.**
**Typical $I_{CC}$ values taken at $V_{CC}$ = 5.0V and 25°C.**
**Notes 5 and 6 refer to DC Electrical Characteristics.**

## OSCILLATOR CHARACTERISTICS

X1 and X2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, X1 should be driven while X2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-up reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-up, the voltage on $V_{CC}$ and RST must come up at the same time for a proper start-up.

## IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. the control bits for the reduced power modes are in the special function register PCON.

**Table 1. External Pin Status During Idle and Power-Down Modes**

| MODE | Port 0 | Port 1 | Port 2 |
|---|---|---|---|
| Idle | Data | Data | Data |
| Power-down | Data | Data | Data |

## DIFFERENCES BETWEEN THE 8XC751 AND THE 80C51

### Memory Organization

The central processing unit (CPU) manipulates operands in two address spaces as shown in Figure 3. The part's internal memory space consists of 2k bytes of program memory, and 64 bytes of data RAM overlapped with the 128-byte special function register area. The differences from the 80C51 are in RAM size (64 bytes vs. 128 bytes), in external RAM access (not available on the 83C751), in internal ROM size (2k bytes vs. 4k bytes), and in external program memory expansion (not available on the 83C751). The 128-byte special function register (SFR) space is accessed as on the 80C51 with some of the registers having been changed to reflect changes in the 83C751 peripheral functions. The stack may be located anywhere in internal RAM by loading the 8-bit stack pointer (SP). It should be noted that stack depth is limited to 64 bytes, the amount of available RAM. A reset loads the stack pointer with 07 (which is pre-incremented on a PUSH instruction).

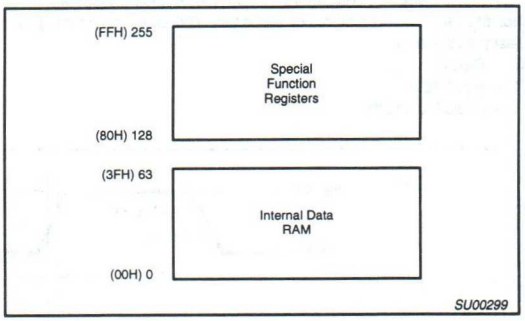



**Figure 3. Memory Map**

### Program Memory

On the 8XC751, program memory is 2048 bytes long and is not externally expandable, so the 80C51 instructions MOVX, LJMP, and LCALL are not implemented. The only fixed locations in program memory are the addresses at which execution is taken up in response to reset and interrupts, which are as follows:

**Program Memory**

| Event | Address |
|---|---|
| Reset | 000 |
| External $\overline{INT0}$ | 003 |
| Counter/timer 0 | 00B |
| External $\overline{INT1}$ | 013 |
| Timer I | 01B |
| $I^2C$ serial | 023 |

### Counter/Timer Subsystem

The 8XC751 has one counter/timer called timer/counter 0. Its operation is similar to mode 2 operation on the 80C51, but is extended to 16 bits with 16 bits of autoload. The controls for this counter are centralized in a single register called TCON.

A watchdog timer, called Timer I, is for use with the $I^2C$ subsystem. In $I^2C$ applications, this timer is dedicated to time-generation and bus monitoring of the $I^2C$. In non-$I^2C$ applications, it is available for use as a fixed time-base.

### Counter Timer – Special Function Register

The counter/timer has only one mode of operation, so the TMOD SFR is not used. There is also only one counter/timer, so there is no need for the TL1 and TH1 SFRs found on the 80C51. These have been replaced on the 83C751 by RTL and RTH, the counter/timer reload registers. Table 3 shows the special function registers, their locations, and reset values.

### Interrupt Subsystem – Fixed Priority

The IP register and the 2-level interrupt system of the 80C51 are eliminated. Simultaneous interrupt conditions are resolved by a single-level, fixed priority as follows:

| | |
|---|---|
| Highest priority: | Pin $\overline{INT0}$ |
| | Counter/timer flag 0 |
| | Pin $\overline{INT1}$ |
| | Timer I |
| Lowest priority: | Serial $I^2C$ |

### Special Function Register – Interrupt Subsystem

Because the interrupt structure is single level on the 83C751, there is no need for the IP SFR, so it is not used.

### Serial Communications

The 8XC751 contains an I²C serial communications port instead of the 80C51 UART. The I²C serial port is a single bit hardware interface with all of the hardware necessary to support multimaster and slave operations. Also included are receiver digital filters and timer (timer I) for communication watch-dog purposes. The I²C serial port is controlled through four special function registers; I²C control, I²C data, I²C status, and I²C configuration.

### Special Function Register – Serial Communications

The 83C751 contains many of the special function registers (SFR) that are found on the 80C51. Due to the different peripheral features on the 83C751, there are several additional SFRs and several that have been changed.

Since the standard UART found on the 80C51 has been replaced by the I²C serial interface, the UART SFRs, SCON, and SBUF have been replaced by I2CON and I2DAT, and two additional I²C registers have been added (I2STA and I2CFG).

### I/O Port Latches (P0, P1, P3)

The port latches function the same as those on the 80C51. Since there is no port 2 on the 83C751, the P2 latch is not used. Port 0 on the 83C751 has only 3 bits, so only 3 bits of the P0 SFR have a useful function.

### Special Function Register – I/O Port Latches

There is no Port2 on the 8XC751, so P2 is not used. Also, only 3 bits of P0 SFR have a useful function.

### Data Pointer (DPTR)

The data pointer (DPTR) consists of a high byte (DPH) and a low byte (DPL). In the 80C51 this register allows the access of external data memory using the MOVX instruction. Since the 83C751 does not support MOVX or external memory accesses, this register is generally used as a 16-bit offset pointer of the accumulator in a MOVC instruction. DPTR may also be manipulated as two independent 8-bit registers.

**Table 2. I²C Special Function Register Addresses**

| REGISTER ADDRESS | | | BIT ADDRESS | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| NAME | SYMBOL | ADDRESS | MSB | | | | | | | LSB |
| I²C control | I2CON | 98 | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 |
| I²C data | I2DAT | 99 | – | – | – | – | – | – | – | – |
| I²C configuration | I2CFG | D8 | DF | DE | DD | DC | DB | DA | D9 | D8 |
| I²C status | I2STA | F8 | FF | FE | FD | FC | FB | FA | F9 | F8 |

### ROM CODE SUBMISSION

When submitting ROM code for the 80C751, the following must be specified:

1. 2k byte user ROM data

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 07FFH | DATA | 7:0 | User ROM Data |

**Table 3. 8XC751 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes) | | | | | | | | | | |
| DPH | High byte | 83H | | | | | | | | | 00H |
| DPL | Low byte | 82H | | | | | | | | | 00H |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| I²CFG*# | I²C configuration | D8H/RD | SLAVEN | MASTRQ | 0 | TIRUN | – | – | CT1 | CT0 | 0000xx00B |
| | | WR | SLAVEN | MASTRQ | CLRTI | TIRUN | – | – | CT1 | CT0 | |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| I²CON*# | I²C control | 98H/RD | RDAT | ATN | DRDY | ARL | STR | STP | MASTER | – | 81H |
| | | WR | CXA | IDLE | CDR | CARL | CSTR | CSTP | XSTR | XSTP | |
| | | | | | | | | | | | |
| I²DAT# | I²C data | 99H/RD | RDAT | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 80H |
| | | WR | XDAT | X | X | X | X | X | X | X | |
| | | | FF | FE | FD | FC | FB | FA | F9 | F8 | |
| I²STA*# | I²C status | F8H | – | IDLE | XDATA | XACTV | MAKSTR | MAKSTP | XSTR | XSTP | x0100000B |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | – | – | EI2 | ETI | EX1 | ET0 | EX0 | 00H |
| | | | | | | | | 82 | 81 | 80 | |
| P0*# | Port 0 | 80H | – | – | – | – | – | – | SDA | SCL | xxxxx111B |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | |
| P1* | Port 1 | 90H | T0 | $\overline{INT1}$ | $\overline{INT0}$ | – | – | – | – | – | FFH |
| P3* | Port 3 | B0H | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | FFH |
| | | | | | | | | | | | |
| PCON# | Power control | 87H | – | – | – | – | – | – | PD | IDL | xxxxxx00B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON*# | Timer/counter control | 88H | GATE | C/T | TF | TR | IE0 | IT0 | IE1 | IT1 | 00H |
| TL# | Timer low byte | 8AH | | | | | | | | | 00H |
| TH# | Timer high byte | 8CH | | | | | | | | | 00H |
| RTL# | Timer low reload | 8BH | | | | | | | | | 00H |
| RTH# | Timer high reload | 8DH | | | | | | | | | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

### I/O Port Structure

The 8XC751 has two 8-bit ports (ports 1 and 3) and one 3-bit port (port 0). All three ports on the 8XC751 are bidirectional. Each consists of a latch (special function register P0, P1, P3), an output driver, and an input buffer. Three port 1 pins and two port 0 pins are multifunctional. In addition to being port pins, these pins serve the function of special features as follows:

| Port Pin | Alternate Function |
|---|---|
| P0.0 | $I^2C$ clock (SCL) |
| P0.1 | $I^2C$ data (SDA) |
| P1.5 | INT0 (external interrupt 0 input) |
| P1.6 | INT1 (external interrupt 1 input) |
| P1.7 | T0 (timer 0 external input) |

Ports 1 and 3 are identical in structure to the same ports on the 80C51. The structure of port 0 on the 8XC751 is similar to that of the 80C51 but does not include address/data input and output circuitry. As on the 80C51, ports 1 and 3 are quasi-bidirectional while port 0 is bidirectional with no internal pullups.

### Timer/Counter

The 8XC751 has two timers: a 16-bit timer/counter and a 10-bit fixed-rate timer. The 16-bit timer/counter's operation is similar to mode 2 operation on the 80C51, but is extended to 16 bits. The timer/counter is clocked by either 1/12 the oscillator frequency or by transitions on the T0 pin. The C/T pin in special function register TCON selects between these two modes. When the TCON TR bit is set, the timer/counter is enabled. Register pair TH and TL are incremented by the clock source. When the register pair overflows, the register pair is reloaded with the values in registers RTH and RTL. The value in the reload registers is left unchanged. See the 83C751 counter/timer block diagram in Figure 4. The TF bit in special function register TCON is set on counter overflow and, if the interrupt is enabled, will generate an interrupt.

### TCON Register

MSB LSB

| GATE | C/T | TF | TR | IE0 | IT0 | IE1 | IT1 |
|---|---|---|---|---|---|---|---|

GATE 1 – Timer/counter is enabled only when INT0 pin is high, and TR is 1.
0 – Timer/counter is enabled when TR is 1.
C/T 1 – Counter/timer operation from T0 pin.
0 – Timer operation from internal clock.
TF 1 – Set on overflow of TH.
0 – Cleared when processor vectors to interrupt routine and by reset.
TR 1 – Timer/counter enabled.
0 – Timer/counter disabled.
IE0 1 – Edge detected in $\overline{\text{INT0}}$.
IT0 1 – $\overline{\text{INT0}}$ is edge triggered.
0 – $\overline{\text{INT0}}$ is level sensitive.
IE1 1 – Edge detected on $\overline{\text{INT1}}$.
IT1 1 – $\overline{\text{INT1}}$ is edge triggered.
0 – $\overline{\text{INT1}}$ is level sensitive.

These flags are functionally identical to the corresponding 80C51 flags, except that there is only one timer on the 83C751 and the flags are therefore combined into one register.

Note that the positions of the IE0/IT0 and IE1/IT1 bits are transposed from the positions used in the standard 80C51 TCON register.

Timer I is used to control the timing of the $I^2C$ bus and also to detect a "bus locked" condition, by causing an interrupt when nothing happens on the $I^2C$ bus for an inordinately long period of time while a transmission is in progress. If the interrupt does not occur, the program can attempt to correct the fault and allow the last $I^2C$ transmission to be repeated.

The $I^2C$ watchdog timer, timer I, is also available as a general-purpose fixed-rate timer when the $I^2C$ interface is not being used. A clock rate of 1/12 the oscillator frequency forms the input to the timer. Timer I has a timeout interval of 1024 machine cycles when used as a fixed-rate timer.

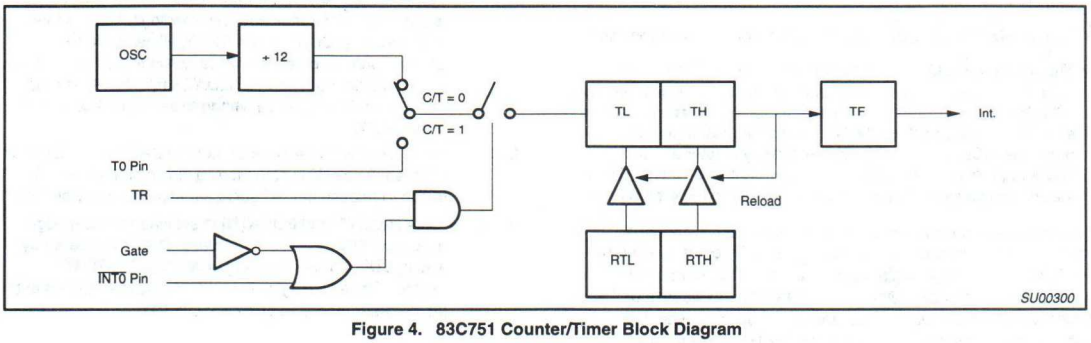



**Figure 4. 83C751 Counter/Timer Block Diagram**

### I²C Serial Interface

The I²C bus uses two wires (SDA and SCL) to transfer information between devices connected to the bus. The main features of the bus are:

- Bidirectional data transfer between masters and slaves
- Serial addressing of slaves (no added wiring)
- Acknowledgment after each transferred byte
- Multimaster bus
- Arbitration between simultaneously transmitting masters without corruption of serial data on bus
- The 82B715 extends communication distance to 100 feet (30M).

A large family of I²C compatible ICs is available. See the I²C section of this manual for more details on the bus and available ICs.

The 83C751 I²C subsystem includes hardware to simplify the software required to drive the I²C bus. The hardware is a single bit interface which in addition to including the necessary arbitration and framing error checks, includes clock stretching and a bus timeout timer. The interface is synchronized to software either through polled loops or interrupts. Refer to the application note AN422, in Section 4, entitled "Using the 8XC751 Microcontroller as an I²C Bus Master" for additional discussion of the 83C751 I²C interface and sample driver routines.

Six time spans are important in I²C operation and are insured by timer I:

- The MINIMUM HIGH time for SCL when this device is the master.
- The MINIMUM LOW time for SCL when this device is a master. This is not very important for a single-bit hardware interface like this one, because the SCL low time is stretched until the software responds to the I²C flags. The software response time normally meets or exceeds the MIN LO time. In cases where the software responds within MIN HI + MIN LO) time, timer I will ensure that the minimum time is met.
- The MINIMUM SCL HIGH TO SDA HIGH time in a stop condition.
- The MINIMUM SDA HIGH TO SDA LOW time between I²C stop and start conditions (4.7µs, see spec.).
- The MINIMUM SDA LOW TO SCL LOW time in a start condition.
- The MAXIMUM SCL CHANGE time while an I²C frame is in progress. A frame is in progress between a start condition and the following stop condition. This time span serves to detect a lack of software response on this 8XC751 as well as external I²C problems. SCL "stuck low" indicates a faulty master or slave. SCL "stuck high" may mean a faulty device, or that noise induced onto the I²C bus caused all masters to withdraw from I²C arbitration.

The first five of these times are 4.7µs (see I²C specification) and are covered by the low order three bits of timer I. Timer I is clocked by the 8XC751 oscillator, which can vary in frequency from 0.5 to 16MHz. Timer I can be preloaded with one of four values to optimize timing for different oscillator frequencies. At lower frequencies, software response time is increased and will degrade maximum performance of the I²C bus. See special function register I2CFG description for prescale values (CT0, CT1).

The MAXIMUM SCL CHANGE time is important, but its exact span is not critical. The complete 10 bits of timer I are used to count out the maximum time. When I²C operation is enabled, this counter is cleared by transitions on the SCL pin. The timer does not run between I²C frames (i.e., whenever reset or stop occurred more recently than the last start). When this counter is running, it will carry out after 1020 to 1023 machine cycles have elapsed since a change on SCL. A carry out causes a hardware reset of the 83C751 I²C interface and generates an interrupt if the timer I interrupt is enabled. In cases where the bus hangup is due to a lack of software response by this 83C751, the reset releases SCL and allows I²C operation among other devices to continue.

### I²C Interrupts

If I²C interrupts are enabled (EA and EI2 are both set to 1), an I²C interrupt will occur whenever the ATN flag is set by a start, stop, arbitration loss, or data ready condition (refer to the description of ATN following). In practice, it is not efficient to operate the I²C interface in this fashion because the I²C interrupt service routine would somehow have to distinguish between hundreds of possible conditions. Also, since I²C can operate at a fairly high rate, the software may execute faster if the code simply waits for the I²C interface.

Typically, the I²C interrupt should only be used to indicate a start condition at an idle slave device, or a stop condition at an idle master device (if it is waiting to use the I²C bus). This is accomplished by enabling the I²C interrupt only during the aforementioned conditions.

### I²C Register I2CON

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **Read** | RDAT | ATN | DRDY | ARL | STR | STP | MASTER | – |
| **Write** | CXA | IDLE | CDR | CARL | CSTR | CSTP | XSTR | XSTP |

Reading I2CON

RDAT — The data from SDA is captured into "Receive DATa" whenever a rising edge occurs on SCL. RDAT is also available (with seven low-order zeros) in the I2DAT register. The difference between reading it here and there is that reading I2DAT clears DRDY, allowing the I²C to proceed on to another bit. Typically, the first seven bits of a received byte are read from I2DAT, while the 8th is read here. Then I2DAT can be written to send the Ack bit and clear DRDY.

ATN — "ATteNtion" is 1 when one or more of DRDY, ARL, STR, or STP is 1. Thus, ATN comprises a single bit that can be tested to release the I²C service routine from a "wait loop."

DRDY — "Data ReaDY" (and thus ATN) is set when a rising edge occurs on SCL, except at idle slave. DRDY is cleared by writing CDR = 1, or by writing or reading the I2DAT register. The following low period on SCL is stretched until the program responds by clearing DRDY.

Checking ATN and DRDY

When a program detects ATN = 1, it should next check DRDY. If DRDY = 1, then if it receives the last bit, it should capture the data from RDAT (in I2DAT or I2CON). Next, if the next bit is to be sent, it should be written to I2DAT. One way or another, it should clear DRDY and then return to monitoring ATN. Note that if any of ARL, STR, or STP is set, clearing DRDY will not release SCL to high, so that the $I^2C$ will not go on to the next bit. If a program detects ATN = 1, and DRDY = 0, it should go on to examine ARL, STR, and STP.

ARL "Arbitration Loss" is 1 when transmit Active was set, but this 83C751 lost arbitration to another transmitter. Transmit Active is cleared when ARL is 1. There are four separate cases in which ARL is set.

1. If the program sent a 1 or repeated start, but another device sent a 0, or a stop, so that SDA is 0 at the rising edge of SCL. (If the other device sent a stop, the setting of ARL will be followed shortly by STP being set.)
2. If the program sent a 1, but another device sent a repeated start, and it drove SDA low before the 83C751 could drive SCL low. (This type of ARL is always accompanied by STR = 1.)
3. In master mode, if the program sent a repeated start, but another device sent a 1, and it drove SCL low before this 83C751 could drive SDA low.
4. In master mode, if the program sent stop, but it could not be sent because another device sent a 0.

STR "STaRt" is set to a 1 when an $I^2C$ start condition is detected at a non-idle slave or at a master. (STR is not set when an idle slave becomes active due to a start bit; the slave has nothing useful to do until the rising edge of SCL sets DRDY.)

STP "SToP" is set to 1 when an $I^2C$ stop condition is detected at a non-idle slave or at a master. (STP is not set for a stop condition at an idle slave.)

MASTER "MASTER" is 1 if this 83C751 is currently a master on the $I^2C$. MASTER is set when MASTRQ is 1 and the bus is not busy (i.e., if a start bit hasn't been received since reset or a "Timer I" time-out, or if a stop has been received since the last start). MASTER is cleared when ARL is set, or after the software writes MASTRQ = 0 and then XSTP = 1.

Writing I2CON

Typically, for each bit in an $I^2C$ message, a service routine waits for ATN = 1. Based on DRDY, ARL, STR, and STP, and on the current bit position in the message, it may then write I2CON with one or more of the following bits, or it may read or write the I2DAT register.

CXA Writing a 1 to "Clear Xmit Active" clears the Transmit Active state. (Reading the I2DAT register also does this.)

Regarding Transmit Active

Transmit Active is set by writing the I2DAT register, or by writing I2CON with XSTR = 1 or XSTP = 1. The $I^2C$ interface will only drive the SDA line low when Transmit Active is set, and the ARL bit will only be set to 1 when Transmit Active is set. Transmit Active is cleared by reading the I2DAT register, or by writing I2CON with CXA = 1. Transmit Active is automatically cleared when ARL is 1.

IDLE Writing 1 to "IDLE" causes a slave's $I^2C$ hardware to ignore the $I^2C$ until the next start condition (but if MASTRQ is 1, then a stop condition will make the 83C751 into a master).

CDR Writing a 1 to "Clear Data Ready" clears DRDY. (Reading or writing the I2DAT register also does this.)

CARL Writing a 1 to "Clear Arbitration Loss" clears the ARL bit.

CSTR Writing a 1 to "Clear STaRt" clears the STR bit.

CSTP Writing a 1 to "Clear SToP" clears the STP bit. Note that if one or more of DRDY, ARL, STR, or STP is 1, the low time of SCL is stretched until the service routine responds by clearing them.

XSTR Writing 1s to "Xmit repeated STaRt" and CDR tells the $I^2C$ hardware to send a repeated start condition. This should only be at a master. Note that XSTR need not and should not be used to send an "initial" (nonrepeated) start; it is sent automatically by the $I^2C$ hardware. Writing XSTR = 1 includes the effect of writing I2DAT with XDAT = 1; it sets Transmit Active and releases SDA to high during the SCL low time. After SCL goes high, the $I^2C$ hardware waits for the suitable minimum time and then drives SDA low to make the start condition.

XSTP Writing 1s to "Xmit SToP" and CDR tells the $I^2C$ hardware to send a stop condition. This should only be done at a master. If there are no more messages to initiate, the service routine should clear the MASTRQ bit in I2CFG to 0 before writing XSTP with 1. Writing XSTP = 1 includes the effect of writing I2DAT with XDAT = 0; it sets Transmit Active and drives SDA low during the SCL low time. After SCL goes high, the $I^2C$ hardware waits for the suitable minimum time and then releases SDA to high to make the stop condition.

**NOTE:** Because of the manner in which register bit addressing is implemented in the 80C51 family, the I2CON register should never be altered by use of the SETB, CLR, CPL, MOV (bit), or JBC instructions. This is due to the fact that read and write functions of this register are different. Testing of I2CON bits via the JB and JNB instructions is supported.

### I²C Register I2DAT

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Read | RDAT | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Write | XDAT | X | X | X | X | X | X | X |

RDAT "Receive DATa" is captured from SDA every rising edge of SCL. Reading I2DAT also clears DRDY and the Transmit Active state.

XDAT "Xmit Data" sets the data for the next bit. Writing I2DAT also clears DRDY and sets the Transmit Active state.

Regarding Software Response Time

Because the 83C751 can run at 16MHz, and because the I²C interface is optimized for high-speed operation, it is quite likely that an I²C service routine will sometimes respond to DRDY (which is set at a rising edge of SCL) and write I2DAT before SCL has gone low again. If XDAT were applied directly to SDA, this situation would produce an I²C protocol violation. The programmer need not worry about this possibility because XDAT is applied to SDA only when SCL is low.

Conversely, a program that includes an I²C service routine may take a long time to respond to DRDY. Typically, an I²C routine operates on a flag-polling basis during a message, with interrupts from other peripheral functions enabled. If an interrupt occurs, it will delay the response of the I²C service routine. The programmer need not worry about this very much either, because the I²C hardware stretches the SCL low time until the service routine responds. The only constraint on the response is that it must not exceed the Timer I time-out, which is at least 765 microseconds.

### I²C Register I2CFG

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Read | SLAVEN | MASTRQ | 0 | TIRUN | – | – | CT1 | CT0 |
| Write | SLAVEN | MASTRQ | CLRTI | TIRUN | – | – | CT1 | CT0 |

SLAVEN Writing a 1 to "SLAVe ENable" enables the slave functions of the I²C subsystem. If SLAVEN and MASTRQ are 0, the I²C hardware is disabled. This bit is cleared to 0 by reset and by an I²C time-out.

MASTRQ Writing a 1 to "MASTRQ" requests mastership of the I²C. If a frame from another master is in progress when this bit is changed from 0 to 1, action is delayed until a stop condition is detected. Then, or immediately if a frame is not in progress, a start condition is sent and DRDY is set (thus making ATN 1 and generating an I²C interrupt). When a master wishes to release mastership status of the I²C, it writes a 1 to XSTP in I2CON. MASTRQ is cleared by reset and by an I²C time-out.

CLRTI Writing a 1 to this bit clears the Timer I interrupt flag. This bit position always reads as a 0.

TIRUN Writing a 1 to this bit lets Timer I run; a zero stops and clears it. Together with SLAVEN, MASTRQ, and MASTER, this bit determines operational modes as shown in Table 4.

CT1,0 These two bits are programmed as a function of the OSC rate, to optimize the MIN HI and LO time of SCL when this 83C751 is a master on the I²C. The time value determined by these bits controls both of these parameters, and also the timing for stop and start conditions. These bits are cleared to 00 by reset.

Values to be used in the CT1 and CT0 bits are shown in Table 5. To allow the I²C bus to run at the maximum rate for a particular oscillator frequency, compare the actual oscillator rate to the $f_{OSC}$ max column in the table. The value for CT1 and CT0 is found in the first line of the table where $f_{OSC}$ max is greater than or equal to the actual frequency.

The table also shows the osc/12 count for various settings of CT1/CT0. This allows calculation of the actual minimum high and low times for SCL as follows:

SCL min high/low time (in microseconds) = 12 * count / osc (in MHz)

For instance, at a 16MHz frequency, with CT1/CT0 set to 10, the minimum SCL high and low times will be 5.25µs.

The table also shows the Timer I timeout period (given in machine cycles) for each CT1/CT0 combination. The timeout period varies because of the way in which minimum SCL high and low times are measured. When the I²C interface is operating, Timer I is preloaded at every SCL transition with a value dependent upon CT1/CT0. The preload value is chosen such that a minimum SCL high or low time has elapsed when Timer I reaches a count of 008 (the actual value preloaded into Timer I is 8 minus the osc/12 count).

**Table 4. Interaction of TIRUN with SLAVEN, MASTRQ, and MASTER**

| SLAVEN, MASTRQ, MASTER | TIRUN | OPERATING MODE |
|---|---|---|
| All 0 | 0 | The I²C interface is disabled. Timer I is cleared and does not run. This is the state assumed after a reset. If an I²C application wants to ignore the I²C at certain times, it should write SLAVEN, MASTRQ, and TIRUN all to zero. |
| All 0 | 1 | The I²C interface is disabled. Timer I operates as a free-running time base. Use this mode only in non-I²C applications. |
| Any or all 1 | 0 | The I²C interface is enabled. The 3 low-order bits of Timer I run for min-time generation, but the hi-order bits do not, so that there is no checking for I²C being "hung." This configuration can be used for very slow I²C operation. |
| Any or all 1 | 1 | The I²C interface is enabled. Timer I runs during frames on the I²C, and is cleared by transitions on SCL, and by Start and Stop conditions. This is the normal state for I²C operation. |

**Table 5. CT1, CT0 Values**

| CT1, CT0 | OSC/12 COUNT | $f_{OSC}$ MAX | TIMEOUT PERIOD |
|---|---|---|---|
| 10 | 7 | 16.8MHz | 1023 cycles |
| 01 | 6 | 14.4MHz | 1022 cycles |
| 00 | 5 | 12.0MHz | 1021 cycles |
| 11 | 4 | 9.6MHz | 1020 cycles |

**I²C Register I2STA**

READ ONLY

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | IDLE | XDATA | XACTV | MAKSTR | MAKSTP | XSTR | XSTP |

MSB LSB

This register is read only and reflects the internal status of the I²C hardware. IDLE, XSTR, and XSTP reflect the status of the like named bits in the I2CON register.

XDATA The content of the transmitter buffer.

XACTV Transmitter active.

MAKSTR This bit is high while the hardware is effecting a start condition.

MAKSTP This bit is high while the hardware is effecting a stop condition.

XSTR This bit is active while the hardware is effecting a repeated start condition.

XSTP This bit is active while the hardware is effecting a repeated stop condition.

**Interrupts**

The interrupt structure is a five-source, one-level interrupt system. Interrupt sources common to the 80C51 are the external interrupts ($\overline{INT0}$, $\overline{INT1}$) and the timer/counter interrupt (ET0). The I²C interrupt (EI2) and Timer I interrupt (ETI) are the other two interrupt sources. The interrupt sources are listed below in their order of polling sequence priority.

Upon interrupt or reset the program counter is loaded with specific values for the appropriate interrupt service routine in program memory. These values are:

| Event | Program Memory Address | Priority |
|---|---|---|
| Reset | 000 | Highest |
| INT0 | 003 | |
| Counter/Timer 0 | 00B | |
| INT1 | 013 | |
| Timer I | 01B | |
| I²C | 023 | Lowest |

The interrupt enable register (IE) is used to individually enable or disable the five sources. Bit $\overline{EA}$ in the interrupt enable register can be used to globally enable or disable all interrupt sources. The interrupt enable register is described below. All other interrupt details are based on the 80C51 interrupt architecture.

**Interrupt Enable Register**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| $\overline{EA}$ | X | X | EI2 | ETI | EX1 | ET0 | EX0 |

| Symbol | Position | Function |
|---|---|---|
| $\overline{EA}$ | IE.7 | Disables all interrupts. If EA = 0, no interrupt will be acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit |
| – | IE.6 | Reserved |
| – | IE.5 | Reserved |
| EI2 | IE.4 | Enables or disables the I²C interrupt. If EI2 = 0, the I²C interrupt is disabled |
| ETI | IE.3 | Enables or disables the Timer I overflow interrupt. If ETI = 0, the Timer I interrupt is disabled. |
| EX1 | IE.2 | Enables or disables external interrupt 1. If EX1 = 0, external interrupt 1 is disabled. |
| ET0 | IE.1 | Enables or disables the Timer 0 overflow interrupt. If ET0 = 0, theTimer 0 interrupt is disabled. |
| EX0 | IE.0 | Enables or disables external interrupt 0. If EX0 = 0, external interrupt 0 is disabled. |

## 87C751 PROGRAMMING CONSIDERATIONS

### EPROM Characteristics

The 87C751 is programmed by using a modified Quick-Pulse Programming algorithm similar to that used for devices such as the 87C451 and 87C51. It differs from these devices in that a serial data stream is used to place the 87C751 in the programming mode.

Figure 5 shows a block diagram of the programming configuration for the 87C751. Port pin P0.2 is used as the programming voltage supply input ($V_{PP}$ signal). Port pin P0.1 is used as the program (PGM/) signal. This pin is used for the 25 programming pulses.

Port 3 is used as the address input for the byte to be programmed and accepts both the high and low components of the eleven bit address. Multiplexing of these address components is performed using the ASEL input. The user should drive the ASEL input high and then drive port 3 with the high order bits of the address. ASEL should remain high for at least 13 clock cycles. ASEL may then be driven low which latches the high order bits of the address internally. the high address should remain on port 3 for at least two clock cycles after ASEL is driven low. Port 3 may then be driven with the low byte of the address. The low address will be internally stable 13 clock cycles later. The address will remain stable provided that the low byte placed on port 3 is held stable and ASEL is kept low. **Note:** ASEL needs to be pulsed high only to change the high byte of the address.

Port 1 is used as a bidirectional data bus during programming and verify operations. During programming mode, it accepts the byte to be programmed. During verify mode, it provides the contents of the EPROM location specified by the address which has been supplied to Port 3.

The XTAL1 pin is the oscillator input and receives the master system clock. This clock should be between 1.2 and 6MHz.

The RESET pin is used to accept the serial data stream that places the 87C751 into various programming modes. This pattern consists of a 10-bit code with the LSB sent first. Each bit is synchronized to the clock input, X1.

### Programming Operation

Figures 6 and 7 show the timing diagrams for the program/verify cycle. RESET should initially be held high for at least two machine cycles. P0.1 (PGM/) and P0.2 ($V_{PP}$) will be at $V_{OH}$ as a result of the RESET operation. At this point, these pins function as normal quasi-bidirectional I/O ports and the programming equipment may pull these lines low. However, prior to sending the 10-bit code on the RESET pin, the programming equipment should drive these pins high ($V_{IH}$). The RESET pin may now be used as the serial data input for the data stream which places the 87C751 in the programming mode. Data bits are sampled during the clock high time and thus should only change during the time that the clock is low. Following transmission of the last data bit, the RESET pin should be held low.

Next the address information for the location to be programmed is placed on port 3 and ASEL is used to perform the address multiplexing, as previously described. At this time, port 1 functions as an output.

A high voltage $V_{PP}$ level is then applied to the $V_{PP}$ input (P0.2). (This sets Port 1 as an input port). The data to be programmed into the EPROM array is then placed on Port 1. This is followed by a series of programming pulses applied to the PGM/ pin (P0.1). These pulses are created by driving P0.1 low and then high. This pulse is repeated until a total of 25 programming pulses have occurred. At the conclusion of the last pulse, the PGM/ signal should remain high.

The $V_{PP}$ signal may now be driven to the $V_{OH}$ level, placing the 87C751 in the verify mode. (Port 1 is now used as an output port). After four machine cycles (48 clock periods), the contents of the addressed location in the EPROM array will appear on Port 1.

The next programming cycle may now be initiated by placing the address information at the inputs of the multiplexed buffers, driving the $V_{PP}$ pin to the $V_{PP}$ voltage level, providing the byte to be programmed to Port1 and issuing the 26 programming pulses on the PGM/ pin, bringing $V_{PP}$ back down to the $V_C$ level and verifying the byte.

### Programming Modes

The 87C751 has four programming features incorporated within its EPROM array. These include the USER EPROM for storage of the application's code, a 16-byte encryption key array and two security bits. Programming and verification of these four elements are selected by a combination of the serial data stream applied to the RESET pin and the voltage levels applied to port pins P0.1 and P0.2. The various combinations are shown in Table 6.

### Encryption Key Table

The 87C751 includes a 16-byte EPROM array that is programmable by the end user. The contents of this array can then be used to encrypt the program memory contents during a program memory verify operation. When a program memory verify operation is performed, the contents of the program memory location is XNOR'ed with one of the bytes in the 16-byte encryption table. The resulting data pattern is then provided to port 1 as the verify data. The encryption mechanism can be disable, in essence, by leaving the bytes in the encryption table in their erased state (FFH) since the XNOR product of a bit with a logical one will result in the original bit. The encryption bytes are mapped with the code memory in 16-byte groups. the first byte in code memory will be encrypted with the first byte in the encryption table; the second byte in code memory will be encrypted with the second byte in the encryption table and so forth up to and including the 16the byte. The encryption repeats in 16-byte groups; the 17th byte in the code memory will be encrypted with the first byte in the encryption table, and so forth.

### Security Bits

Two security bits, security bit 1 and security bit 2, are provided to limit access to the USER EPROM and encryption key arrays. Security bit 1 is the program inhibit bit, and once programmed performs the following functions:

1. Additional programming of the USER EPROM is inhibited.
2. Additional programming of the encryption key is inhibited.
3. Verification of the encryption key is inhibited.
4. Verification of the USER EPROM and the security bit levels may still be performed.

(If the encryption key array is being used, this security bit should be programmed by the user to prevent unauthorized parties from reprogramming the encryption key to all logical zero bits. Such programming would provide data during a verify cycle that is the logical complement of the USER EPROM contents).

Security bit 2, the verify inhibit bit, prevents verification of both the USER EPROM array and the encryption key arrays. The security bit levels may still be verified.

### Programming and Verifying Security Bits

Security bits are programmed employing the same techniques used to program the USER EPROM and KEY arrays using serial data streams and logic levels on port pins indicated in Table 6. When programming either security bit, it is not necessary to provide address or data information to the 87C751 on ports 1 and 3.

Verification occurs in a similar manner using the RESET serial stream shown in Table 6. Port 3 is not required to be driven and the results of the verify operation will appear on ports 1.6 and 1.7.

Ports 1.7 contains the security bit 1 data and is a logical one if programmed and a logical zero if erased. Likewise, P1.6 contains the security bit 2 data and is a logical one if programmed and a logical zero if erased.

## Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Flourless part number 2345–5 or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-s/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000µW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

**Table 6. Implementing Program/Verify Modes**

| OPERATION | SERIAL CODE | P0.1 (PGM/) | P0.2 ($V_{PP}$) |
|---|---|---|---|
| Program user EPROM | 296H | –* | $V_{PP}$ |
| Verify user EPROM | 296H | $V_{IH}$ | $V_{IH}$ |
| Program key EPROM | 292H | –* | $V_{PP}$ |
| Verify key EPROM | 292H | $V_{IH}$ | $V_{IH}$ |
| Program security bit 1 | 29AH | –* | $V_{PP}$ |
| Program security bit 2 | 298H | –* | $V_{PP}$ |
| Verify security bits | 29AH | $V_{IH}$ | $V_{IH}$ |

**NOTE:**
* Pulsed from $V_{IH}$ to $V_{IL}$ and returned to $V_{IH}$.

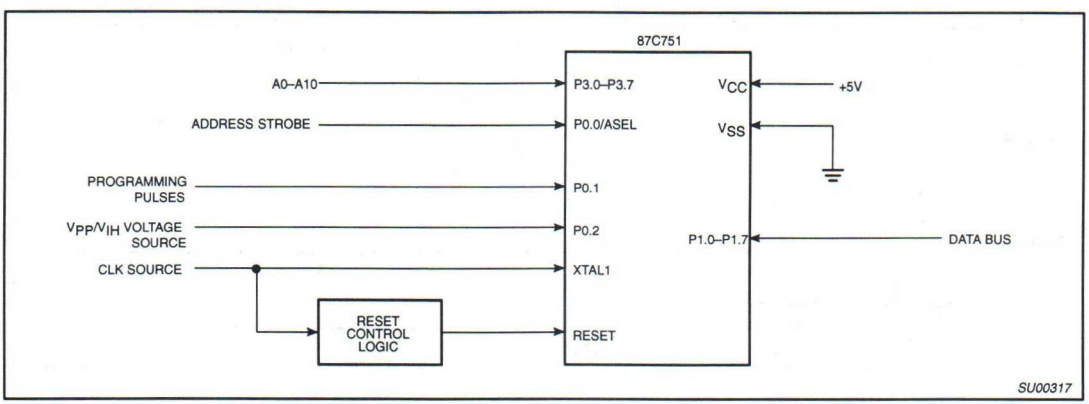



**Figure 5. Programming Configuration**

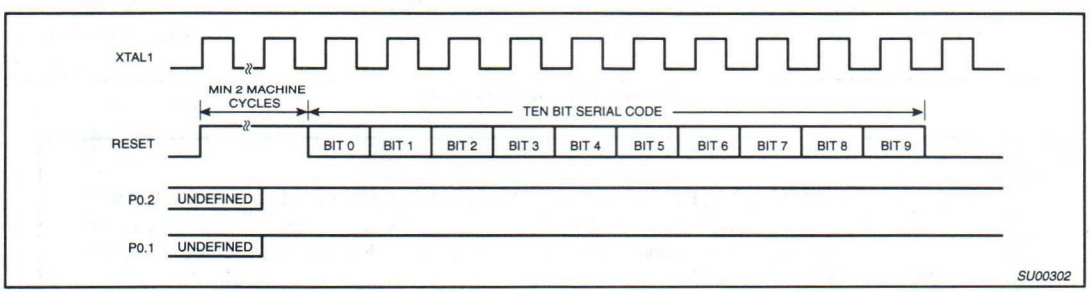



**Figure 6. Entry into Program/Verify Modes**

## EPROM PROGRAMMING AND VERIFICATION

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $1/t_{CLCL}$ | Oscillator/clock frequency | 1.2 | 6 | MHz |
| $t_{AVGL}$[1] | Address setup to P0.1 (PROG–) low | 10µs + 24$t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after P0.1 (PROG–) high | 48$t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to P0.1 (PROG–) low | 38$t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after P0.1 (PROG–) high | 36$t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to P0.1 (PROG–) low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after P0.1 (PROG–) | 10 | | µs |
| $t_{GLGH}$ | P0.1 (PROG–) width | 90 | 110 | µs |
| $t_{AVQV}$[2] | $V_{PP}$ low ($V_{CC}$) to data valid | | 48$t_{CLCL}$ | |
| $t_{GHGL}$ | P0.1 (PROG–) high to P0.1 (PROG–) low | 10 | | µs |
| $t_{MASEL}$ | ASEL high time | 13$t_{CLCL}$ | | |
| $t_{HAHLD}$ | Address hold time | 2$t_{CLCL}$ | | |
| $t_{HASET}$ | Address setup to ASEL | 13$t_{CLCL}$ | | |
| $t_{ADSTA}$ | Low address to valid data | | 48$t_{CLCL}$ | |

**NOTES:**
1. Address should be valid at least 24$t_{CLCL}$ before the rising edge of P0.2 ($V_{PP}$).
2. For a pure verify mode, i.e., no program mode in between, $t_{AVQV}$ is 14$t_{CLCL}$ maximum.

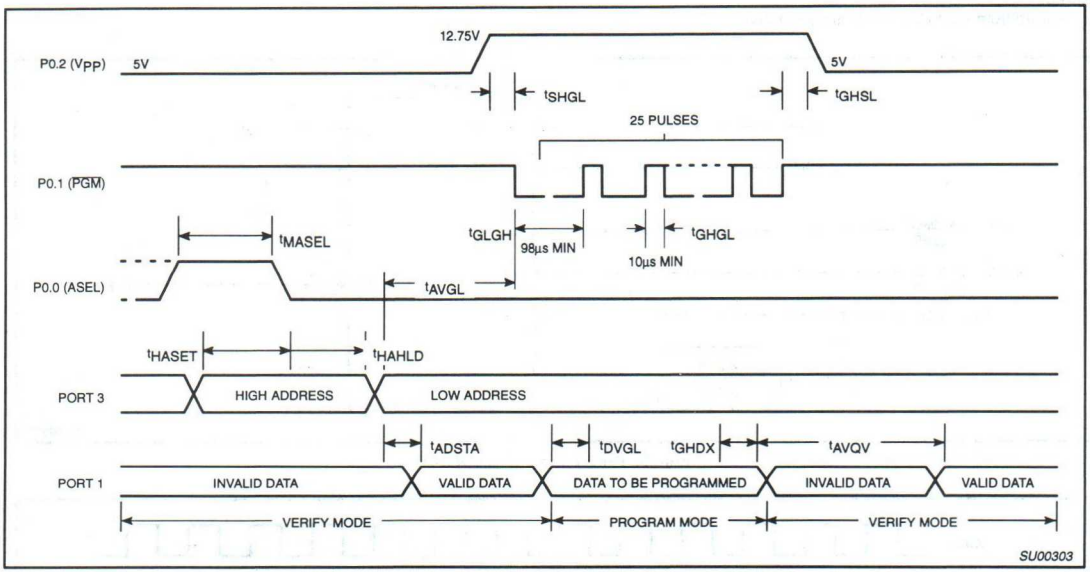



**Figure 7. Program/Verify Cycle**

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# CMOS single-chip 8-bit microcontrollers 83C752/87C752

*CMOS single-chip 8-bit microcontroller with A/D, PWM*

## DESCRIPTION

The Philips 83C752/87C752 offers many of the advantages of the 80C51 architecture in a small package and at low cost.

The 8XC752 Microcontroller is fabricated with Philips high-density CMOS technology. Philips epitaxial substrate minimizes CMOS latch-up sensitivity.

The 8XC752 contains a 2k × 8 ROM (83C752) EPROM (87C752), a 64 × 8 RAM, 21 I/O lines, a 16-bit auto-reload counter/timer, a fixed-priority level interrupt structure, a bidirectional inter-integrated circuit ($I^2C$) serial bus interface, an on-chip oscillator, a five channel multiplexed 8-bit A/D converter, and an 8-bit PWM output.

The onboard inter-integrated circuit ($I^2C$) bus interface allows the 8XC752 to operate as a master or slave device on the $I^2C$ small area network. This capability facilitates I/O and RAM expansion, access to EEPROM, processor-to-processor communication, and efficient interface to a wide variety of dedicated $I^2C$ peripherals.

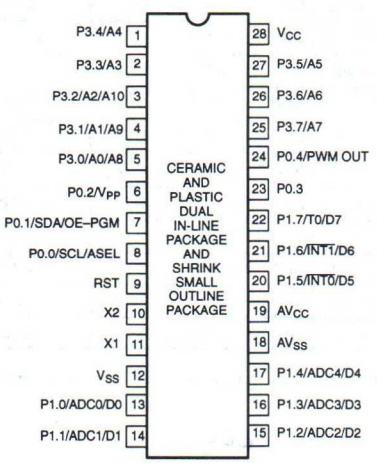
The EPROM version of this device, the 87C752, is also available in both quartz-lid erasable and plastic one-time programmable (OTP) packages. Once the array has been programmed, it is functionally equivalent to the masked ROM 83C752. Thus, unless explicitly stated otherwise, all references made to the 83C752 apply equally to the 87C752.

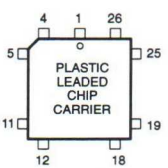
The 83C752 supports two power reduction modes of operation referred to as the idle mode and the power-down mode.

## FEATURES

- Available in erasable quartz lid or One-Time Programmable plastic packages
- 80C51 based architecture
- Inter-integrated Circuit ($I^2C$) serial bus interface
- Small package sizes
  - 28-pin DIP
  - 28-pin PLCC
  - 28-pin SSOP
- Wide oscillator frequency range
- Low power consumption:
  - Normal operation: less than 11mA @ 5V, 12MHz
  - Idle mode
  - Power-down mode
- 2k × 8 ROM (83C752)
  EPROM (87C752)
- 64 × 8 RAM
- 16-bit auto reloadable counter/timer
- 5-channel 8-bit A/D converter
- 8-bit PWM output/timer
- Fixed-rate timer
- Boolean processor
- CMOS and TTL compatible
- Well suited for logic replacement, consumer and industrial applications

## PIN CONFIGURATIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P3.4/A4 | 15 | P1.2/ADC2/D2 |
| 2 | P3.3/A3 | 16 | P1.3/ADC3/D3 |
| 3 | P3.2/A2/A10 | 17 | P1.4/ADC4/D4 |
| 4 | P3.1/A1/A9 | 18 | $AV_{SS}$ |
| 5 | P3.0/A0/A8 | 19 | $AV_{CC}$ |
| 6 | $P0.2/V_{PP}$ | 20 | P1.5/$\overline{INT0}$/D5 |
| 7 | P0.1/SDA/OE-PGM | 21 | P1.6/$\overline{INT1}$/D6 |
| 8 | P0.0/SCL/ASEL | 22 | P1.7/T0/D7 |
| 9 | RST | 23 | P0.3 |
| 10 | X2 | 24 | P0.4/PWM OUT |
| 11 | X1 | 25 | P3.7/A7 |
| 12 | $V_{SS}$ | 26 | P3.6/A6 |
| 13 | P1.0/ADC0/D0 | 27 | P3.5/A5 |
| 14 | P1.1/ADC1/D1 | 28 | $V_{CC}$ |

SU00318

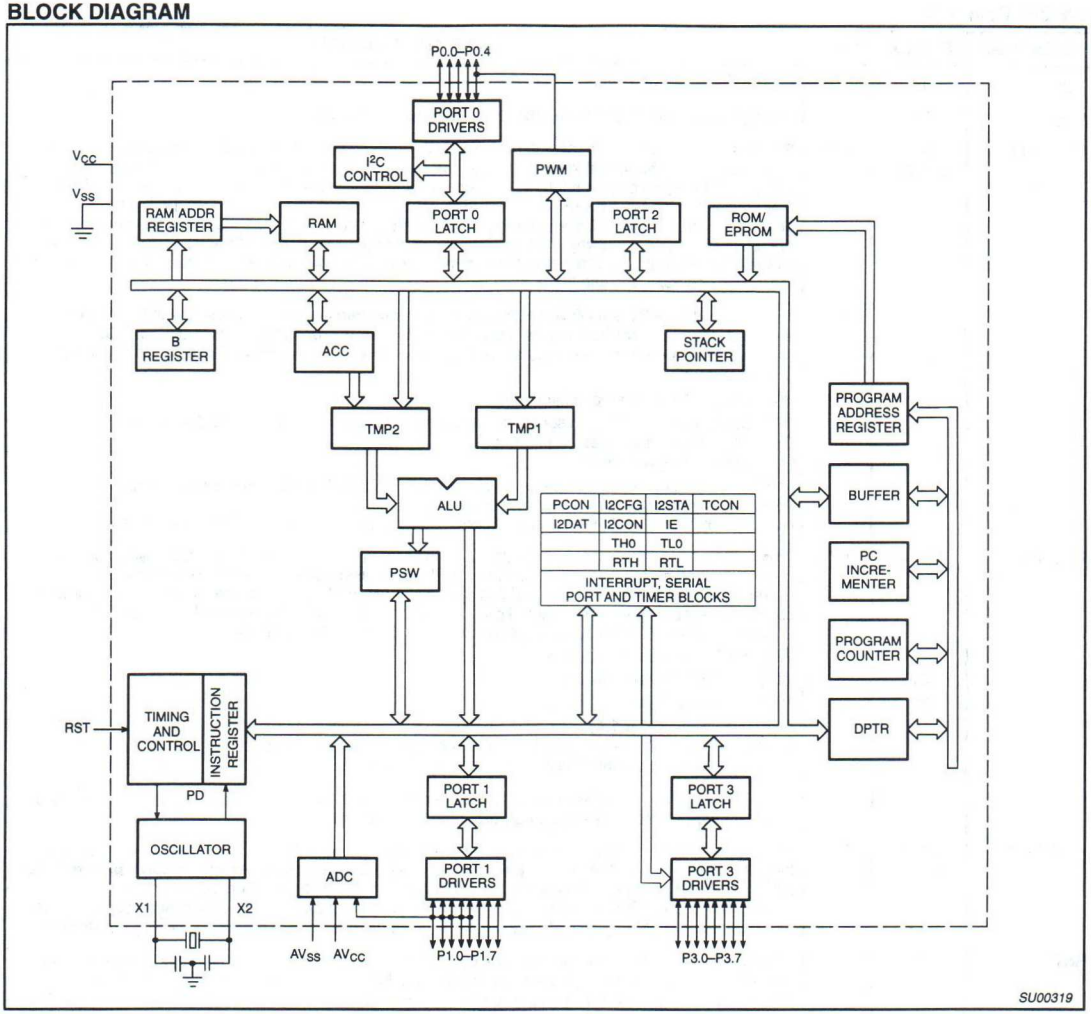
## PART NUMBER SELECTION

| ROM | EPROM | | TEMPERATURE RANGE °C AND PACKAGE | FREQUENCY | DRAWING NUMBER |
|---|---|---|---|---|---|
| | S87C752–1F28 | UV | 0 to +70, 28-pin Ceramic Dual In-line Package | 3.5 to 12MHz | 0589B |
| | S87C752–2F28 | UV | –40 to +85, 28-pin Ceramic Dual In-line Package | 3.5 to 12MHz | 0589B |
| | S87C752–4F28 | UV | 0 to +70, 28-pin Ceramic Dual In-line Package | 3.5 to 16MHz | 0589B |
| | S87C752–5F28 | UV | –40 to +85, 28-pin Ceramic Dual In-line Package | 3.5 to 16MHz | 0589B |
| S83C752–1DB | S87C752–1DB | OTP | 0 to +70, 28-pin Plastic Shrink Small Outline Package | 3.5 to 12MHz | SOT341-1 |
| S83C752–1N28 | S87C752–1N28 | OTP | 0 to +70, 28-pin Plastic Dual In-line Package | 3.5 to 12MHz | SOT117-2 |
| S83C752–2N28 | S87C752–2N28 | OTP | –40 to +85, 28-pin Plastic Dual In-line Package | 3.5 to 12MHz | SOT117-2 |
| S83C752–4DB | S87C752–4DB | OTP | 0 to +70, 28-pin Plastic Shrink Small Outline Package | 3.5 to 16MHz | SOT341-1 |
| S83C752–4N28 | S87C752–4N28 | OTP | 0 to +70, 28-pin Plastic Dual In-line Package | 3.5 to 16MHz | SOT117-2 |
| S83C752–5N28 | S87C752–5N28 | OTP | –40 to +85, 28-pin Plastic Dual In-line Package | 3.5 to 16MHz | SOT117-2 |
| S83C752–1A28 | S87C752–1A28 | OTP | 0 to +70, 28-pin Plastic Leaded Chip Carrier | 3.5 to 12MHz | SOT261-3 |
| S83C752–2A28 | S87C752–2A28 | OTP | –40 to +85, 28-pin Plastic Leaded Chip Carrier | 3.5 to 12MHz | SOT261-3 |
| S83C752–4A28 | S87C752–4A28 | OTP | 0 to +70, 28-pin Plastic Leaded Chip Carrier | 3.5 to 16MHz | SOT261-3 |
| S83C752–5A28 | S87C752–5A28 | OTP | –40 to +85, 28-pin Plastic Leaded Chip Carrier | 3.5 to 16MHz | SOT261-3 |
| S83C752–6A28 | S87C752–6A28 | OTP | –55 to +125, 28-pin Plastic Leaded Chip Carrier | 3.5 to 12MHz | SOT261-3 |
| S83C752–6N28 | S87C752–6N28 | OTP | –55 to +125, 28-pin Plastic Dual In-line Package | 3.5 to 12MHz | SOT117-2 |

**NOTE:**
1. OTP = One Time Programmable EPROM. UV = UV Erasable EPROM.

PLCC28 IN24

## BLOCK DIAGRAM



## PIN DESCRIPTION

| MNEMONIC | PIN NO. | TYPE | NAME AND FUNCTION |
|---|---|---|---|
| $V_{SS}$ | 12 | I | **Circuit Ground Potential.** |
| $V_{CC}$ | 28 | I | **Supply voltage during normal, idle, and power-down operation.** |
| P0.0–P0.4 | 8–6<br>23, 24 | I/O | **Port 0:** Port 0 is a 5-bit bidirectional port. Port 0.0–P0.2 are open drain. Port 0.0–P0.2 pins that have 1s written to them float, and in that state can be used as high-impedance inputs. P0.3–P0.4 are bidirectional I/O port pins with internal pull-ups. Port 0 also serves as the serial $I^2C$ interface. When this feature is activated by software, SCL and SDA are driven low in accordance with the $I^2C$ protocol. These pins are driven low if the port register bit is written with a 0 or if the $I^2C$ subsystem presents a 0. The state of the pin can always be read from the port register by the program. Port 0.3 and 0.4 have internal pull-ups that function identically to port 3. Pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs.<br><br>To comply with the $I^2C$ specification, P0.0 and P0.1 are open drain bidirectional I/O pins with the electrical characteristics listed in the tables that follow. While these differ from "standard TTL" characteristics, they are close enough for the pins to still be used as general-purpose I/O in non-$I^2C$ applications. |
| | 6 | I | **$V_{PP}$ (P0.2)** – Programming voltage input. (See Note 2.) |
| | 7 | I | **OE/PGM (P0.1)** – Input which specifies verify mode (output enable) or the program mode.<br>OE/PGM = 1 output enabled (verify mode).<br>OE/PGM = 0 program mode. |
| | 8 | I | **ASEL (P0.0)** – Input which indicates which bits of the EPROM address are applied to port 3.<br>ASEL = 0 low address byte available on port 3.<br>ASEL = 1 high address byte available on port 3 (only the three least significant bits are used). |
| P1.0–P1.7 | 13–17,<br>20–22 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. P0.3–P0.4 pins are bidirectional I/O port pins with internal pull-ups. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 1 also serves the special function features of the SC80C51 family as listed below: |
| | 20 | I | **$\overline{INT0}$ (P1.5):** External interrupt. |
| | 21 | I | **$\overline{INT1}$ (P1.6):** External interrupt. |
| | 22 | I | **T0 (P1.7):** Timer 0 external input. |
| | 13–17 | I | **ADC0 (P1.0)–ADC4 (P1.4):** Port 1 also functions as the inputs to the five channel multiplexed A/D converter. These pins can be used as outputs only if the A/D function has been disabled. These pins can be used as inputs while the A/D converter is enabled.<br><br>Port 1 serves to output the addressed EPROM contents in the verify mode and accepts as inputs the value to program into the selected address during the program mode. |
| P3.0–P3.7 | 5–1,<br>27–25 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also functions as the address input for the EPROM memory location to be programmed (or verified). The 11-bit address is multiplexed into this port as specified by P0.0/ASEL. |
| RST | 9 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on RESET using only an external capacitor to $V_{CC}$. After the device is reset, a 10-bit serial sequence, sent LSB first, applied to RESET, places the device in the programming state allowing programming address, data and $V_{PP}$ to be applied for programming or verification purposes. The RESET serial sequence must be synchronized with the X1 input. |
| X1 | 11 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. X1 also serves as the clock to strobe in a serial bit stream into RESET to place the device in the programming state. |
| X2 | 10 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |
| $AV_{CC}$ [1] | 19 | I | **Analog supply voltage and reference input.** |
| $AV_{SS}$ [1] | 18 | I | **Analog supply and reference ground.** |

**NOTE:**

1. $AV_{SS}$ (reference ground) must be connected to 0V (ground). $AV_{CC}$ (reference input) cannot differ from $V_{CC}$ by more than ±0.2V, and must be in the range 4.5V to 5.5V.
2. When P0.2 is at or close to 0V, it may affect the internal ROM operation. We recommend that P0.2 be tied to $V_{CC}$ via a small pull-up (e.g., 2kΩ).

## OSCILLATOR CHARACTERISTICS

X1 and X2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, X1 should be driven while X2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## IDLE MODE

The 8XC752 includes the 80C51 power-down and idle mode features. In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals except the A/D and PWM stay active. The functions that continue to run while in the idle mode are Timer 0, the $I^2C$ interface including Timer I, and the interrupts. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset. Upon powering-up the circuit, or exiting from idle mode, sufficient time must be allowed for stabilization of the internal analog reference voltages before an A/D conversion is started.

### Special Function Registers

The special function registers (directly addressable only) contain all of the 8XC751 registers except the program counter and the four register banks. Most of the 21 special function registers are used to control the on-chip peripheral hardware. Other registers include arithmetic registers (ACC, B, PSW), stack pointer (SP) and data pointer registers (DPH, DPL). Nine of the SFRs are bit addressable.

### Data Pointer

The data pointer (DPTR) consists of a high byte (DPH) and a low byte (DPL). In the 80C51 this register allows the access of external data memory using the MOVX instruction. Since the 83C752 does not support MOVX or external memory accesses, this register is generally used as a 16-bit offset pointer of the accumulator in a MOVC instruction. DPTR may also be manipulated as two independent 8-bit registers.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. The control bits for the reduced power modes are in the special function register PCON.

**Table 1. External Pin Status During Idle and Power-Down Modes**

| MODE | Port 0* | Port 1 | Port 2 |
|---|---|---|---|
| Idle | Data | Data | Data |
| Power-down | Data | Data | Data |

* Except for PWM output (P0.4).

## DIFFERENCES BETWEEN THE 8XC752 AND THE 80C51

### Program Memory

On the 8XC752, program memory is 2048 bytes long and is not externally expandable, so the 80C51 instructions MOVX, LJMP, and LCALL are not implemented. If these instructions are executed, the appropriate number of instruction cycles will take place along with external fetches; however, no operation will take place. The LJMP may not respond to all program address bits. The only fixed locations in program memory are the addresses at which execution is taken up in response to reset and interrupts, which are as follows:

| Event | Program Memory Address |
|---|---|
| Reset | 000 |
| External $\overline{INT0}$ | 003 |
| Counter/timer 0 | 00B |
| External $\overline{INT1}$ | 013 |
| Timer I | 01B |
| $I^2C$ serial | 023 |
| ADC | 02B |
| PWM | 033 |

### Memory Organization

The 8XC752 manipulates operands in three memory address spaces. The first is the program memory space which contains program instructions as well as constants such as look-up tables. The program memory space contains 2k bytes in the 8XC752.

The second memory space is the data memory array which has a logical address space of 128 bytes. However, only the first 64 (0 to 3FH) are implemented in the 8XC752.

The third memory space is the special function register array having a 128-byte address space (80H to FFH). Only selected locations in this memory space are used (see Table 2). Note that the architecture of these memory spaces (internal program memory, internal data memory, and special function registers) is identical to the 80C51, and the 8XC752 varies only in the amount of memory physically implemented.

The 8XC752 does not directly address any external data or program memory spaces. For this reason, the MOVX instructions in the 80C51 instruction set are not implemented in the 83C752, nor are the alternate I/O pin functions $\overline{RD}$ and $\overline{WR}$.

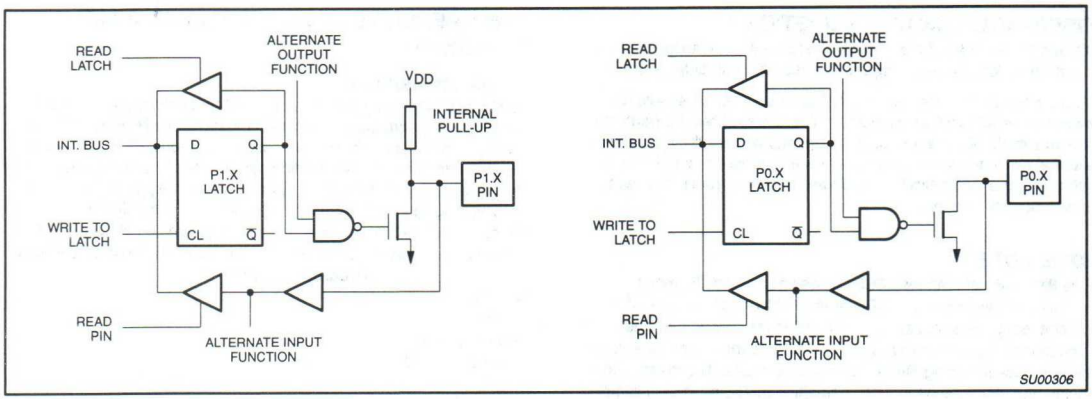



**Figure 1. Port Bit Latches and I/O Buffers**

## I/O Ports

The I/O pins provided by the 83C752 consist of port 0, port 1, and port 3.

### Port 0

Port 0 is a 5-bit bidirectional I/O port and includes alternate functions on some pins of this port. Pins P0.3 and P0.4 are provided with internal pullups while the remaining pins (P0.0, P0.1, and P0.2) have open drain output structures. The alternate functions for port 0 are:

P0.0 SCL – the $I^2C$ bus clock
P0.1 SDA – the $I^2C$ bus data
P0.4 PWM – the PWM output

If the alternate functions, $I^2C$ and PWM, are not being used, then these pins may be used as I/O ports.

### Port 1

Port 1 is an 8-bit bidirectional I/O port whose structure is identical to the 80C51, but also includes alternate input functions on all pins. The alternate pin functions for port 1 are:

P1.0-P1.4 - ADC0-ADC4 - A/D converter analog inputs
P1.5 $\overline{INT0}$ - external interrupt 0 input
P1.6 $\overline{INT1}$ - external interrupt 1 input
P1.7 - T0 - timer 0 external input

If the alternate functions $\overline{INT0}$, $\overline{INT1}$, or T0 are not being used, these pins may be used as standard I/O ports. It is necessary to connect $AV_{CC}$ and $AV_{SS}$ to $V_{CC}$ and $V_{SS}$, respectively, in order to use these pins as standard I/O pins. When the A/D converter is enabled, the analog channel connected to the A/D may not be used as a digital input; however, the remaining analog inputs may be used as digital inputs. They may not be used as digital outputs. While the A/D is enabled, the analog inputs are floating.

### Port 3

Port 3 is an 8-bit bidirectional I/O port whose structure is identical to the 80C51. Note that the alternate functions associated with port 3 of the 80C51 have been moved to port 1 of the 83C752 (as applicable). See Figure 1 for port bit configurations.

## Counter/Timer Subsystem

The 8XC752 has one counter/timer called timer/counter 0. Its operation is similar to mode 2 operation on the 80C51, but is extended to 16 bits with 16 bits of autoload. The controls for this counter are centralized in a single register called TCON.

A watchdog timer, called Timer I, is for use with the $I^2C$ subsystem. In $I^2C$ applications, this timer is dedicated to time-generation and bus monitoring of the $I^2C$. In non-$I^2C$ applications, it is available for use as a fixed time-base.

## Interrupt Subsystem—Fixed Priority

The IP register and the 2-level interrupt system of the 80C51 are eliminated. The interrupt structure is a seven-source, one-level interrupt system similar to the 8XC751. Simultaneous interrupt conditions are resolved by a single-level, fixed priority as follows:

| | |
|---|---|
| Highest priority: | Pin $\overline{INT0}$ |
| | Counter/timer flag 0 |
| | Pin $\overline{INT1}$ |
| | PWM |
| | Timer I |
| | Serial $I^2C$ |
| Lowest priority: | ADC |

The vector addresses are as follows:

| Source | Vector Address |
|---|---|
| INT0 | 0003H |
| TF0 | 000BH |
| INT1 | 0013H |
| TIMER I | 001BH |
| SIO | 0023H |
| ADC | 002BH |
| PWM | 0033H |

### Interrupt Control Registers

The 80C51 interrupt enable register is modified to take into account the different interrupt sources of the 8XC752.

**Interrupt Enable Register**

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| EA | EAD | ETI | ES | EPWM | EX1 | ET0 | EX0 |

| Position | Symbol | Function |
|---|---|---|
| IE.7 | EA | Global interrupt disable when EA = 0 |
| IE.6 | EAD | A/D conversion complete |
| IE.5 | ETI | Timer I |
| IE.4 | ES | $I^2C$ serial port |
| IE.3 | EPWM | PWM counter overflow |
| IE.2 | EX1 | External interrupt 1 |
| IE.1 | ET0 | Timer 0 overflow |
| IE.0 | EX0 | External interrupt 0 |

## Serial Communications

The 8XC752 contains an $I^2C$ serial communications port instead of the 80C51 UART. The $I^2C$ serial port is a single bit hardware interface with all of the hardware necessary to support multimaster and slave operations. Also included are receiver digital filters and timer (timer I) for communication watch-dog purposes. The $I^2C$ serial port is controlled through four special function registers; $I^2C$ control, $I^2C$ data, $I^2C$ status, and $I^2C$ configuration.

The $I^2C$ bus uses two wires (SDA and SCL) to transfer information between devices connected to the bus. The main technical features of the bus are:

- Bidirectional data transfer between masters and slaves
- Serial addressing of slaves
- Acknowledgment after each transferred byte
- Multimaster bus
- Arbitration between simultaneously transmitting master without corruption of serial data on bus
- With 82B715, communication distance is extended to beyond 100 feet (30M)

A large family of $I^2C$ compatible ICs is available. See the $I^2C$ section for more details on the bus and available ICs.

The 83C752 $I^2C$ subsystem includes hardware to simplify the software required to drive the $I^2C$ bus. This circuitry is the same as that on the 83C751. (See the 83C751 section for a detailed discussion of this subsystem).

## Pulse Width Modulation Output (P0.4)

The PWM outputs pulses of programmable length and interval. The repetition frequency is defined by an 8-bit prescaler which generates the clock for the counter. The prescaler register is PWMP. The prescaler and counter are not associated with any other timer. The 8-bit counter counts modulo 255, that is from 0 to 254 inclusive. The value of the 8-bit counter is compared to the contents of a compare register, PWM. When the counter value matches the contents of this register, the output of the PWM is set high. When the counter reaches zero, the output of the PWM is set low. The pulse width ratio (duty cycle) is defined by the contents of the compare register and is in the range of 0 to 1 programmed in increments of 1/255. The PWM output can be set to be continuously high by loading the compare register with 0 and the output can be set to be continuously low by loading the compare register with 255. The PWM output is enabled by a bit in a special function register, PWENA. When enabled, the pin output is driven with a fully active pull-up. That is, when the output is high, a strong pull-up is continuously applied. when disabled, the pin functions as a normal bidirectional I/O pin, however, the counter remains active.

The PWM function is disabled during RESET and remains disabled after reset is removed until re-enabled by software. The PWM output is high during power down and idle. The counter is disabled during idle. The repetition frequency of the PWM is given by:

$$f_{PWM} = f_{OSC} / 2 (1 + PWMP) 255$$

The low/high ratio of the PWM signal is PWM / (255 – PWM) for PWM not equal to 255. For PWM = 255, the output is always low.

The repetition frequency range is 92Hz to 23.5kHz for an oscillator frequency of 12MHz.

An interrupt will be asserted upon PWM counter overflow if the interrupt is not masked off.

The PWM output is an alternative function of P0.4. In order to use this port as a bidirectional I/O port, the PWM output must be disabled by clearing the enable/disable bit in PWENA. In this case, the PWM subsystem can be used as an interval timer by enabling the PWM interrupt.

**Table 2. 8XC752 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| ADAT# | A/D result | 84H | | | | | | | | | 00H |
| ADCON# | A/D control | A0H | – | – | ENADC | ADCI | ADCS | AADR2 | AADR1 | AADR0 | C0H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| DPTR: | Data pointer (2 bytes) | | | | | | | | | | |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| I²CFG*# | I²C configuration | D8H/RD | SLAVEN | MASTRQ | 0 | TIRUN | – | – | CT1 | CT0 | 0000xx00B |
| | | WR | SLAVEN | MASTRQ | CLRTI | TIRUN | – | – | CT1 | CT0 | |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| I²CON*# | I²C control | 98H/RD | RDAT | ATN | DRDY | ARL | STR | STP | MASTER | – | 81H |
| | | WR | CXA | IDLE | CDR | CARL | CSTR | CSTP | XSTR | XSTP | |
| I²DAT*# | I²C data | 99H/RD | RDAT | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 80H |
| | | WR | XDAT | X | X | X | X | X | X | X | |
| | | | FF | FE | FD | FC | FB | FA | F9 | F8 | |
| I²STA*# | I²C status | F8H | – | IDLE | XDATA | XACTV | MAKSTR | MAKSTP | XSTR | XSTP | x0100000B |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt enable | A8H | EA | EAD | ETI | ES | EPWM | EX1 | ET0 | EX0 | 00H |
| | | | – | – | – | 84 | 83 | 82 | 81 | 80 | xxx11111B |
| P0*# | Port 0 | 80H | – | – | – | PWM0 | – | – | SDA | SCL | |
| | | | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | FFH |
| P1*# | Port 1 | 90H | T0 | $\overline{INT1}$ | INT0 | ADC4 | ADC3 | ADC2 | ADC1 | ADC0 | |
| P3* | Port 3 | B0H | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | FFH |
| PCON# | Power control | 87H | – | – | – | – | – | – | PD | IDL | xxxx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| PWCM# | PWM compare | 8EH | | | | | | | | | xxxxxxxxB |
| PWENA# | PWM enable | FEH | – | – | – | – | – | – | – | PWE | FEH |
| PWMP# | PWM prescaler | 8FH | | | | | | | | | 00H |
| RTL# | Timer low reload | 8BH | | | | | | | | | 00H |
| RTH# | Timer high reload | 8DH | | | | | | | | | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| TL# | Timer low | 8AH | | | | | | | | | 00H |
| TH# | Timer high | 8CH | | | | | | | | | 00H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON*# | Timer control | 88H | GATE | C/T | TF | TR | IE0 | IT0 | IE1 | IT1 | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

## Special Function Register Addresses

Special function registers for the 8XC752 are identical to those of the 80C51, except for the changes listed below:

80C51 special function registers not present in the 8XC752 are TMOD (89), P2 (A0) and IP (B8). The 80C51 registers TH1, TL1, SCON, and SBUF are replaced with the 8XC752 registers RTH, RTL, I2CON, and I2DAT, respectively. Additional special function registers are I2CFG (D8) and I2STA (FB), ADCON (A0), ADAT (84), PWM (8E), PWMP (8F), and PWENA (FE). See Table 3.

## A/D Converter

The analog input circuitry consists of a 5-input analog multiplexer and an A to D converter with 8-bit resolution. The conversion takes 40 machine cycles, i.e., 40µs at 12MHz oscillator frequency. The A/D converter is controlled using the ADCON control register. Input channels are selected by the analog multiplexer through ADCON register bits 0–2.

The 83C752 contains a five-channel multiplexed 8-bit A/D converter. The conversion requires 40 machine cycles (40µs at 12MHz oscillator frequency).

The A/D converter is controlled by the A/D control register, ADCON. Input channels are selected by the analog multiplexer by bits ADCON.0 through ADCON.2. The ADCON register is not bit addressable.

### ADCON Register

MSB LSB

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| X | X | ENADC | ADCI | ADCS | AADR2 | AADR1 | AADR0 |

| ADCI | ADCS | Operation |
|---|---|---|
| 0 | 0 | ADC not busy, a conversion can be started. |
| 0 | 1 | ADC busy, start of a new conversion is blocked. |
| 1 | 0 | Conversion completed, start of a new conversion is blocked. |
| 1 | 1 | Not possible. |

| INPUT CHANNEL SELECTION | | | |
|---|---|---|---|
| ADDR2 | ADDR1 | ADDR0 | INPUT PIN |
| 0 | 0 | 0 | P1.0 |
| 0 | 0 | 1 | P1.1 |
| 0 | 1 | 0 | P1.2 |
| 0 | 1 | 1 | P1.3 |
| 1 | 0 | 0 | P1.4 |

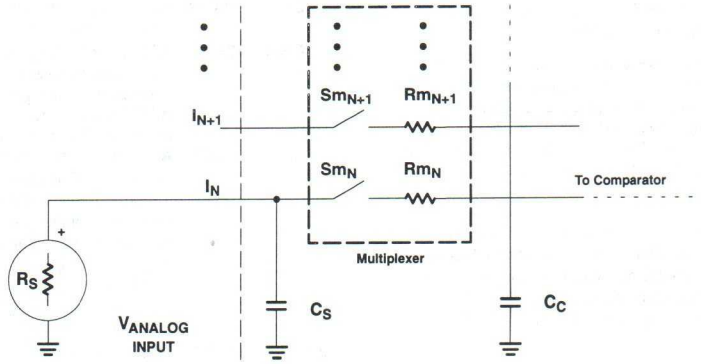
| Position | Symbol | Function |
|---|---|---|
| ADCON.5 | ENADC | Enable A/D function when ENADC = 1. Reset forces ENADC = 0. |
| ADCON.4 | ADCI | ADC interrupt flag. This flag is set when an ADC conversion is complete. If IE.6 = 1, an interrupt is requested when ADCI = 1. The ADCI flag is cleared when conversion data is read. This flag is read only. |
| ADCON.3 | ADCS | ADC start. Setting this bit starts an A/D conversion. Once set, ADCS remains high throughout the conversion cycle. On completion of the conversion, it is reset just before the ADCI interrupt flag is cleared. ADCS cannot be reset by software. ADCS should not be used to monitor the A/D converter status. ADCI should be used for this purpose. |
| ADCON.2 | AADR2 | Analog input select. |
| ADCON.1 | AADR1 | Analog input select. |
| ADCON.0 | AADR0 | Analog input select. This binary coded address selects one of the five analog input port pins of P1 to be input to the converter. It can only be changed when ADCI and ADCS are both low. AADR2 is the most significant bit. |

The completion of the 8-bit ADC conversion is flagged by ADCI in the ADCON register, and the result is stored in the special function register ADAT.

An ADC conversion in progress is unaffected by an ADC start. The result of a completed conversion remains unaffected provided ADCI remains at a logic 1. While ADCS is a logic 1 or ADCI is a logic 1, a new ADC START will be blocked and consequently lost. An ADC conversion in progress is aborted when the idle or power-down mode is entered. The result of a completed conversion (ADCI = logic 1) remains unaffected when entering the idle mode. See Figure 2 for an A/D input equivalent circuit.

The analog input pins ADC0-ADC4 may be used as digital inputs and outputs when the A/D converter is disabled by a 0 in the ENADC bit in ADCON. When the A/D is enabled, the analog input channel that is selected by the ADDR2-ADDR0 bits in ADCON cannot be used as a digital input. Reading the selected A/D channel as a digital input will always return a 1. The unselected A/D inputs may always be used as digital inputs. Unselected analog inputs will be floating and may not be used as digital outputs.

The A/D reference inputs on the 8XC752 are tied together with the analog supply pins $AV_{CC}$ and $AV_{SS}$. This means that the reference voltage on the A/D cannot be varied separately from the analog supply pins. $AV_{SS}$ must be connected to 0V and $AV_{CC}$ must be connected to a supply voltage between 4.5V and 5.5V. A/D measurements may be made in the range of 4.5V to 5.5V. Increasing the voltage on the A/D ground reference above 0V or reducing the voltage on the positive A/D reference below 4.5V is not permitted.




Rm = 0.5 - 3 kΩ
CS + CC = 15pF maximum
RS = Recommended < 9.6 kΩ for 1 LSB @ 12MHz

**NOTE:**
Because the analog to digital converter has a sampled-data comparator, the input looks capacitive to a source. When a conversion is initiated, switch Sm closes for 8tcy (8µs @ 12MHz crystal frequency) during which time capacitance Cs + Cc is charged. It should be noted that the sampling causes the analog input to present a varying load to an analog source.

SU00199

**Figure 2. A/D Input: Equivalent Circuit**

## A/D CONVERTER PARAMETER DEFINITIONS

The following definitions are included to clarify some specifications given and do not represent a complete set of A/D parameter definitions.

### Absolute Accuracy Error

Absolute accuracy error of a given output is the difference between the theoretical analog input voltage to produce a given output and the actual analog input voltage required to produce the same code. Since the same output code is produced by a band of input voltages, the "required input voltage" is defined as the midpoint of the band of input voltage that will produce that code. Absolute accuracy error not specified with a code is the maximum over all codes.

### Nonlinearity

If a straight line is drawn between the end points of the actual converter characteristics such that zero offset and full scale errors are removed, then non-linearity is the maximum deviation of the code transitions of the actual characteristics from that of the straight line so constructed. This is also referred to as relative accuracy and also integral non-linearity.

### Differential Non-Linearity

Differential non-linearity is the maximum difference between the actual and ideal code widths of the converter. The code widths are the differences expressed in LSB between the code transition points, as the input voltage is varied through the range for the complete set of codes.

### Gain Error

Gain error is the deviation between the ideal and actual analog input voltage required to cause the final code transition to a full-scale output code after the offset error has been removed. This may sometimes be referred to as full scale error.

### Offset Error

Offset error is the difference between the actual input voltage that causes the first code transition and the ideal value to cause the first code transition. This ideal value is 1/2 LSB above $V_{ref-}$.

### Channel to Channel Matching

Channel to channel matching is the maximum difference between the corresponding code transitions of the actual characteristics taken from different channels under the same temperature, voltage and frequency conditions.

### Crosstalk

Crosstalk is the measured level of a signal at the output of the converter resulting from a signal applied to one deselected channel.

### Total Error

Maximum deviation of any step point from a line connecting the ideal first transition point to the ideal last transition point.

### Relative Accuracy

Relative accuracy error is the deviation of the ADC's actual code transition points from the ideal code transition points on a straight line which connects the ideal first code transition point and the final code transition point, after nullifying offset error and gain error. It is generally expressed in LSBs or in percent of FSR.

## COUNTER/TIMER

The 8XC752 counter/timer is designated Timer 0 and is separate from Timer I of the I²C serial port and from the PWM. Its operation is similar to mode 2 of the 80C51 counter/timer, extended to 16 bits. When Timer 0 is used in the external counter mode, the T0 input (P1.7) is sampled every S4P1. The counter/timer function is controlled using the timer control register (TCON).

### TCON Register

MSB LSB

| GATE | C/T | TF | TR | IE0 | IT0 | IE1 | IT1 |
|---|---|---|---|---|---|---|---|

| Position | Symbol | Function |
|---|---|---|
| TCON.7 | GATE | 1 – Timer 0 is enabled only when INT0 pin is high and TR is 1. |
| | | 0 – Timer 0 is enabled only when TR is 1. |
| TCON.6 | C/T | 1 – Counter operation from T0 pin. |
| | | 0 – Timer operation from internal clock. |
| TCON.5 | TF | 1 – Set on overflow of T0. |
| | | 0 – Cleared when processor vectors to interrupt routine and by reset. |
| TCON.4 | TR | 1 – Enable timer 0 |
| | | 0 – Disable timer 0 |
| TCON.3 | IE0 | 1 – Edge detected on INT0 |
| TCON.2 | IT0 | 1 – INT0 is edge triggered. |
| | | 0 – INT0 is level sensitive. |
| TCON.1 | IE1 | 1 – Edge detected on INT1 |
| TCON.0 | IT1 | 1 – INT1 is edge triggered. |
| | | 0 – INT1 is level sensitive. |

These flags are functionally identical to the corresponding 80C51 flags except that there is only one of the 80C51 style timers, and the flags are combined into one register.

Note that the positions of the IE0/IT0 and IE1/IT1 bits are transposed from the positions used in the standard 80C51 TCON register.

A communications watchdog timer, Timer I, is described in the I²C section. In I²C applications, this timer is dedicated to time generation and bus monitoring for the I²C. In non-I²C applications, it is available for use as a fixed time base.

The 16-bit timer/counter's operation is similar to mode 2 operation on the 80C51, but is extended to 16 bits. The timer/counter is clocked by either 1/12 the oscillator frequency or by transitions on the T0 pin. The C/T pin in special function register TCON selects between these two modes. When the TCON TR bit is set, the timer/counter is enabled. Register pair TH and TL are incremented by the clock source. When the register pair overflows, the register pair is reloaded with the values in registers RTH and RTL. The value in the reload registers is left unchanged. The TF bit in special function register TCON is set on counter overflow and, if the interrupt is enabled, will generate an interrupt (see Figure 3).

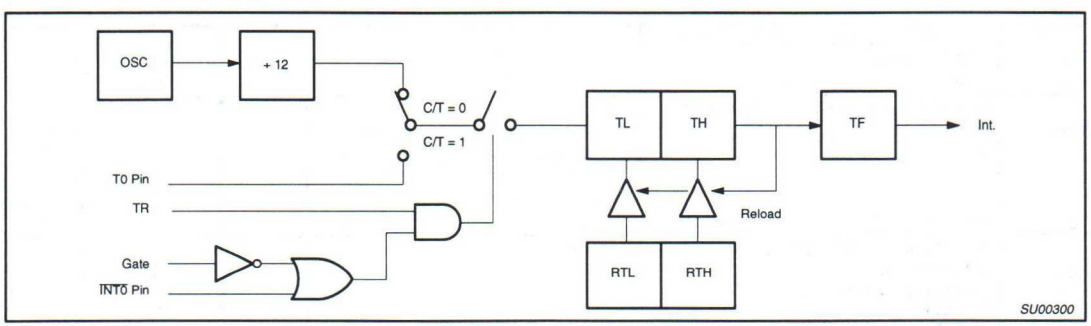



**Figure 3. 83C752 Counter/Timer Block Diagram**

## Table 3. I²C Special Function Register Addresses

| REGISTER ADDRESS | | | BIT ADDRESS | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| NAME | SYMBOL | ADDRESS | MSB | | | | | | | LSB |
| I²C control | I2CON | 98 | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 |
| I²C data | I2DAT | 99 | – | – | – | – | – | – | – | – |
| I²C configuration | I2CFG | D8 | DF | DE | DD | DC | DB | DA | D9 | D8 |
| I²C status | I2STA | F8 | FF | FE | FD | FC | FB | FA | F9 | F8 |

## ABSOLUTE MAXIMUM RATINGS[1, 3, 4]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage from $V_{CC}$ to $V_{SS}$ | –0.5 to +6.5 | V |
| Voltage from any pin to $V_{SS}$ (except $V_{PP}$) | –0.5 to $V_{CC}$ + 0.5 | V |
| Power dissipation | 1.0 | W |
| Voltage from $V_{PP}$ pin to $V_{SS}$ | –0.5 to + 13.0 | V |

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $AV_{CC}$ = 5V ±5, $AV_{SS}$ = 0V[4]
$V_{CC}$ = 5V ± 10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS[4] MIN | Typical[1] | MAX | UNIT |
|---|---|---|---|---|---|---|
| $I_{CC}$ | Supply current (see Figure 6) | | | | | |
| **Inputs** | | | | | | |
| $V_{IL}$ | Input low voltage, except SDA, SCL | (0 to 70°C)<br>(–40 to +85°C) | –0.5<br>–0.5 | | $0.2V_{CC}$–0.1<br>$0.2V_{CC}$–0.15 | V<br>V |
| $V_{IH}$ | Input high voltage, except X1, RST | (0 to 70°C)<br>(–40 to +85°C) | $0.2V_{CC}$+0.9<br>($0.2V_{CC}$+1) | | $V_{CC}$+0.5<br>$V_{CC}$+0.5 | V<br>V |
| $V_{IH1}$ | Input high voltage, X1, RST | (0 to 70°C)<br>(–40 to +85°C) | $0.7V_{CC}$<br>$0.7V_{CC}$ to 0.1 | | $V_{CC}$+0.5<br>$V_{CC}$+0.5 | V<br>V |
| | SDA, SCL, P0.2 | | | | | |
| $V_{IL1}$ | Input low voltage | (0 to 70°C)<br>(–40 to +85°C) | –0.5<br>–0.5 | | $0.3V_{CC}$<br>$0.3V_{CC}$–0.1 | V<br>V |
| $V_{IH2}$ | Input high voltage | (0 to 70°C)<br>(–40 to –85°C) | $0.7V_{CC}$<br>$0.7V_{CC}$+0.1 | | $V_{CC}$+0.5<br>$V_{CC}$+0.5 | V |
| **Outputs** | | | | | | |
| $V_{OL}$ | Output low voltage, ports 1, 3, 0.3, and 0.4 (PWM disabled) | $I_{OL}$ = 1.6mA[2] | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0.2 | $I_{OL}$ = 3.2mA[2] | | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1, 3, 0.3, and 0.4 (PWM disabled) | $I_{OH}$ = –60µA,<br>$I_{OH}$ = –25µA<br>$I_{OH}$ = –10µA<br>$I_{OH}$ = –400µA | 2.4<br>$0.75V_{CC}$<br>$0.9V_{CC}$<br>2.4 | | | V<br>V<br>V<br>V |
| $V_{OH2}$ | Output high voltage, P0.4 (PWM enabled) | $I_{OH}$ = –40µA | $0.9V_{CC}$ | | | V |
| $V_{OL2}$ | Port 0.0 and 0.1 ($I^2C$) – Drivers<br>Output low voltage<br>Driver, receiver combined: | $I_{OL}$ = 3mA<br>(over $V_{CC}$ range) | | | 0.4 | V |
| C | Capacitance | | | | 10 | pF |
| $I_{IL}$ | Logical 0 input current, ports 1, 3, 0.3, and 0.4 (PWM disabled)[11] | $V_{IN}$ = 0.45V (0 to 70°C)<br>$V_{IN}$ = 0.45V (0 to +85°C) | | | –50<br>–75 | µA<br>µA |
| $I_{TL}$ | Logical 1 to 0 transition current, ports 1, 3, 0.3 and 0.4[11] | $V_{IN}$ = 2V (0 to 70°C)<br>$V_{IN}$ = 2V (–40 to +85°C) | | | –650<br>–750 | µA<br>µA |
| $I_{LI}$ | Input leakage current, port 0.0, 0.1 and 0.2 | 0.45 < $V_{IN}$ < $V_{CC}$ | | | ±10 | µA |
| $R_{RST}$ | Reset pull-down resistor | | 25 | | 175 | kΩ |
| $C_{IO}$ | Pin capacitance | Test freq = 1MHz,<br>$T_{amb}$ = 25°C | | | 10 | pF |
| $I_{PD}$ | Power-down current[5] | $V_{CC}$ = 2 to 5.5V<br>$V_{CC}$ = 2 to 6.0V (83C752) | | | 50 | µA |
| $V_{PP}$ | $V_{PP}$ program voltage (87C752 only) | $V_{SS}$ = 0V<br>$V_{CC}$ = 5V±10%<br>$T_{amb}$ = 21°C to 27°C | 12.5 | | 13.0 | V |
| $I_{PP}$ | Program current (87C752 only) | $V_{PP}$ = 13.0V | | | 50 | mA |

## DC ELECTRICAL CHARACTERISTICS (Continued)

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $AV_{CC}$ = 5V ±5, $AV_{SS}$ = 0V[4]
$V_{CC}$ = 5V ± 10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS[4] | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | Typical[1] | MAX | |
| **Analog Inputs** (A/D guaranteed only with quartz window covered.) | | | | | | |
| $AV_{CC}$ | Analog supply voltage[10] | $AV_{CC} = V_{CC}$±0.2V | 4.5 | | 5.5 | V |
| $AI_{CC}$ | Analog operating supply current | $AV_{CC}$ = 5.12V | | | 3[9] | mA |
| $AV_{IN}$ | Analog input voltage[12] | | $AV_{SS}$–0.2 | | $AV_{CC}$+0.2 | V |
| $C_{IA}$ | Analog input capacitance | | | | 15 | pF |
| $t_{ADS}$ | Sampling time | | | | $8t_{CY}$ | s |
| $t_{ADC}$ | Conversion time | | | | $40t_{CY}$ | s |
| R | Resolution | | | | 8 | bits |
| $E_{RA}$ | Relative accuracy | | | | ±1 | LSB |
| $OS_e$ | Zero scale offset | | | | ±1 | LSB |
| $G_e$ | Full scale gain error | | | | 0.4 | % |
| $M_{CTC}$ | Channel to channel matching | | | | ±1 | LSB |
| $C_t$ | Crosstalk | 0–100kHz | | | –60 | dB |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 10mA (NOTE: This is 85°C spec.)
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 67mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
3. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
4. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.
5. Power-down $I_{CC}$ is measured with all output pins disconnected; port 0 = $V_{CC}$; X2, X1 n.c.; RST = $V_{SS}$.
6. $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; RST = port 0 = $V_{CC}$. $I_{CC}$ will be slightly higher if a crystal oscillator is used.
7. Idle $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; port 0 = $V_{CC}$; RST = $V_{SS}$.
8. Load capacitance for ports = 80pF.
9. The resistor ladder network is not disconnected in the power down or idle modes. Thus, to conserve power, the user may remove $AV_{CC}$.
10. If the A/D function is not required, or if the A/D function is only needed periodically, $AV_{CC}$ may be removed without affecting the operation of the digital circuitry. Contents of ADCON and ADAT are not guaranteed to be valid. If $AV_{CC}$ is removed, the A/D inputs must be lowered to less than 0.5V. Digital inputs on P1.0–P1.4 will not function normally.
11. These parameters do not apply to P1.0–P1.4 if the A/D function is enabled.
12. The input voltage slew rate should be <10V/ms. The maximum slew rate depends on the clock frequency of the microcontroller. Designers should use low pass filters before the A/D inputs as a precaution to noise edges causing false readings.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[4, 8]

| SYMBOL | PARAMETER | 12MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | Oscillator frequency: | | | 3.5<br>3.5 | 12<br>16 | MHz<br>MHz |
| **External Clock (Figure 4)** | | | | | | |
| $t_{CHCX}$ | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | Fall time | | 20 | | 20 | ns |

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal.
The designations are:

C – Clock
D – Input data
H – Logic level high
L – Logic level low
Q – Output data
T – Time
V – Valid
X – No longer a valid logic level
Z – Float

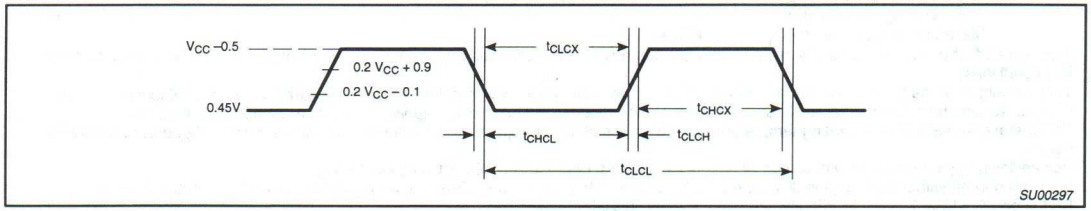



**Figure 4. External Clock Drive**

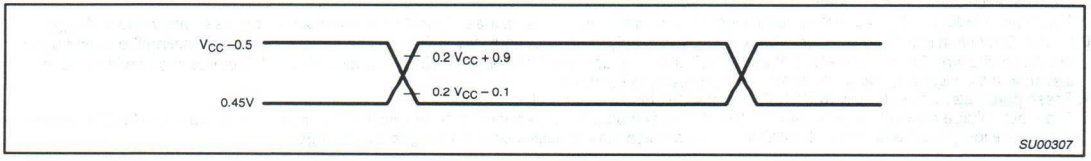



**Figure 5. AC Testing Input/Output**

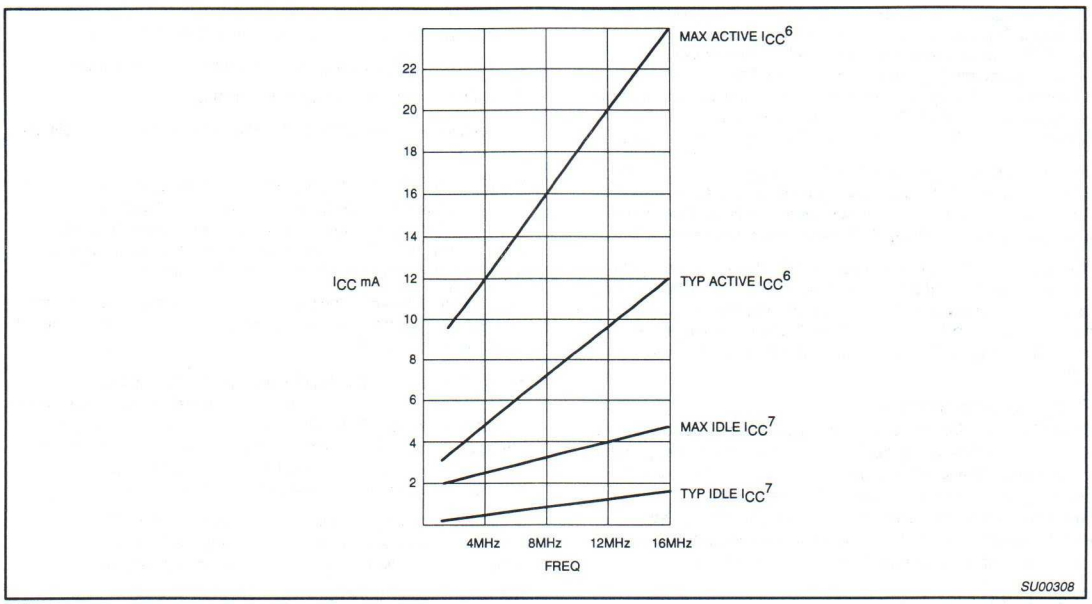



**Figure 6. $I_{CC}$ vs. FREQ**
**Maximum $I_{CC}$ values taken at $V_{CC}$ = 5.5V and worst case temperature.**
**Typical $I_{CC}$ values taken at $V_{CC}$ = 5.0V and 25°C.**
**Notes 6 and 7 refer to AC Electrical Characteristics.**

## PROGRAMMING CONSIDERATIONS

### EPROM Characteristics

The 87C752 is programmed by using a modified Quick-Pulse Programming algorithm similar to that used for devices such as the 87C451 and 87C51. It differs from these devices in that a serial data stream is used to place the 87C752 in the programming mode.

Figure 7 shows a block diagram of the programming configuration for the 87C752. Port pin P0.2 is used as the programming voltage supply input ($V_{PP}$ signal). Port pin P0.1 is used as the program (PGM/) signal. This pin is used for the 25 programming pulses.

Port 3 is used as the address input for the byte to be programmed and accepts both the high and low components of the eleven bit address. Multiplexing of these address components is performed using the ASEL input. The user should drive the ASEL input high and then drive port 3 with the high order bits of the address. ASEL should remain high for at least 13 clock cycles. ASEL may then be driven low which latches the high order bits of the address internally. The high address should remain on port 3 for at least two clock cycles after ASEL is driven low. Port 3 may then be driven with the low byte of the address. The low address will be internally stable 13 clock cycles later. The address will remain stable provided that the low byte placed on port 3 is held stable and ASEL is kept low. **Note:** ASEL needs to be pulsed high only to change the high byte of the address.

Port 1 is used as a bidirectional data bus during programming and verify operations. During programming mode, it accepts the byte to be programmed. During verify mode, it provides the contents of the EPROM location specified by the address which has been supplied to Port 3.

The XTAL1 pin is the oscillator input and receives the master system clock. This clock should be between 1.2 and 6MHz.

The RESET pin is used to accept the serial data stream that places the 87C752 into various programming modes. This pattern consists of a 10-bit code with the LSB sent first. Each bit is synchronized to the clock input, X1.

### Programming Operation

Figures 8 and 9 show the timing diagrams for the program/verify cycle. RESET should initially be held high for at least two machine cycles. P0.1 (PGM/) and P0.2 ($V_{PP}$) will be at $V_{OH}$ as a result of the RESET operation. At this point, these pins function as normal quasi-bidirectional I/O ports and the programming equipment may pull these lines low. However, prior to sending the 10-bit code on the RESET pin, the programming equipment should drive these pins high ($V_{IH}$). The RESET pin may now be used as the serial data input for the data stream which places the 87C752 in the programming mode. Data bits are sampled during the clock high time and thus should only change during the time that the clock is low. Following transmission of the last data bit, the RESET pin should be held low.

Next the address information for the location to be programmed is placed on port 3 and ASEL is used to perform the address multiplexing, as previously described. At this time, port 1 functions as an output.

A high voltage $V_{PP}$ level is then applied to the $V_{PP}$ input (P0.2). (This sets Port 1 as an input port). The data to be programmed into the EPROM array is then placed on Port 1. This is followed by a series of programming pulses applied to the PGM/ pin (P0.1). These pulses are created by driving P0.1 low and then high. This pulse is repeated until a total of 25 programming pulses have occurred. At the conclusion of the last pulse, the PGM/ signal should remain high.

The $V_{PP}$ signal may now be driven to the $V_{OH}$ level, placing the 87C752 in the verify mode. (Port 1 is now used as an output port). After four machine cycles (48 clock periods), the contents of the addressed location in the EPROM array will appear on Port 1.

The next programming cycle may now be initiated by placing the address information at the inputs of the multiplexed buffers, driving the $V_{PP}$ pin to the $V_{PP}$ voltage level, providing the byte to be programmed to Port1 and issuing the 26 programming pulses on the PGM/ pin, bringing $V_{PP}$ back down to the $V_C$ level and verifying the byte.

## Programming Modes

The 87C752 has four programming features incorporated within its EPROM array. These include the USER EPROM for storage of the application's code, a 16-byte encryption key array and two security bits. Programming and verification of these four elements are selected by a combination of the serial data stream applied to the RESET pin and the voltage levels applied to port pins P0.1 and P0.2. The various combinations are shown in Table 4.

## Encryption Key Table

The 87C752 includes a 16-byte EPROM array that is programmable by the end user. The contents of this array can then be used to encrypt the program memory contents during a program memory verify operation. When a program memory verify operation is performed, the contents of the program memory location is XNOR'ed with one of the bytes in the 16-byte encryption table. The resulting data pattern is then provided to port 1 as the verify data. The encryption mechanism can be disable, in essence, by leaving the bytes in the encryption table in their erased state (FFH) since the XNOR product of a bit with a logical one will result in the original bit. The encryption bytes are mapped with the code memory in 16-byte groups. the first byte in code memory will be encrypted with the first byte in the encryption table; the second byte in code memory will be encrypted with the second byte in the encryption table and so forth up to and including the 16the byte. The encryption repeats in 16-byte groups; the 17th byte in the code memory will be encrypted with the first byte in the encryption table, and so forth.

## Security Bits

Two security bits, security bit 1 and security bit 2, are provided to limit access to the USER EPROM and encryption key arrays. Security bit 1 is the program inhibit bit, and once programmed performs the following functions:

1. Additional programming of the USER EPROM is inhibited.
2. Additional programming of the encryption key is inhibited.
3. Verification of the encryption key is inhibited.
4. Verification of the USER EPROM and the security bit levels may still be performed.

(If the encryption key array is being used, this security bit should be programmed by the user to prevent unauthorized parties from reprogramming the encryption key to all logical zero bits. Such programming would provide data during a verify cycle that is the logical complement of the USER EPROM contents).

Security bit 2, the verify inhibit bit, prevents verification of both the USER EPROM array and the encryption key arrays. The security bit levels may still be verified.

## Programming and Verifying Security Bits

Security bits are programmed employing the same techniques used to program the USER EPROM and KEY arrays using serial data streams and logic levels on port pins indicated in Table 4. When programming either security bit, it is not necessary to provide address or data information to the 87C752 on ports 1 and 3.

Verification occurs in a similar manner using the RESET serial stream shown in Table 4. Port 3 is not required to be driven and the results of the verify operation will appear on ports 1.6 and 1.7.

Ports 1.7 contains the security bit 1 data and is a logical one if programmed and a logical zero if erased. Likewise, P1.6 contains the security bit 2 data and is a logical one if programmed and a logical zero if erased.

## Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Flourless part number 2345–5 or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-sec/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000uW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

**Table 4. Implementing Program/Verify Modes**

| OPERATION | SERIAL CODE | P0.1 (PGM/) | P0.2 ($V_{PP}$) |
|---|---|---|---|
| Program user EPROM | 296H | –* | $V_{PP}$ |
| Verify user EPROM | 296H | $V_{IH}$ | $V_{IH}$ |
| Program key EPROM | 292H | –* | $V_{PP}$ |
| Verify key EPROM | 292H | $V_{IH}$ | $V_{IH}$ |
| Program security bit 1 | 29AH | –* | $V_{PP}$ |
| Program security bit 2 | 298H | –* | $V_{PP}$ |
| Verify security bits | 29AH | $V_{IH}$ | $V_{IH}$ |

**NOTE:**
* Pulsed from $V_{IH}$ to $V_{IL}$ and returned to $V_{IH}$.

## EPROM PROGRAMMING AND VERIFICATION

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $1/t_{CLCL}$ | Oscillator/clock frequency | 1.2 | 6 | MHz |
| $t_{AVGL}$[1] | Address setup to P0.1 (PROG–) low | 10µs + $24t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after P0.1 (PROG–) high | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to P0.1 (PROG–) low | $38t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to P0.1 (PROG–) low | $38t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after P0.1 (PROG–) high | $36t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to P0.1 (PROG–) low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after P0.1 (PROG–) | 10 | | µs |
| $t_{GLGH}$ | P0.1 (PROG–) width | 90 | 110 | µs |
| $t_{AVQV}$[2] | $V_{PP}$ low ($V_{CC}$) to data valid | | $48t_{CLCL}$ | |
| $t_{GHGL}$ | P0.1 (PROG–) high to P0.1 (PROG–) low | 10 | | µs |
| $t_{SYNL}$ | P0.0 (sync pulse) low | $4t_{CLCL}$ | | |
| $t_{SYNH}$ | P0.0 (sync pulse) high | $8t_{CLCL}$ | | |
| $t_{MASEL}$ | ASEL high time | $13t_{CLCL}$ | | |
| $t_{MAHLD}$ | Address hold time | $2t_{CLCL}$ | | |
| $t_{HASET}$ | Address setup to ASEL | $13t_{CLCL}$ | | |
| $t_{ADSTA}$ | Low address to address stable | $13t_{CLCL}$ | | |

**NOTES:**
1. Address should be valid at least $24t_{CLCL}$ before the rising edge of P0.2 ($V_{PP}$).
2. For a pure verify mode, i.e., no program mode in between, $t_{AVQV}$ is $14t_{CLCL}$ maximum.

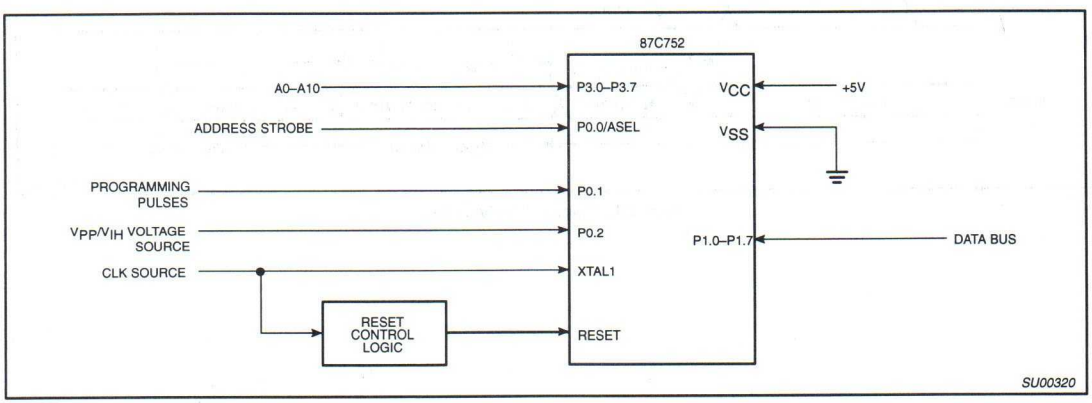



**Figure 7. Programming Configuration**

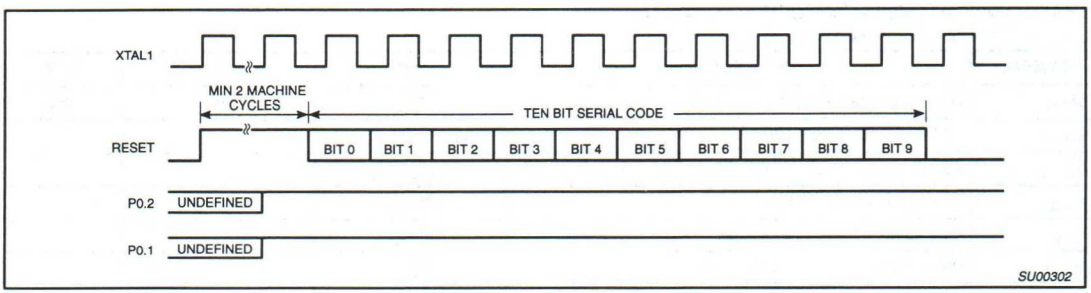



**Figure 8. Entry into Program/Verify Modes**

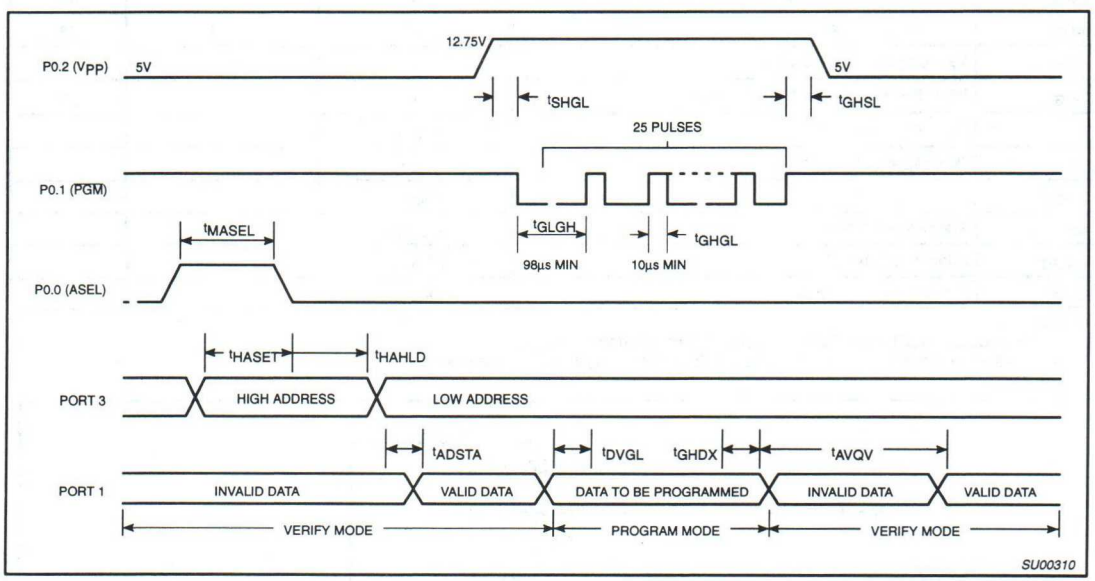



**Figure 9. Program/Verify Cycle**

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

# CMOS single-chip 8-bit microcontrollers 83C754/87C754

## DESCRIPTION

The Philips 83C754/87C754 offers many of the advantages of the 80C51 architecture in a small package and at low cost.

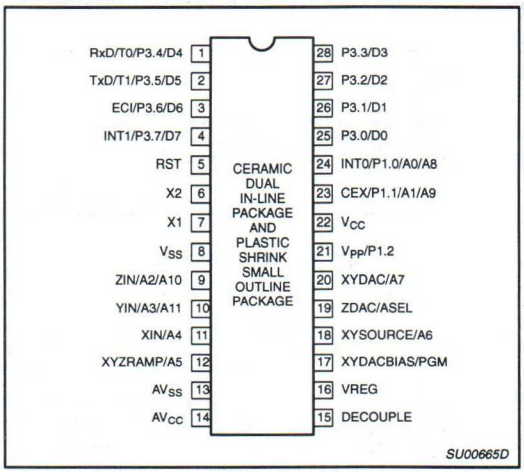
The 8XC754 Microcontroller is fabricated with Philips high-density CMOS technology. Philips epitaxial substrate minimizes CMOS latch-up senitivity.

The 8XC754 contains a 4k × 8 ROM (83C754) EPROM (87C754), a single module PCA, a 256 × 8 RAM, 11 I/O lines, two 16-bit counter/timers, a two-priority level interrupt structure, a full duplex serial channel, an on-chip oscillator, and an 8-bit D/A converter.

The EPROM version of this device, the 87C754, is also available in both quartz-lid erasable and plastic one-time programmable (OTP) packages. Once the array has been programmed, it ifs functionally equivalent to the masked ROM 83C754. Thus, unless explicitly stated otherwise, all references made to the 87C754 apply equally to the 83C754.

The 8XC754 supports two power reduction modes of operation referred to as the idle mode and the power-down mode.

## PIN CONFIGURATION




## FEATURES

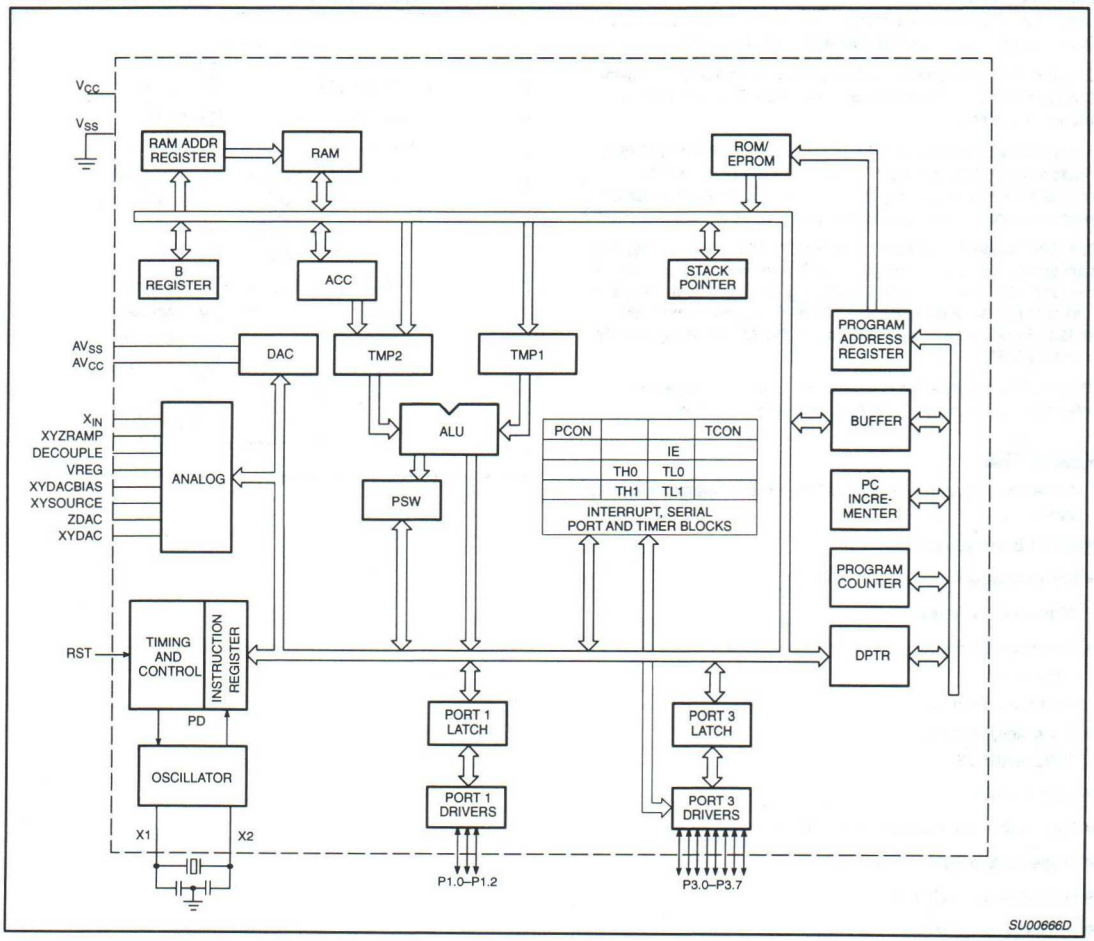
- Available in erasable quartz lid or One-Time Programmable plastic packages
- 80C51-based architecture
- Small package sizes – 28-pin SSOP
- Wide oscillator frequency range
- Power control modes:
  - Idle mode
  - Power-down mode
- 4k × 8 ROM (83C754)
  EPROM (87C754)
- 256 × 8 RAM
- Two 16-bit auto reloadable counter/timers
- Single module PCA counter/timer
- Full duplex serial channel
- Boolean processor
- CMOS and TTL compatible

## PART NUMBER SELECTION

| ROM | EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQUENCY | DRAWING NUMBER |
|---|---|---|---|---|---|
| | P87C754EBF FA | UV | 0 to +70, 28-pin Ceramic Dual In-line Package | 3.5 to 16MHz | 0589B |
| P83C754EBD DB | P87C754EBD DB | OTP | 0 to +70, 28-pin Shrink Small Outline Package | 3.5 to 16MHz | SOT341-1 |

**NOTE:**
1. OTP = One Time Programmable EPROM. UV = UV Erasable EPROM.

## BLOCK DIAGRAM



## PIN DESCRIPTION

| MNEMONIC | DIP PIN NO. | TYPE | NAME AND FUNCTION |
|---|---|---|---|
| $V_{SS}$ | 8 | I | **Circuit Ground Potential.** |
| $V_{CC}$ | 22 | I | **Supply voltage during normal, idle, and power-down operation.** |
| P1.0–P1.2 | 21, 23, 24 | I/O | **Port 1:** Port 1 is a 3-bit bidirectional I/O port with internal pull-ups on P1.0 and P1.1. Port 1 pins that have 1s written to them can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups (P1.0, P1.1). (See DC Electrical Characteristics: $I_{IL}$). Port 1 also serves the special function features listed below (Note: P1.0 does **not** have the strong pullup that is on for 2 oscillator periods.): |
| | 24 | I | **$\overline{INT0}$ (P1.0):** External interrupt 0. |
| | 23 | O | **CEX (P1.1):** PCA clock output. |
| | 21 | I | **$V_{PP}$ (P1.2):** Programming voltage input (open drain). |
| P3.0–P3.7 | 1–4, 25–28 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also functions as the data input for the EPROM memory location to be programmed (or verified). (Note: P3.5 does **not** have the strong pullup that is on for 2 oscillator periods.) |
| | | | Port 3 also serves the special function as listed below: |
| | 3 | I | **ECI (P3.6):** External PCA clock input. |
| | 1 | I | **RxD/T0 (P3.4):** Serial port receiver data input.<br>Timer 0 external clock input. |
| | 4 | I | **$\overline{INT1}$:** External interrupt 1. |
| | 2 | I | **TxD/T1 (P3.5):** Serial port transmitter data.<br>Timer 1 external clock input. |
| RST | 5 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running resets the device. After the device is reset, a 10-bit serial sequence, sent LSB first, applied to RESET, places the device in the programming state allowing programming address, data and $V_{PP}$ to be applied for programming or verification purposes. The RESET serial sequence must be synchronized with the X1 input. (Note: The 83/87C754 does not have an internal reset resistor.) |
| X1 | 7 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. X1 also serves as the clock to strobe in a serial bit stream into RESET to place the device in the programming state. |
| X2 | 6 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |
| $AV_{CC}$ [1] | 14 | I | **Analog supply voltage and reference input.** |
| $AV_{SS}$ [1] | 13 | I | **Analog supply and reference ground.** |
| ZIN | 9 | I | **ZIN:** Input to analog multiplexer. |
| YIN | 10 | I | **YIN:** Input to analog multiplexer. |
| XIN | 11 | I | **XIN:** Input to analog multiplexer. |
| XYZRAMP | 12 | O | **XYZRAMP:** Provides a low impedance pulldown to $V_{SS}$ under S/W control. |
| DECOUPLE | 15 | O | **Decouple:** Output from regulated supply for connection of decoupling capacitors. |
| VREG | 16 | O | **VREG:** Provides regulated analog supply output. |
| XYDACBIAS | 17 | O | **XYDACBIAS:** Provides source voltage for bias of external circuitry.<br>– Input which specifies verify mode (output enable) or the program mode.<br>/PGM = 1 output enabled (verify mode).<br>/PGM = 0 program mode. |
| XYSOURCE | 18 | O | **XYSOURCE:** Provides source voltage from regulated analog supply. |
| ZDAC | 19 | O | **ZDAC:** Switchable outp from the internal DAC. |
| | | | **ASEL (P0.0) –** Input which indicates which bits of the EPROM address are applied to port 3.<br>ASEL = 0 low address byte available on port 3.<br>ASEL = 1 high address byte available on port 3 (only the three least significant bits are used). |
| XYDAC | 20 | O | **XYDAC:** Non-switchable output from the internal DAC. |

**NOTE:**
1. $AV_{SS}$ (reference ground) must be connected to 0V (ground). $AV_{CC}$ (reference input) cannot differ from $V_{CC}$ by more than ±0.2V, and must be in the range 4.5V to 5.5V.

## OSCILLATOR CHARACTERISTICS

X1 and X2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, X1 should be driven while X2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## IDLE MODE

The 8XC754 includes the 80C51 power-down and idle mode features. In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset. Upon powering-up the circuit, or exiting from idle mode, sufficient time must be allowed for stabilization of the internal analog reference voltages before a D/A conversion is started.

### Special Function Registers

The special function registers (directly addressable only) contain all of the 8XC754 registers except the program counter and the four register banks. Most of the special function registers are used to control the on-chip peripheral hardware. Other registers include arithmetic registers (ACC, B, PSW), stack pointer (SP) and data pointer registers (DPH, DPL). Twelve of the SFRs are bit addressable.

### Data Pointer

The data pointer (DPTR) consists of a high byte (DPH) and a low byte (DPL). In the 80C51 this register allows the access of external data memory using the MOVX instruction. Since the 83C754 does not support MOVX or external memory accesses, this register is generally used as a 16-bit offset pointer of the accumulator in a MOVC instruction. DPTR may also be manipulated as two independent 8-bit registers.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. The control bits for the reduced power modes are in the special function register PCON.

**Table 1. External Pin Status During Idle and Power-Down Modes**

| MODE | Port 1 | Port 3 |
|---|---|---|
| Idle | Data | Data |
| Power-down | Data | Data |

## STANDARD SERIAL INTERFACE

The serial port is full duplex, meaning it can transmit and receive simultaneously. It is also receive-buffered, meaning it can commence reception of a second byte before a previously received byte has been read from the register. (However, if the first byte still has not been read by the time reception of the second byte is complete, one of the bytes will be lost.) The serial port receive and transmit registers are both accessed at Special Function Register SBUF. Writing to SBUF loads the transmit register, and reading SBUF accesses a physically separate receive register.

The serial port can operate in 4 modes:

**Mode 0:** Serial data enters and exits through RxD. TxD outputs the shift clock. 8 bits are transmitted/received (LSB first). The baud rate is fixed at 1/12 the oscillator frequency.

**Mode 1:** 10 bits are transmitted (through TxD) or received (through RxD): a start bit (0), 8 data bits (LSB first), and a stop bit (1). On Receive, the stop bit goes into RB8 in Special Function Register SCON. The baud rate is variable.

**Mode 2:** 11 bits are transmitted (through TxD) or received (through RxD): a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). On Transmit, the 9th data bit (TB8 in SCON) can be assigned the value of 0 or 1. Or, for example, the parity bit (P, in the PSW) could be moved into TB8. On Receive, the 9th data bit goes into RB8 in Special Function Register SCON, while the stop bit is ignored. The baud rate is programmable to either 1/32 or 1/64 the oscillator frequency.

**Mode 3:** 11 its are transmitted (through TxD) or received (through RxD): a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). In fact, Mode 3 is the same as Mode 2 in all respects except baud rate. the baud rate in Mode 3 is variable.

In all four modes, transmission is initiated by any instruction that uses SBUF as a destination register. Reception is initiated in Mode 0 by the condition RI = 0 and REN = 1. Reception is initiated in the other modes by the incoming start bit if REN = 1.

### Multiprocessor Communications

Modes 2 and 2 have a special provision for multiprocessor communications. In these modes, 9 data bits are received. The 9th one goes into RB8. Then comes a stop bit. The port can be programmed such that when the stop bit is received, the serial port interrupt will be activated only if RB8 = 1. This feature is enabled by setting bit SM2 in SCON. A way to use this feature in multiprocessor systems is as follows:

When the master processor wants to transmit a block of data to one of several slaves, it first sends out an address byte which identifies the target slave. An address byte differs from a data byte in that the 9th bit is 1 in an address byte and 9 in a data byte. With SM2 = 1, no slave will be interrupted by a data byte. An address byte, however, will interrupt all slaves, so that each slave can examine the received byte and see if it is being addressed. The addressed slave will clear its SM2 bit and prepare to receive the data bytes that will be coming. The slaves that were not being addressed leave their SM2s set, and go on about their business, ignoring the coming data bytes.

SM2 has no effect in Mode 0, and in Mode 1 can be used to check the validity of the stop bit. In a Mode 1 reception, if SM2 = 1, the receive interrupt will not be activated unless a valid stop bit is received.

## Serial Port Control Register

The serial port control and status register is the Special Function Register SCON, shown in Figure 1. This register contains not only the mode selection bits, but also the 9th data bit for transmit and receive (TB8 and RB8), and the serial port interrupt bits (TI and RI).

## Baud Rates

The baud rate in Mode 0 is fixed: Mode 0 Baud Rate = Oscillator Frequency / 12. The baud rate in Mode 2 depends on the value of bit SMOD in Special function Register PCON. If SMOD = 0 (which is the value on reset), the baud rate is 1/64 the oscillator frequency. If SMOD = 1, the baud rate is 1/32 the oscillator frequency.

$$\text{Mode 2 Baud Rate} = \frac{2^{SMOD}}{64} \times (\text{Oscillator Frequency})$$

In the 8XC754, the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate.

## Using Timer 1 to Generate Baud Rates

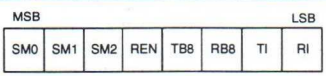
When Timer 1 is used as the baud rate generator, the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate and the value of SMOD as follows:

$$\text{Mode 1, 3 Baud Rate} = \frac{2^{SMOD}}{32} \times (\text{Timer 1 Overflow Rate})$$

The Timer 1 interrupt should be disabled in this application. The Timer itself can be configured for either "timer" or "counter" operation, and in any of its 3 running modes. In the most typical applications, it is configured for "timer" operation, in the auto-reload mode (high nibble of TMOD = 0010B). In that case the baud rate is given by the formula:

$$\text{Mode 1, 3 Baud Rate} = \frac{2^{SMOD}}{32} \times \frac{\text{Oscillator Frequency}}{12 \times [256 - (TH1)]}$$

One can achieve very low baud rates with Timer 1 by leaving the Timer 1 interrupt enabled, and configuring the Timer to run as a 16-bit timer (high nibble of TMOD = 0001B), and using the Timer 1 interrupt to do a 16-bit software reload. Figure 2 lists various commonly used baud rates and how they can be obtained from Timer 1.




Where SM0, SM1 specify the serial port mode, as follows:

| SM0 | SM1 | Mode | Description | Baud Rate |
|---|---|---|---|---|
| 0 | 0 | 0 | shift register | $f_{OSC}$/ 12 |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{OSC}$/64 or $f_{OSC}$/32 |
| 1 | 1 | 3 | 9-bit UART | variable |

**SM2** Enables the multiprocessor communication feature in Modes 2 and 3. In Mode 2 or 3, if SM2 is set to 1, then RI will not be activated if the received 9th data bit (RB8) is 0. In Mode 1, if SM2=1 then RI will not be activated if a valid stop bit was not received. In Mode 0, SM2 should be 0.

**REN** Enables serial reception. Set by software to enable reception. Clear by software to disable reception.

**TB8** The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired.

**RB8** In Modes 2 and 3, is the 9th data bit that was received. In Mode 1, it SM2=0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used.

**TI** Transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software.

**RI** Receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software.

*SU00120*

**Figure 1.**

| Baud Rate | $f_{OSC}$ | SMOD | Timer 1 | | |
|---|---|---|---|---|---|
| | | | C/T | Mode | Reload Value |
| Mode 0 Max: 1.67MHz | 20MHz | X | X | X | X |
| Mode 2 Max: 625k | 20MHz | 1 | X | X | X |
| Mode 1, 3 Max: 104.2k | 20MHz | 1 | 0 | 2 | FFH |
| 19.2k | 11.059MHz | 1 | 0 | 2 | FDH |
| 9.6k | 11.059MHz | 0 | 0 | 2 | FDH |
| 4.8k | 11.059MHz | 0 | 0 | 2 | FAH |
| 2.4k | 11.059MHz | 0 | 0 | 2 | F4H |
| 1.2k | 11.059MHz | 0 | 0 | 2 | E8H |
| 137.5 | 11.986MHz | 0 | 0 | 2 | 1DH |
| 110 | 6MHz | 0 | 0 | 2 | 72H |
| 110 | 12MHz | 0 | 0 | 1 | FEEBH |

**Figure 2. Timer 1 Generated Commonly Used Baud Rates**

## DIFFERENCES BETWEEN THE 8XC754 AND THE 80C51

### Program Memory

On the 8XC754, program memory is 4096 bytes long and is not externally expandable, so the 80C51 instructions MOVX, LJMP, and LCALL are not implemented. If these instructions are executed, the appropriate number of instruction cycles will take place along with external fetches; however, no operation will take place. The LJMP may not respond to all program address bits. The only fixed locations in program memory are the addresses at which execution is taken up in response to reset and interrupts, which are as follows:

| | Program Memory |
|---|---|
| **Event** | **Address** |
| Reset | 000 |
| External $\overline{INT0}$ | 003 |
| Timer 0 | 00B |
| External $\overline{INT1}$ | 013 |
| PCA | 01B |
| SIO/TF1 | 023 |

### Memory Organization

The 8XC754 manipulates operands in three memory address spaces. The first is the program memory space which contains program instructions as well as constants such as look-up tables. The program memory space contains 4k bytes in the 8XC754.

The second memory space is the data memory array which has a logical address space of 256 bytes.

The third memory space is the special function register array having a 128-byte address space (80H to FFH). Only selected locations in this memory space are used (see Table 2). Note that the architecture of these memory spaces (internal program memory, internal data memory, and special function registers) is identical to the 80C51, and the 8XC754 varies only in the amount of memory physically implemented.

The 8XC754 does not directly address any external data or program memory spaces. For this reason, the MOVX instructions in the 80C51 instruction set are not implemented in the 83C754, nor are the alternate I/O pin functions $\overline{RD}$ and $\overline{WR}$.

### I/O Ports

The I/O pins provided by the 8XC754 consist of port 1 and port 3.

**Port 1**

Port 1 is a 3-bit bidirectional I/O port and includes alternate functions on some pins of this port. P1.1 is provided with internal pullups while the remaining pins (P1.0 and P1.2) are an open drain output structure. The alternate functions for port 1 are:

INT0 – External interrupt 0.
PCAOUT – PCA clock output
$V_{PP}$ – External programming voltage.

**Port 3**

Port 3 is an 8-bit bidirectional I/O port structure. P3.5 is open drain.

The alternate functions for port 3 are:

RxD – Serial port receiver data input.
T1 – Timer 1 external clock input.
$\overline{INT1}$ – External interrupt 1.
TxD – Serial port transmitter data.
T0 – Timer 0 external clock input.
ECI – PCA external clock input.

### Analog Section

The analog section of the 8XC754, shown in Figure 3, consists of four major elements: a bandgap referenced voltage regulator, an 8-bit DAC, an input multiplexer and comparator, and a low impedance pulldown device.

The bandgap voltage regulator uses the $AV_{CC}$ pin as its supply and produces a regulated output on the VREG pin. The bandgap reference is enabled/disabled by AC0. The regulator also supplies the analog supply voltage for the DAC. The regulator may be switched on/off by means of the AC1 bit in the analog control register (ACON0). The regulator output may also be supplied to the XYDACBIAS and XYSOURCE pins by means of bits AC3 and AC4, respectively. The DECOUPLE pin is provided for decoupling the regulator output.

The DAC is an 8-bit device and its output appears on the XYDAC pin. In addition, the DAC output may also be routed to the ZDAC pin by means of bit AC6 in the ACON0 register. The DAC output is not buffered, so external load impedances should be taken into consideration when using either of these outputs.

A 3-input multiplexer is provided, whose output is connected to the positive reference of a comparator. The multiplexer output is controlled by bits MUX2:0 of ACON1. A bandgap reference supplies the negative reference of the comparator. The output of the comparator may be used the trigger the capture input of PCA module.

A low impedance pulldown is supplied at the XYZRAMP pin and is controlled by bit AC5 of ACON0.

### Interrupt Subsystem—Fixed Priority

The interrupt structure is a seven-source, two-level interrupt system. Simultaneous interrupt conditions are resolved by a single-level, fixed priority as follows:

| | |
|---|---|
| Highest priority: | Pin $\overline{INT0}$ |
| | Timer flag 0 |
| | Pin $\overline{INT1}$ |
| | PCA |
| Lowest priority: | Serial I/O – TF1 |

The vector addresses are as follows:

| Source | Vector Address |
|---|---|
| INT0 | 0003H |
| TF0 | 000BH |
| INT1 | 0013H |
| PCA | 001BH |
| SIO/TF1 | 0023H |

**Interrupt Enable Register**

MSB LSB

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| EA | – | – | ES/T1 | EC | EX1 | ET0 | EX0 |

| Position | Symbol | Function |
|---|---|---|
| IE.7 | EA | Global interrupt disable when EA = 0 |
| IE.6 | – | |
| IE.5 | – | |
| IE.4 | ES/T1 | Serial port/Timer Flag 1 |
| IE.3 | EC | PCA interrupt |
| IE.2 | EX1 | External interrupt 1 |
| IE.1 | ET0 | Timer 0 overflow |
| IE.0 | EX0 | External interrupt 0 |

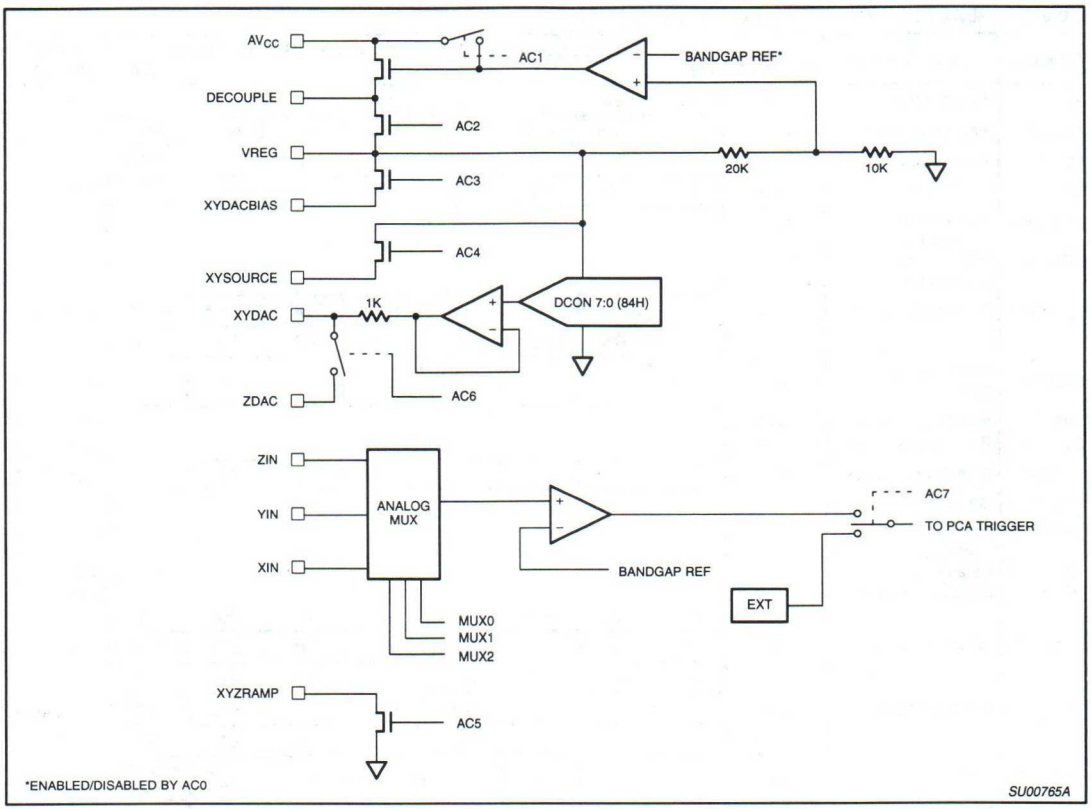



**Figure 3. Analog Section**

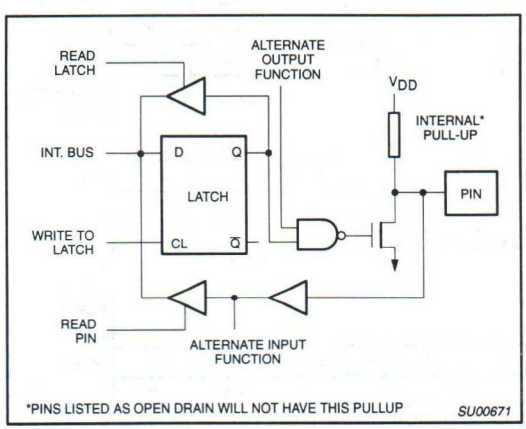



**Figure 4. Typical Port Bit Latches and I/O Buffers**

**Table 2. 8XC754 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| ACON0* | Analog Control 0 | A0H | AC7 | AC6 | AC5 | AC4 | AC3 | AC2 | AC1 | AC0 | |
| ACON1* | Analog Control 1 | C0H | – | – | – | – | TSI | MUX2 | MUX1 | MUX0 | |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| CCAPH# | PCA Module Capture High | FEH | | | | | | | | | |
| CCAPL# | PCA Module Capture Low | EEH | | | | | | | | | |
| CCAPM# | PCA Module Mode | DEH | – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | x0000000B |
| | | | DF | DE | DD | DC | DB | DA | D9 | D8 | |
| CCON*# | PCA Counter Control | D8H | CF | CR | – | CCF4 | – | – | – | – | 00x00000B |
| CH# | PCA Counter High | F9H | | | | | | | | | 00H |
| CL# | PCA Counter Low | E9H | | | | | | | | | 00H |
| CMOD# | PCA Counter Mode | D9H | CODL | WDTE | – | – | – | CPS1 | CPS0 | ECF | 00xxx000B |
| DCON | DAC Control | 84H | | | | | | | | | |
| DPTR: | Data pointer (2 bytes) | | | | | | | | | | |
| DPL | Data pointer low | 82H | | | | | | | | | 00H |
| DPH | Data pointer high | 83H | | | | | | | | | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE*# | Interrupt Enable | A8H | EA | – | – | ES/T1 | EC | EX1 | ET0 | EX0 | 00H |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IP* | Interrupt Priority | B8H | – | – | – | PS/T1 | PPC | PX1 | PT0 | PX0 | x0000000B |
| | | | – | – | – | 84 | 83 | 82 | 81 | 80 | xxx11111B |
| P1*# | Port 1 | 90H | – | – | – | – | – | ZIN | XYZRAMP | XYSOURCE | |
| P3*# | Port 3 | B0H | INT1 | ECI | TxD | RxD | – | – | – | – | |
| PCON | Power control | 87H | SMOD1 | SMOD0 | – | POF | GF1 | GF0 | PD | IDL | 00xxxx00B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| SBUF | Serial Data Buffer | 99H | | | | | | | | | xxxxxxxxB |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial Control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | |
| TCON* | Timer Control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| TH0 | Timer High 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer High 1 | 8DH | | | | | | | | | 00H |
| TL0 | Timer Low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer Low 1 | 8BH | | | | | | | | | 00H |
| TMOD | Timer Mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |
| | | | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 | |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

## COUNTER/TIMER

The 8XC754 counter/timers are designated Timer 0 and 1. They are identical to the 80C51 counter/timers. (Timer 1 shares its interrupt with the serial port.)

## Programmable Counter Array (PCA)

The Programmable Counter Array is a special Timer that has one 16-bit capture/compare module associated with it. The module can be programmed to operate in one of four modes: rising and/or falling edge capture, software timer, high-speed output, or pulse width modulator. The basic PCA configuration is shown in Figure 5.

The PCA timer can be programmed to run at: 1/12 the oscillator frequency, 1/4 the oscillator frequency, the Timer 0 overflow, or the input on the ECI pin (P3.1). The timer count source is determined from the CPS1 and CPS0 bits in the CMOD SFR as follows (see Figure 8):

| CPS1 | CPS0 | PCA Timer Count Source |
|---|---|---|
| 0 | 0 | 1/12 oscillator frequency |
| 0 | 1 | 1/4 oscillator frequency |
| 1 | 0 | Timer 0 overflow |
| 1 | 1 | External Input at ECI pin |

In the CMOD SFR are three additional bits associated with the PCA. They are CIDL which allows the PCA to stop during idle mode, WDTE which enables or disables the watchdog function, and ECF which when set causes an interrupt and the PCA overflow flag, CF (in the CCON SFR) to be set when the PCA timer overflows. These functions are shown in Figure 6.

The watchdog timer function is implemented in module 4 as implemented in other parts that have a PCA that are available on the market.

The CCON SFR contains the run control bit for the PCA and the flags for the PCA timer (CF) and module (refer to Figure 9). To run the PCA the CR bit (CCON.6) must be set by software. The PCA is shut off by clearing this bit. The CF bit (CCON.7) is set when the PCA counter overflows and an interrupt will be generated if the ECF bit in the CMOD register is set, The CF bit can only be cleared by software. Bit 4 of the CCON register is the flag for the module and is set by hardware when either a match or a capture occurs. This flag can only be cleared by software. The PCA interrupt system shown in Figure 7.

The CCAPM register contains the bits that control the mode in which the module will operate. The ECCF bit enables the CCF flag in the CCON SFR to generate an interrupt when a match or compare occurs in the associated module. PWM (CCAPM.1) enables the pulse width modulation mode. The TOG bit (CCAPM.2) when set causes the CEX output associated with the module to toggle when there is a match between the PCA counter and the module's capture/compare register. The match bit MAT (CCAPM.3), when set, will cause the CCF bit in the CCON register to be set when there is a match between the PCA counter and the module's capture/compare register.

The next two bits CAPN (CCAPM.4) and CAPP (CCAPM.5) determine the edge that a capture input will be active on. The CAPN bit enables the negative edge, and the CAPP bit enables the positive edge. If both bits are set both edges will be enabled and a capture will occur for either transition. The last bit in the register ECOM (CCAPM.6) when set enables the comparator function. Figure 11 shows the CCAPM settings for the various PCA functions.

There are two additional registers associated with the PCA module. They are CCAPH and CCAPL and these are the registers that store the 16-bit count when a capture occurs or a compare should occur. When the module is used in the PWM mode these registers are used to control the duty cycle of the output.

### PCA Capture Mode

To use the PCA module in the capture mode, either one or both of the CCAPM bits CAPN and CAPP must be set. The external CEX input for the module is sampled for transition. When a valid transition occurs, the PCA hardware loads the value of the PCA counter registers (CH and CL) into the module's capture registers (CCAPL and CCAPH). If the CCF bit for the module in the CCON SFR and the ECCF bit in the CCAPM SFR are set, then an interrupt will be generated. Refer to Figure 12.

### 16-bit Software Timer Mode

The PCA modules can be used as software timers by setting both the ECOM and MAT bits in the module's CCAPM register. The PCA timer will be compared to the module's capture registers and when a match occurs an interrupt will occur if the CCF (CCON SFR) and the ECCF (CCAPM SFR) bits for the module are both set (see Figure 13).

### High Speed Output Mode

In this mode the CEX output associated with the PCA module will toggle each time a match occurs between the PCA counter and the module's capture registers. To activate this mode the TOG, MAT, and ECOM bits in the module's CCAPM SFR must be set (see Figure 14).

### Pulse Width Modulator Mode

The PCA module can be used as a PWM output. Figure 15 shows the PWM function. The frequency of the output depends on the source for the PCA timer. The duty cycle of the module is independently variable using the module's capture register CCAPL. When the value of the PCA CL SFR is less than the value in the module's CCAPL SFR, the output will be low, when it is equal to or greater than the output will be high. When CL overflows from FF to 00, CCAPL is reloaded with the value in CCAPH. This allows updating the PWM without glitches. The PWM and ECOM bits in the module's CCAPM register must be set to enable the PWM mode.

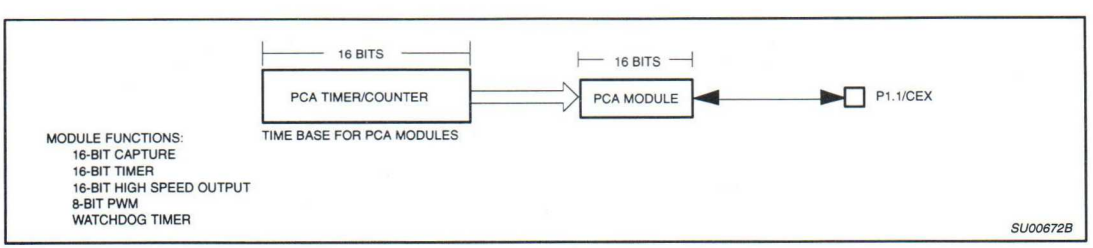



**Figure 5. Programmable Counter Array (PCA)**

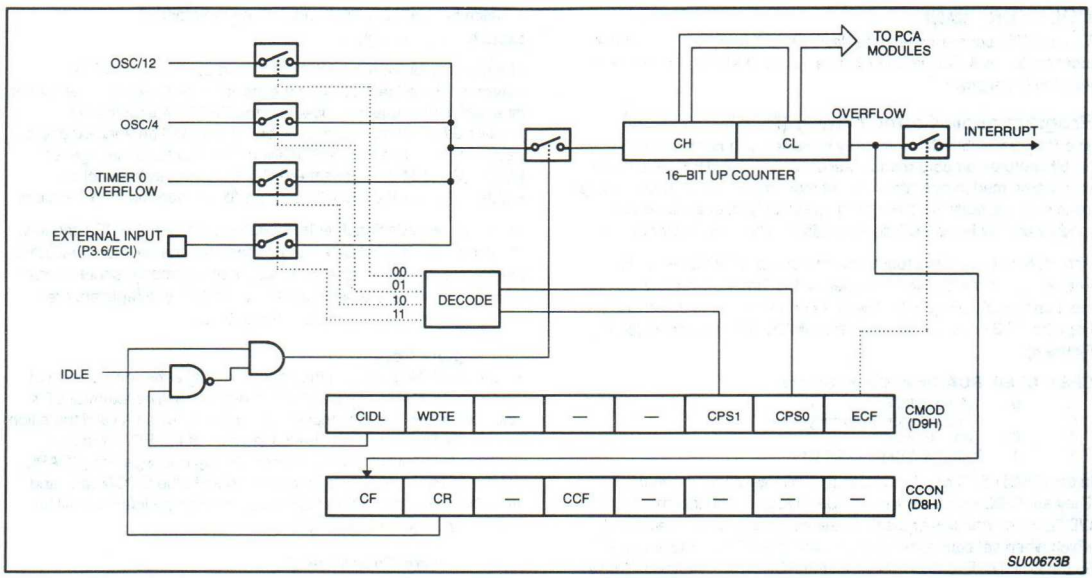



**Figure 6. PCA Timer/Counter**

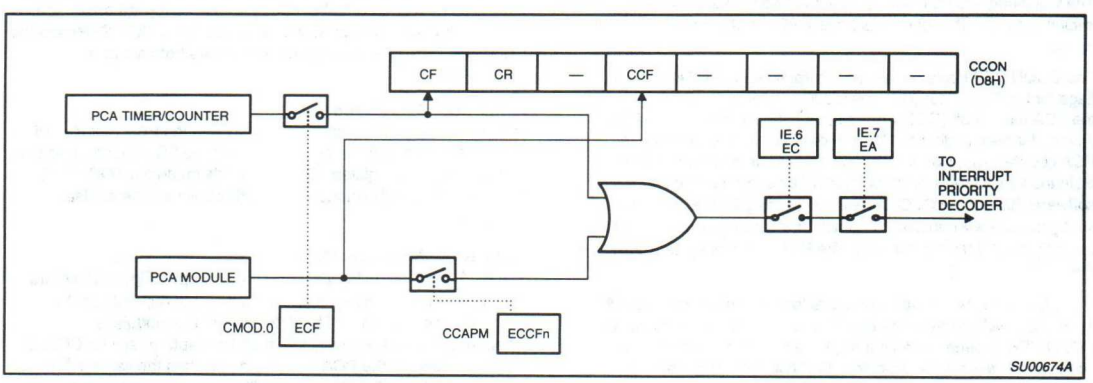



**Figure 7. PCA Interrupt System**

**CMOD** Address = OD9H Reset Value = 00XX X000B

Bit Addressable

| CIDL | WDTE | – | – | – | CPS1 | CPS0 | ECF |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Bit:

| Symbol | Function |
|---|---|
| CIDL | Counter Idle control: CIDL = 0 programs the PCA Counter to continue functioning during idle Mode. CIDL = 1 programs it to be gated off during idle. |
| WDTE | Watchdog Timer Enable: WDTE = 0 disables Watchdog Timer function on PCA Module. WDTE = 1 enables it. |
| – | Not implemented, reserved for future use.* |
| CPS1 | PCA Count Pulse Select bit 1. |
| CPS0 | PCA Count Pulse Select bit 0. |

| CPS1 | CPS0 | Selected PCA Input** | |
|---|---|---|---|
| 0 | 0 | 0 | Internal clock, $f_{OSC} \div 12$ |
| 0 | 1 | 1 | Internal clock, $f_{OSC} \div 4$ |
| 1 | 0 | 2 | Timer 0 overflow |
| 1 | 1 | 3 | External clock at ECI/P3.1 pin (max. rate = $f_{OSC} \div 8$) |

| | |
|---|---|
| ECF | PCA Enable Counter Overflow interrupt: ECF = 1 enables CF bit in CCON to generate an interrupt. ECF = 0 disables that function of CF. |

NOTE:
* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.
** $f_{OSC}$ = oscillator frequency

SU00675A

**Figure 8. CMOD: PCA Counter Mode Register**

**CCON** Address = OD8H Reset Value = 00X0 0000B

Bit Addressable

| CF | CR | — | CCF | — | — | — | — |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Bit:

| Symbol | Function |
|---|---|
| CF | PCA Counter Overflow flag. Set by hardware when the counter rolls over. CF flags an interrupt if bit ECF in CMOD is set. CF may be set by either hardware or software but can only be cleared by software. |
| CR | PCA Counter Run control bit. Set by software to turn the PCA counter on. Must be cleared by software to turn the PCA counter off. |
| – | Not implemented, reserved for future use*. |
| CCF | PCA Module interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software. |

NOTE:
* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

SU00676A

**Figure 9. CCON: PCA Counter Control Register**

**CCAPMn** Address CCAPM 0DEH Reset Value = X000 0000B

Not Bit Addressable

| | — | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF |
|---|---|---|---|---|---|---|---|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Symbol | Function |
|---|---|
| – | Not implemented, reserved for future use*. |
| **ECOM** | Enable Comparator. ECOM = 1 enables the comparator function. |
| **CAPP** | Capture Positive, CAPP = 1 enables positive edge capture. |
| **CAPN** | Capture Negative, CAPN = 1 enables negative edge capture. |
| **MAT** | Match. When MAT = 1, a match of the PCA counter with this module's compare/capture register causes the CCF bit in CCON to be set, flagging an interrupt. |
| **TOG** | Toggle. When TOG = 1, a match of the PCA counter with this module's compare/capture register causes the CEX pin to toggle. |
| **PWM** | Pulse Width Modulation Mode. PWM4 = 1 enables the CEX pin to be used as a pulse width modulated output. |
| **ECCF** | Enable CCF interrupt. Enables compare/capture flag CCF in the CCON register to generate an interrupt. |

**NOTE:**
*User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

*SU00677A*

**Figure 10. CCAPM: PCA Modules Compare/Capture Registers**

| – | ECOM | CAPP | CAPN | MAT | TOG | PWM | ECCF | MODULE FUNCTION |
|---|---|---|---|---|---|---|---|---|
| X | 0 | 0 | 0 | 0 | 0 | 0 | 0 | No operation |
| X | X | 1 | 0 | 0 | 0 | 0 | X | 16-bit capture by a positive-edge trigger on CEX |
| X | X | 0 | 1 | 0 | 0 | 0 | X | 16-bit capture by a negative trigger on CEX |
| X | X | 1 | 1 | 0 | 0 | 0 | X | 16-bit capture by a transition on CEX |
| X | 1 | 0 | 0 | 1 | 0 | 0 | X | 16-bit Software Timer |
| X | 1 | 0 | 0 | 1 | 1 | 0 | X | 16-bit High Speed Output |
| X | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 8-bit PWM |
| X | 1 | 0 | 0 | 1 | X | 0 | X | Watchdog Timer |

**Figure 11. PCA Module Modes (CCAPM Register)**

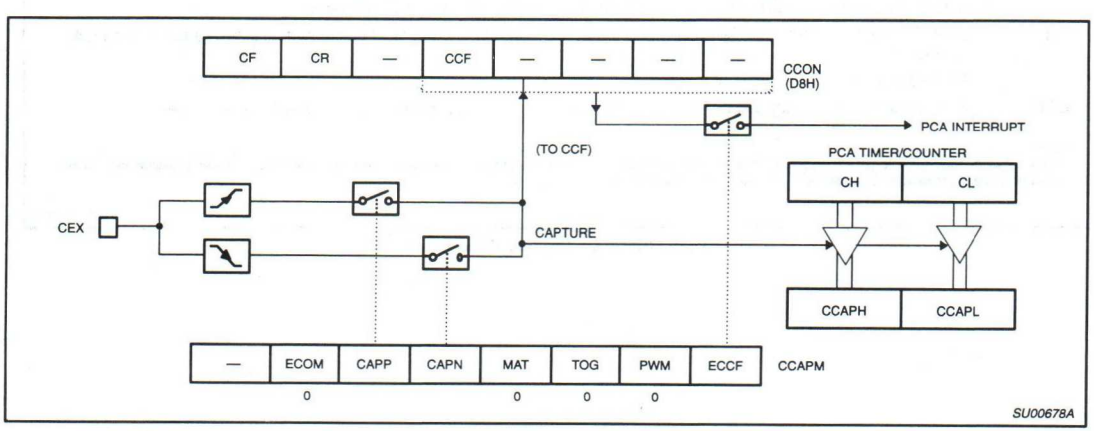



**Figure 12. PCA Capture Mode**

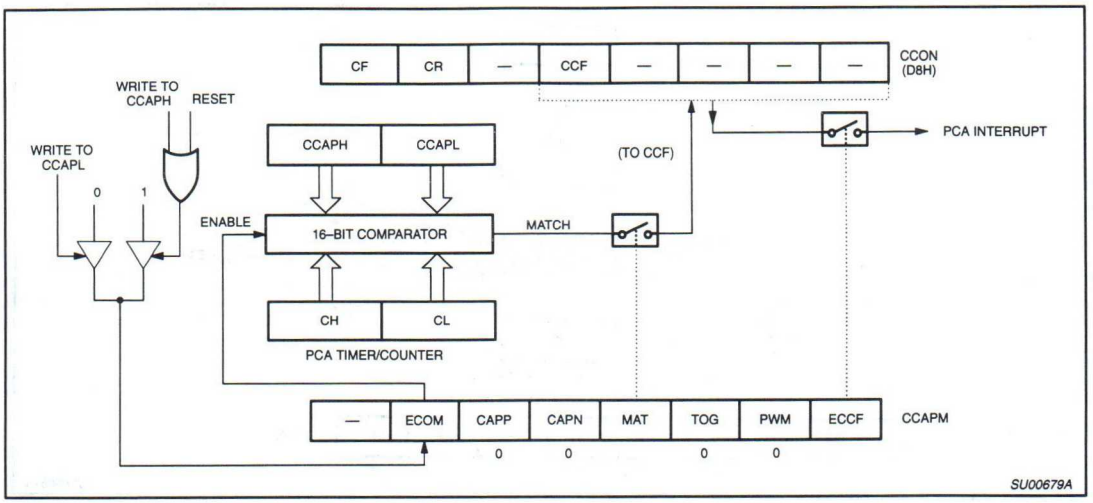



**Figure 13. PCA Compare Mode**

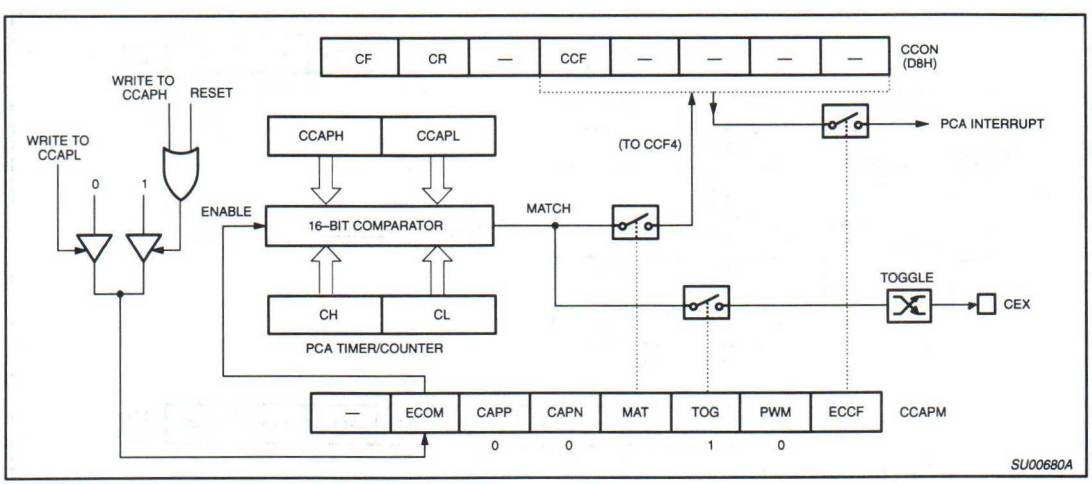



**Figure 14. PCA High Speed Output Mode**

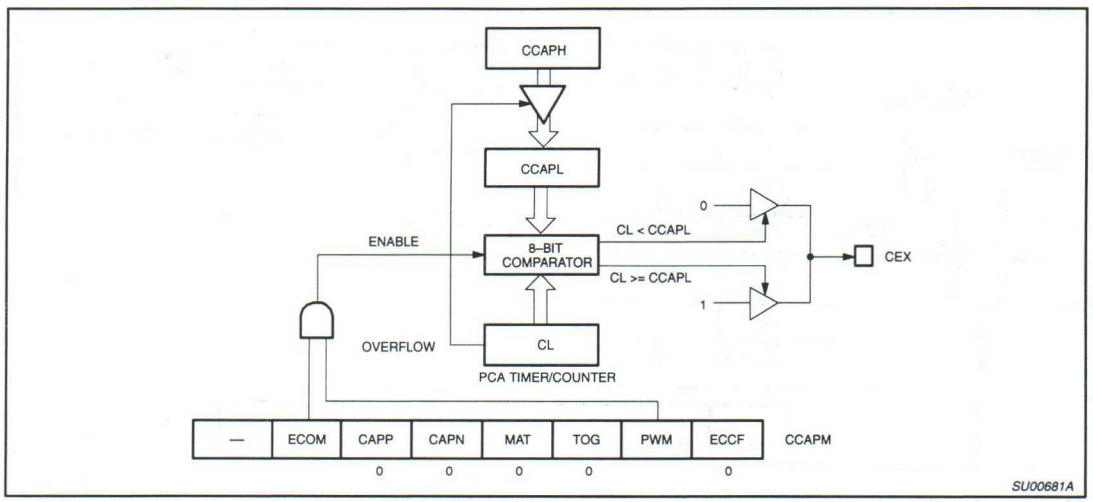



**Figure 15. PCA PWM Mode**

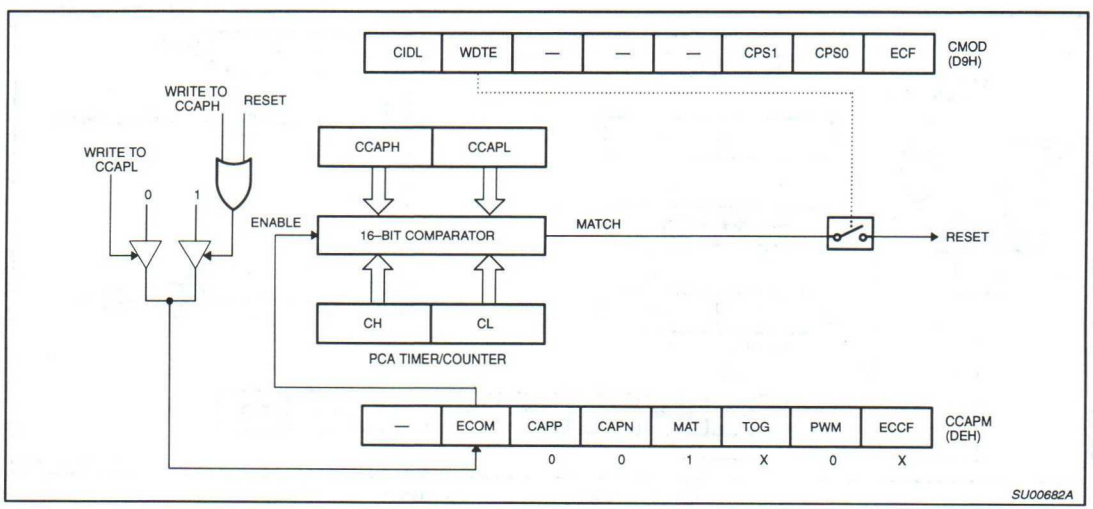



**Figure 16. PCA Watchdog Timer**

## ABSOLUTE MAXIMUM RATINGS[1, 3, 4]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage from $V_{CC}$ to $V_{SS}$ | –0.5 to +6.5 | V |
| Voltage from any pin to $V_{SS}$ (except $V_{PP}$) | –0.5 to $V_{CC}$ + 0.5 | V |
| Power dissipation | 1.0 | W |
| Voltage from $V_{PP}$ pin to $V_{SS}$ | –0.5 to + 13.0 | V |

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, $AV_{CC}$ = 5V ±5, $AV_{SS}$ = 0V[4]
$V_{CC}$ = 5V ± 10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS[4] | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $I_{CC}$ | Supply current (see Figure 19) | | | | | |
| **Inputs** | | | | | | |
| $V_{IL}$ | Input low voltage, port 1, 3 | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IH}$ | Input high voltage, port 1, 3 | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, X1, RST | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| **Outputs** | | | | | | |
| $V_{OL}$ | Output low voltage, port 3 | $I_{OL}$ = 1.6mA[2] | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 1.0, 1.1, 1.2 | $I_{OL}$ = 3.2mA[2] | | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 3, 1.0, 1.1 | $I_{OH}$ = –60µA, | 2.4 | | | V |
| $I_{LI}$ | Input leakage current, port 1, 3, RST | 0.45 < $V_{IN}$ < $V_{CC}$ | | | +10 | µA |
| $I_{IL}$ | Logical 0 input cirrent, ports 1 and 3 | $V_{IN}$ = 0.45V | | | –50 | µA |
| $C_{IO}$ | Pin capacitance | Test freq = 1MHz, $T_{amb}$ = 25°C | | | 10 | pF |
| $I_{PD}$ | Power-down current[5] | $V_{CC}$ = 2 to 5.5V<br>$V_{CC}$ = 2 to 6.0V (83C754) | | | 50 | µA |
| $V_{PP}$ | $V_{PP}$ program voltage (87C754 only) | $V_{SS}$ = 0V<br>$V_{CC}$ = 5V±10%<br>$T_{amb}$ = 21°C to 27°C | 12.5 | | 13.0 | V |
| $I_{PP}$ | Program current (87C754 only) | $V_{PP}$ = 13.0V | | | 50 | mA |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. Under steady state (non-transient) conditions, IOL must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 10mA
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 67mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
3. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
4. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.
5. Power-down $I_{CC}$ is measured with all output pins disconnected; port 0 = $V_{CC}$; X2, X1 n.c.; RST = $V_{SS}$.
6. $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; RST = port 0 = $V_{CC}$. $I_{CC}$ will be slightly higher if a crystal oscillator is used.
7. Idle $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; port 0 = $V_{CC}$; RST = $V_{SS}$.
8. Load capacitance for ports = 80pF.

## ANALOG SECTION ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, $AV_{CC}$ = 5V ±5, $AV_{SS}$ = 0V[4]
$V_{CC}$ = 5V ± 10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS[4] | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| **Analog Inputs** (D/A guaranteed only with quartz window covered.) | | | | | | |
| $AV_{CC}$ | Analog supply voltage | | 4.5 | – | 5.5 | V |
| | Sensor resistor | | 330 | – | 3K | Ω |
| $IAV_{CC}$ | | AC0 = 0 IC only | – | 0.88 | 1.5 | mA |
| | | AC0 = 1 | – | – | 10 | µA |
| **Regulator** | | | | | | |
| VREG | | | 3.6 | 3.8 | 4.0 | V |
| IVREG | | | 13 | – | 55 | mA |
| CDECOUPLE | | Stability requirement | 3 | 10 | – | µF |
| RDSONQ1 | | | – | 7 | – | Ω |
| ILEAKAGEQ1 | | | – | TBD | – | µA |
| ILEAKAGEQ2 | | | – | TBD | – | µA |
| PSRR | | 100Hz | – | –40 | – | dB |
| VREGREJ | VREG rejection of 1 Volt $AV_{CC}$ step change | | –100 | – | 100 | mV |
| TVREG | VREG turn on time | Q1 off, 330Ω sensor | – | 2 | 5 | ms |
| **MUX and Comparator** | | | | | | |
| | Comparator trip point | | 1.14 | 1.26 | 1.38 | V |
| | Comparator delay input | 0.04V/µs | – | 50 | – | ns |
| | Comparator delay change | $AV_{CC}$ 4.5 to 5.5V | –10 | 2 | 10 | ns |
| | MUX impedance | | – | 1 | – | kΩ |
| ILEAKAGEMUX | | | – | TBD | – | µA |
| | | | | | | |
| | | | | | | |
| **Digital-to-Analog Conversion** | | | | | | |
| | ZDAC, XYDAC monotonicity | | 0 | – | – | bits |
| | ZDAC, XYDAC impedance | | – | 10 | – | kΩ |
| | DAC selection switch impedance | | – | 40 | – | Ω |
| | DAC settling | | – | 1 | – | µs |
| | ZDAC switch impedance | | – | 50 | – | Ω |
| | ZDAC switch impedance change | $AV_{CC}$ 4.5 to 5.5V | –20 | – | 20 | Ω |
| | ZDAC switch leakage | | – | TBD | – | µA |
| **Switches** | | | | | | |
| | XYZRAMP impedance | | – | 25 | 100 | Ω |
| | XYZRAMP impedance change | $AV_{CC}$ 4.5 to 5.5V | –25 | – | 25 | Ω |
| | XYZRAMP leakage | | – | TBD | – | µA |
| | XYZRAMP discharge to 1LSB (1.6mV) | | – | 1.5 | 10 | µs |
| | XYZRAMP delay turn on time | | – | 6 | 50 | ns |
| | XYZRAMP start time change | $AV_{CC}$ 4.5 to 5.5V | –10 | – | 10 | ns |
| | XYDACBIAS impedance | | – | 7 | 13 | Ω |
| | XYDACBIAS leakage | | – | TBD | – | µA |
| | XYDACBIAS switching time | | – | 130 | 1000 | ns |
| | XYSOURCE impedance | | – | 150 | 300 | Ω |
| | XYSOURCE impedance change | $AV_{CC}$ 4.5 to 5.5V | –100 | – | 100 | Ω |
| | XYSOURCE leakage | | – | TBD | – | µA |
| | XYSOURCE switching time | | – | 30 | 500 | ns |

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[4, 8]

| SYMBOL | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | Oscillator frequency | | | 3.5 | 16 | MHz |
| **External Clock (Figure 17)** | | | | | | |
| $t_{CHCX}$ | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | Fall time | | 20 | | 20 | ns |

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal.
The designations are:

C – Clock
D – Input data
H – Logic level high
L – Logic level low
Q – Output data
T – Time
V – Valid
X – No longer a valid logic level
Z – Float

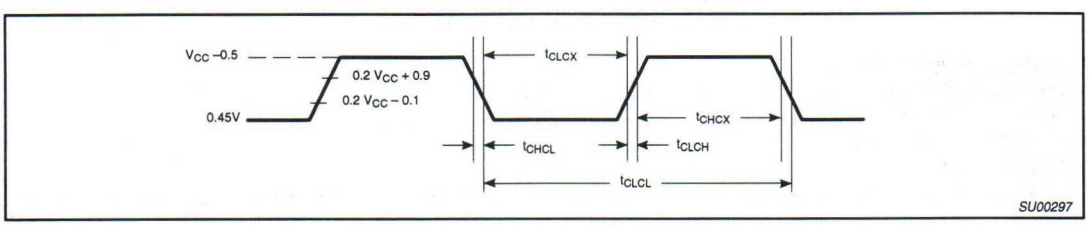



**Figure 17. External Clock Drive**

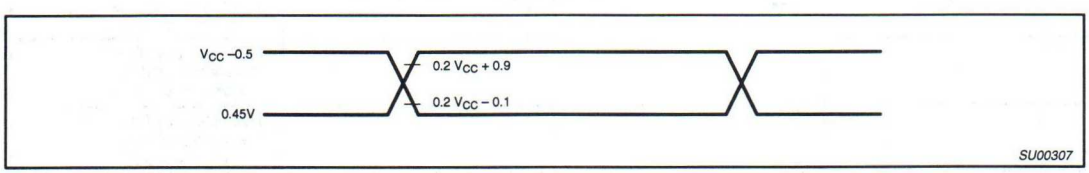



**Figure 18. AC Testing Input/Output**

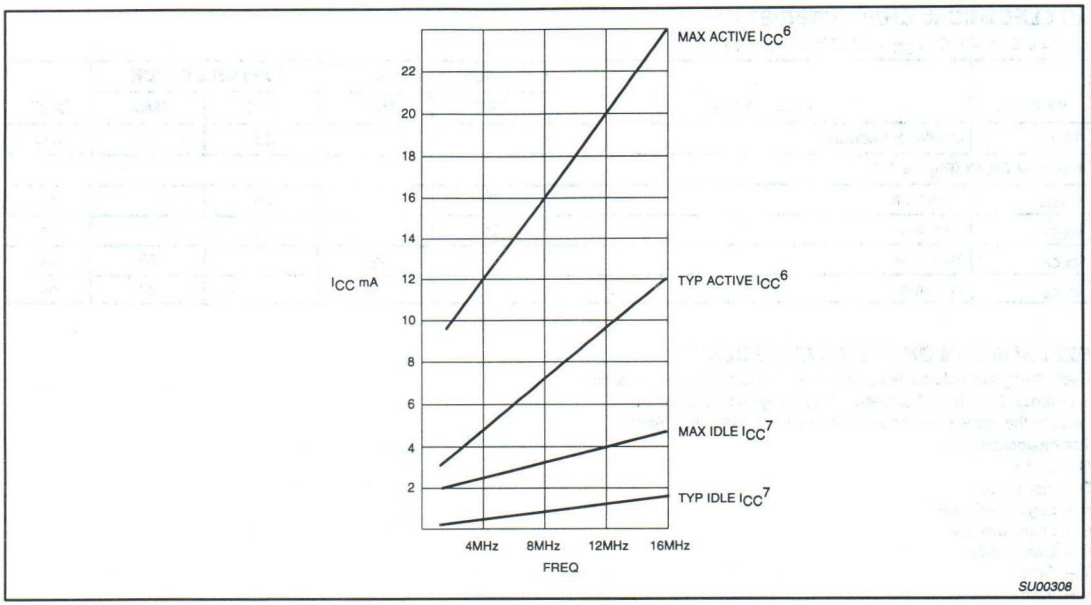



**Figure 19. $I_{CC}$ vs. FREQ**
**Maximum $I_{CC}$ values taken at $V_{CC}$ = 5.5V and worst case temperature.**
**Typical $I_{CC}$ values taken at $V_{CC}$ = 5.0V and 25°C.**
**Notes 6 and 7 refer to AC Electrical Characteristics.**

## ROM CODE SUBMISSION

When submitting ROM code for the 83C754, the following must be specified:

1. 4k byte user ROM data
2. 64 byte ROM encryption key
3. ROM security bits.

| ADDRESS | CONTENT | BIT(S) | COMMENT |
|---|---|---|---|
| 0000H to 0FFFH | DATA | 7:0 | User ROM Data |
| 1000H to 101FH | KEY | 7:0 | ROM Encryption Key<br>FFH = no encryption |
| 1020H | SEC | 0 | ROM Security Bit 1<br>0 = enable security<br>1 = disable security |
| 1020H | SEC | 1 | ROM Security Bit 2<br>0 = enable security<br>1 = disable security |

**Security Bit 1:** When programmed, this bit has two effects on masked ROM parts:
1. External MOVC is disabled, and
2. EA# is latched on Reset.

**Security Bit 2:** When programmed, this bit inhibits Verify User ROM.

If the ROM Code file does not include the options, the following information must be included with the ROM code.

For each of the following, check the appropriate box, and send to Philips along with the code:

Security Bit #1: ☐ Enabled ☐ Disabled

Security Bit #2: ☐ Enabled ☐ Disabled

Encryption: ☐ No ☐ Yes If Yes, must send key file.

## PROGRAMMING CONSIDERATIONS

### EPROM Characteristics

The 87C754 is programmed by using a modified Quick-Pulse Programming algorithm similar to that used for devices such as the 87C751 and 87C752.

Figure 20 shows a block diagram of the programming configuration for the 87C754. Port pin P0.2 is used as the programming voltage supply input ($V_{PP}$ signal). Port pin P0.1 is used as the program (PGM/) signal. This pin is used for the 5 programming pulses.

Port 3 is used as the address input for the byte to be programmed and accepts both the high and low components of the eleven bit address. Multiplexing of these address components is performed using the ASEL input. The user should drive the ASEL input high and then drive port 3 with the high order bits of the address. ASEL should remain high for at least 13 clock cycles. ASEL may then be driven low which latches the high order bits of the address internally. The high address should remain on port 3 for at least two clock cycles after ASEL is driven low. Port 3 may then be driven with the low byte of the address. The low address will be internally stable 13 clock cycles later. The address will remain stable provided that the low byte placed on port 3 is held stable and ASEL is kept low.
**Note:** ASEL needs to be pulsed high only to change the high byte of the address.

Port 1 is used as a bidirectional data bus during programming and verify operations. During programming mode, it accepts the byte to be programmed. During verify mode, it provides the contents of the EPROM location specified by the address which has been supplied to Port 3.

The XTAL1 pin is the oscillator input and receives the master system clock. This clock should be between 1.2 and 16MHz.

The RESET pin is used to accept the serial data stream that places the 87C754 into various programming modes. This pattern consists of a 10-bit code with the LSB sent first. Each bit is synchronized to the clock input, X1.

### Programming Operation

Figures 21 and 22 show the timing diagrams for the program/verify cycle. RESET should initially be held high for at least two machine cycles. P0.1 (PGM/) and P0.2 ($V_{PP}$) will be at $V_{OH}$ as a result of the RESET operation. At this point, these pins function as normal quasi-bidirectional I/O ports and the programming equipment may pull these lines low. However, prior to sending the 10-bit code on the RESET pin, the programming equipment should drive these pins high ($V_{IH}$). The RESET pin may now be used as the serial data input for the data stream which places the 87C754 in the programming mode. Data bits are sampled during the clock high time and thus should only change during the time that the clock is low. Following transmission of the last data bit, the RESET pin should be held low.

Next the address information for the location to be programmed is placed on port 3 and ASEL is used to perform the address multiplexing, as previously described. At this time, port 1 functions as an output.

A high voltage $V_{PP}$ level is then applied to the $V_{PP}$ input (P0.2). (This sets Port 1 as an input port). The data to be programmed into the EPROM array is then placed on Port 1. This is followed by a series of programming pulses applied to the PGM/ pin (P0.1). These pulses are created by driving P0.1 low and then high. This pulse is repeated until a total of 5 programming pulses have occurred. At the conclusion of the last pulse, the PGM/ signal should remain high.

The $V_{PP}$ signal may now be driven to the $V_{OH}$ level, placing the 87C754 in the verify mode. (Port 1 is now used as an output port). After four machine cycles (48 clock periods), the contents of the addressed location in the EPROM array will appear on Port 1.

The next programming cycle may now be initiated by placing the address information at the inputs of the multiplexed buffers, driving the $V_{PP}$ pin to the $V_{PP}$ voltage level, providing the byte to be programmed to Port1 and issuing the 5 programming pulses on the PGM/ pin, bringing $V_{PP}$ back down to the $V_C$ level and verifying the byte.

### Programming Modes

The 87C754 has four programming features incorporated within its EPROM array. These include the USER EPROM for storage of the application's code, a 64-byte encryption key array and two security bits. Programming and verification of these four elements are selected by a combination of the serial data stream applied to the RESET pin and the voltage levels applied to port pins P0.1 and P0.2. The various combinations are shown in Table 3.

### Encryption Key Table

The 87C754 includes a 64-byte EPROM array that is programmable by the end user. The contents of this array can then be used to encrypt the program memory contents during a program memory verify operation. When a program memory verify operation is performed, the contents of the program memory location is XNOR'ed with one of the bytes in the 64-byte encryption table. The resulting data pattern is then provided to port 1 as the verify data. The encryption mechanism can be disabled, in essence, by leaving the bytes in the encryption table in their erased state (FFH) since the XNOR product of a bit with a logical one will result in the original bit. The encryption bytes are mapped with the code memory in 64-byte groups. the first byte in code memory will be encrypted with the first byte in the encryption table; the second byte in code memory will be encrypted with the second byte in the encryption table and so forth up to and including the 64th byte. The encryption repeats in 64-byte groups; the 65th byte in the code memory will be encrypted with the first byte in the encryption table, and so forth.

### Security Bits

Two security bits, security bit 1 and security bit 2, are provided to limit access to the USER EPROM and encryption key arrays. Security bit 1 is the program inhibit bit, and once programmed performs the following functions:

1. Additional programming of the USER EPROM is inhibited.
2. Additional programming of the encryption key is inhibited.
3. Verification of the encryption key is inhibited.
4. Verification of the USER EPROM and the security bit levels may still be performed.

(If the encryption key array is being used, this security bit should be programmed by the user to prevent unauthorized parties from reprogramming the encryption key to all logical zero bits. Such programming would provide data during a verify cycle that is the logical complement of the USER EPROM contents).

Security bit 2, the verify inhibit bit, prevents verification of both the USER EPROM array and the encryption key arrays. The security bit levels may still be verified.

## Programming and Verifying Security Bits

Security bits are programmed employing the same techniques used to program the USER EPROM and KEY arrays using serial data streams and logic levels on port pins indicated in Table 3. When programming either security bit, it is not necessary to provide address or data information to the 87C754 on ports 1 and 3.

Verification occurs in a similar manner using the RESET serial stream shown in Table 3. Port 3 is not required to be driven and the results of the verify operation will appear on ports 1.6 and 1.7.

Ports 1.7 contains the security bit 1 data and is a logical one if programmed and a logical zero if erased. Likewise, P1.6 contains the security bit 2 data and is a logical one if programmed and a logical zero if erased.

## Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. **For this and secondary effects, it is recommended that an opaque label be placed over the window.** For elevated temperature or environments where solvents are being used, apply Kapton tape Flourless part number 2345–5 or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-sec/cm$^2$. Exposing the EPROM to an ultraviolet lamp of 12,000uW/cm$^2$ rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

**Table 3. Implementing Program/Verify Modes**

| OPERATION | SERIAL CODE | $\overline{PGM}$ | $V_{PP}$ |
|---|---|---|---|
| Program user EPROM | 296H | –* | $V_{PP}$ |
| Verify user EPROM | 296H | $V_{IH}$ | $V_{IH}$ |
| Program key EPROM | 292H | –* | $V_{PP}$ |
| Verify key EPROM | 292H | $V_{IH}$ | $V_{IH}$ |
| Program security bit 1 | 29AH | –* | $V_{PP}$ |
| Program security bit 2 | 298H | –* | $V_{PP}$ |
| Verify security bits | 29AH | $V_{IH}$ | $V_{IH}$ |

**NOTE:**

* Pulsed from $V_{IH}$ to $V_{IL}$ and returned to $V_{IH}$.

## EPROM PROGRAMMING AND VERIFICATION

$T_{amb}$ = 21°C to +27°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| $1/t_{CLCL}$ | Oscillator/clock frequency | 1.2 | 16 | MHz |
| $t_{AVGL}$[1] | Address setup to $\overline{PGM}$ low | $10\mu s + 24t_{CLCL}$ | | |
| $t_{GHAX}$ | Address hold after $\overline{PGM}$ high | $48t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PGM}$ low | $38t_{CLCL}$ | | |
| $t_{DVGL}$ | Data setup to $\overline{PGM}$ low | $38t_{CLCL}$ | | |
| $t_{GHDX}$ | Data hold after $\overline{PGM}$ high | $36t_{CLCL}$ | | |
| $t_{SHGL}$ | $V_{PP}$ setup to $\overline{PGM}$ low | 10 | | µs |
| $t_{GHSL}$ | $V_{PP}$ hold after $\overline{PGM}$ | 10 | | µs |
| $t_{GLGH}$ | $\overline{PGM}$ width | 90 | 110 | µs |
| $t_{AVQV}$[2] | $V_{PP}$ low ($V_{CC}$) to data valid | | $48t_{CLCL}$ | |
| $t_{GHGL}$ | $\overline{PGM}$ high to $\overline{PGM}$ low | 10 | | µs |
| $t_{SYNL}$ | P0.0 (sync pulse) low | $4t_{CLCL}$ | | |
| $t_{SYNH}$ | P0.0 (sync pulse) high | $8t_{CLCL}$ | | |
| $t_{MASEL}$ | ASEL high time | $13t_{CLCL}$ | | |
| $t_{MAHLD}$ | Address hold time | $2t_{CLCL}$ | | |
| $t_{HASET}$ | Address setup to ASEL | $13t_{CLCL}$ | | |
| $t_{ADSTA}$ | Low address to address stable | $13t_{CLCL}$ | | |

**NOTES:**
1. Address should be valid at least $24t_{CLCL}$ before the rising edge of $V_{PP}$.
2. For a pure verify mode, i.e., no program mode in between, $t_{AVQV}$ is $14t_{CLCL}$ maximum.

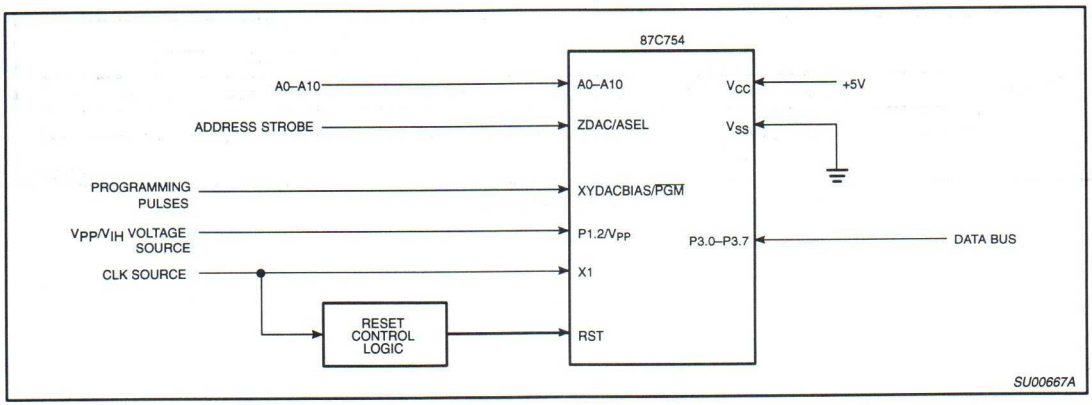



**Figure 20. Programming Configuration**

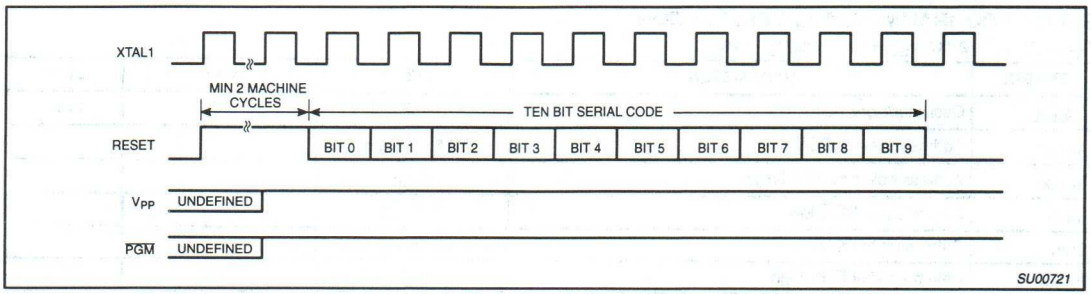



**Figure 21. Entry into Program/Verify Modes**

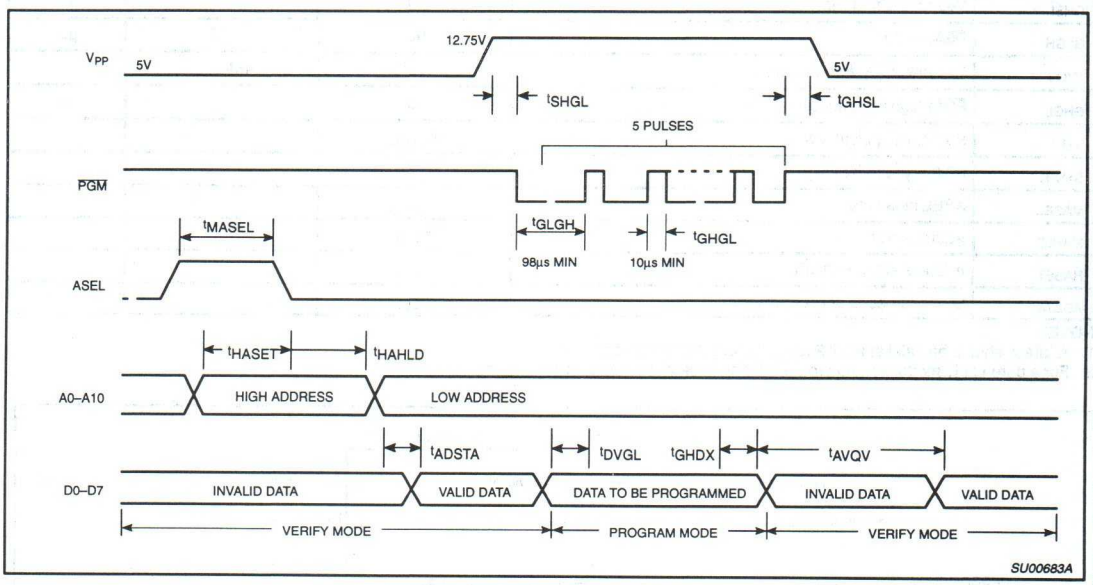



**Figure 22. Program/Verify Cycle**

# Microcontroller with TrackPoint™ microcode from IBM TPM754

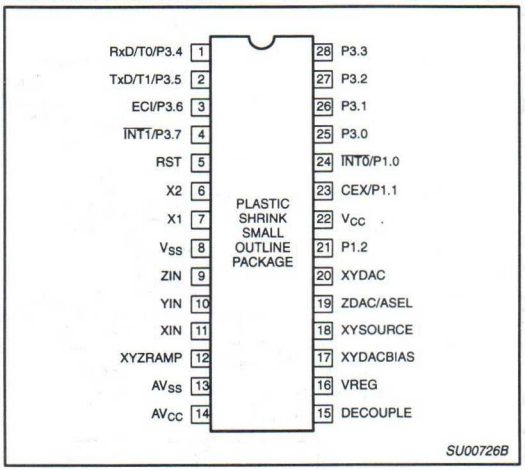
The Philips Semiconductors TPM754 is a small package, low cost, ROM-coded 80C51 with IBM®'s TrackPoint™ pointing algorithms and control code. TrackPoint is the result of years of human factors research and innovation at IBM. The result is a "velocity sensitive" pointing solution more efficient and easier to use than "position sensitive" devices such as the mouse, the trackball, or the touchpad.

IBM has licensed Philips Semiconductors to sell microcontrollers with TrackPoint code. By purchasing a TPM from Philips, the purchaser becomes a sub-licensee of Philips. The selling price of Philips' TPM includes the royalties for IBM's intellectual property, which Philips in turn pays to IBM. Customers for TPMs do not need to sign any licensing agreement with either IBM or Philips. This code is the intellectual property of IBM, which is covered by numerous patents, and must be treated accordingly.

The TPM754 contains IBM® TrackPoint™ code, a single module PCA, a 256 × 8 RAM, 21 I/O lines, two 16-bit counter/timers, a two-priority level interrupt structure, a full duplex serial channel, an on-chip oscillator, and an 8-bit D/A converter.

For identical device without TrackPoint code, see the 8XC754 datasheet.

## PIN CONFIGURATION




## FEATURES

- 80C51-based architecture
- Small package sizes – 28-pin SSOP
- Power control modes:
  - Idle mode
  - Power-down mode
- 256 × 8 RAM
- Two 16-bit auto reloadable counter/timers
- Single module PCA counter/timer
- Full duplex serial channel
- Boolean processor
- CMOS and TTL compatible

## ORDERING INFROMATION

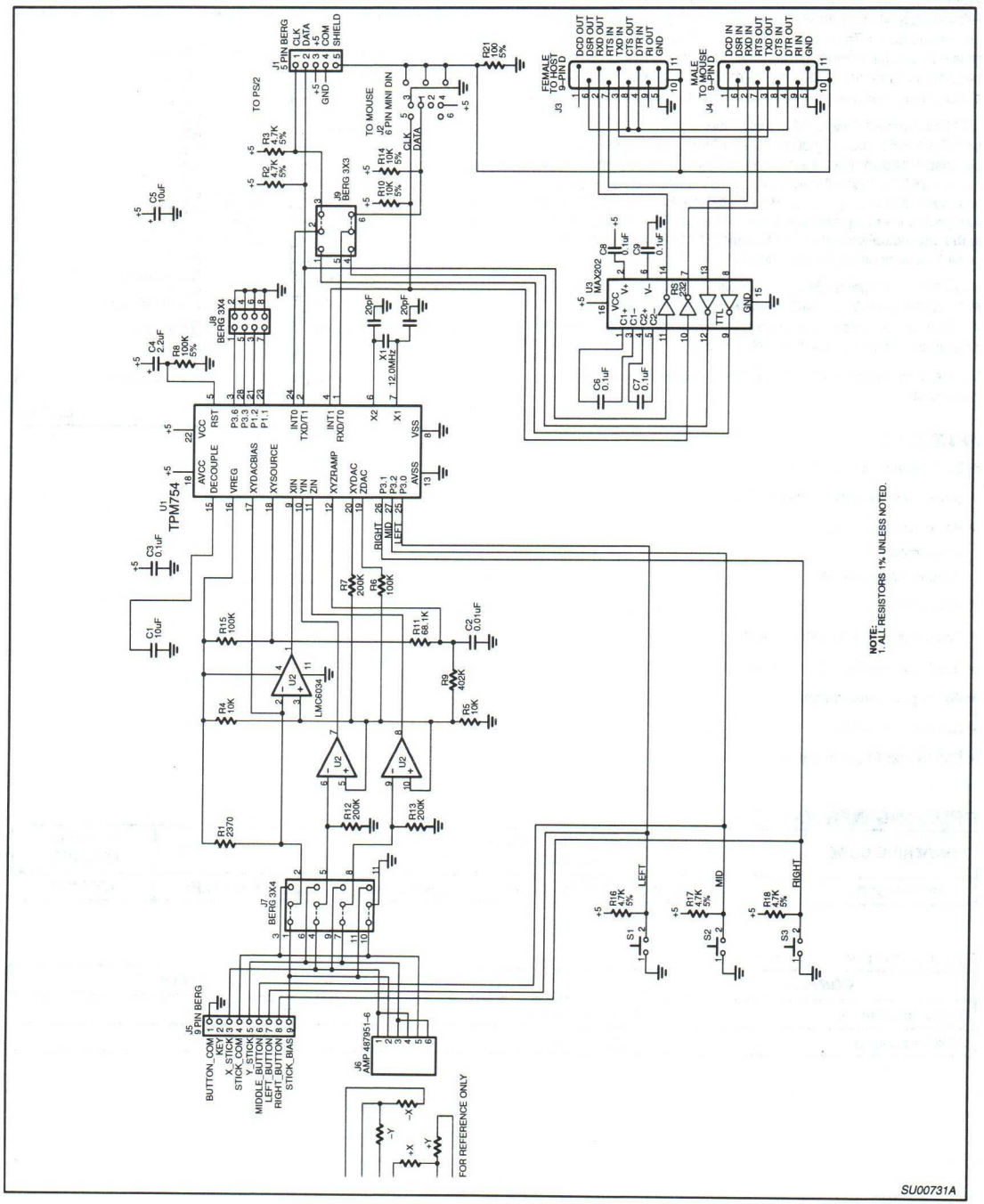
| ORDERING CODE | TEMPERATURE RANGE °C AND PACKAGE | FREQUENCY | DRAWING NUMBER |
|---|---|---|---|
| PTPM754 DB | 0 to +70, 28-pin Shrink Small Outline Package | 3.5 to 12MHz | SOT341-1 |

For compatible pointing device, contact:

| COMPANY | CONTACT | TELEPHONE |
|---|---|---|
| Bokam Engineering | Ms. Jane Kamenster | (714)513-2200 |
| CTS Corporation | Mr. Dave Poole | (219)589-7169 |

IBM is a registered trademark, and TrackPoint is a trademark of IBM Corporation.

## SCHEMATIC OF TRACKPOINT SYSTEM



## PIN DESCRIPTION

| MNEMONIC | DIP PIN NO. | TYPE | NAME AND FUNCTION |
|---|---|---|---|
| $V_{SS}$ | 8 | I | **Circuit Ground Potential.** |
| $V_{CC}$ | 22 | I | **Supply voltage during normal, idle, and power-down operation.** |
| P1.0–P1.2 | 21, 23, 24 | I/O | **Port 1:** Port 1 is a 3-bit bidirectional I/O port with internal pull-ups on P1.0 and P1.1. Port 1 pins that have 1s written to them can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups (P1.0, P1.1). (See DC Electrical Characteristics: $I_{IL}$). Port 1 also serves the special function features listed below (Note: P1.0 does **not** have the strong pullup that is on for 2 oscillator periods.): |
| | 24 | I | **$\overline{INT0}$ (P1.0):** External interrupt 0. |
| | 23 | O | **CEX (P1.1):** PCA clock output. |
| P3.0–P3.7 | 1–4, 25–28 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). (Note: P3.5 does **not** have the strong pullup that is on for 2 oscillator periods.) Port 3 also serves the special function as listed below: |
| | 3 | I | **ECI (P3.6):** External PCA clock input. |
| | 1 | I | **RxD/T0 (P3.4):** Serial port receiver data input. Timer 0 external clock input. |
| | 4 | I | **$\overline{INT1}$:** External interrupt 1. |
| | 2 | I | **TxD/T1 (P3.5):** Serial port transmitter data. Timer 1 external clock input. |
| RST | 5 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running resets the device. **(NOTE: The TPM754 does not have an internal reset resistor.)** |
| X1 | 7 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| X2 | 6 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |
| $AV_{CC}$ [1] | 14 | I | **Analog supply voltage and reference input.** |
| $AV_{SS}$ [1] | 13 | I | **Analog supply and reference ground.** |
| ZIN | 9 | I | **ZIN:** Input to analog multiplexer. |
| YIN | 10 | I | **YIN:** Input to analog multiplexer. |
| XIN | 11 | I | **XIN:** Input to analog multiplexer. |
| XYZRAMP | 12 | O | **XYZRAMP:** Provides a low impedance pulldown to $V_{SS}$ under S/W control. |
| DECOUPLE | 15 | O | **DECOUPLE:** Output from regulated supply for connection of decoupling capacitors. |
| VREG | 16 | O | **VREG:** Provides regulated analog supply output. |
| XYDACBIAS | 17 | O | **XYDACBIAS:** Provides source voltage for bias of external circuitry. |
| XYSOURCE | 18 | O | **XYSOURCE:** Provides source voltage from regulated analog supply. |
| ZDAC | 19 | O | **ZDAC:** Switchable output from the internal DAC. |
| XYDAC | 20 | O | **XYDAC:** Non-switchable output from the internal DAC. |

**NOTE:**
1. $AV_{SS}$ (reference ground) must be connected to 0V (ground). $AV_{CC}$ (reference input) cannot differ from $V_{CC}$ by more than ±0.2V, and must be in the range 4.5V to 5.5V.

## OSCILLATOR CHARACTERISTICS

X1 and X2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, X1 should be driven while X2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## IDLE MODE

The TPM754 includes the 80C51 power-down and idle mode features. In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals except the D/A stays active. The functions that continue to run while in the idle mode are the timers and the interrupts. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset. Upon powering-up the circuit, or exiting from idle mode, sufficient time must be allowed for stabilization of the internal analog reference voltages before a D/A conversion is started.

## I/O Ports

The I/O pins provided by the TPM754 consist of port 1 and port 3.

### Port 1

Port 1 is a 3-bit bidirectional I/O port and includes alternate functions on some pins of this port. Pins P1.0 and P1.1 are provided with internal pullups while the remaining pin (P1.2) has an open drain output structure. The alternate functions for port 1 are:

$\overline{INT0}$ – External interrupt 0.
CEX – PCA clock output.

### Port 3

Port 3 is an 8-bit bidirectional I/O port structure.

The alternate functions for port 3 are:

RxD – Serial port receiver data input.
T1 – Timer 1 external clock input.
$\overline{INT1}$ – External interrupt 1.
TxD – Serial port transmitter data.
T0 – Timer 0 external clock input.
ECI – PCA external clock input.

## Analog Section

The analog section of the TPM754, shown in Figure 1, consists of four major elements: a bandgap referenced voltage regulator, an 8-bit DAC, an input multiplexer and comparator, and a low impedance pulldown device.

The bandgap voltage regulator uses the $AV_{CC}$ pin as its supply and produces a regulated output on the VREG pin. The regulator also supplies the analog supply voltage for the DAC. The regulator may be switched on/off by means of the AC1 bit in the analog control register (ACON0). The regulator output may also be supplied to the XYDACBIAS and XYSOURCE pins by means of bits AC3 and AC4, respectively. The DECOUPLE pin is provided for decoupling the regulator output.

The DAC is an 8-bit device and its output appears on the XYDAC pin. In addition, the DAC output may also be routed to the ZDAC pin by means of bit AC6 in the ACON0 register. The DAC output is not buffered, so external load impedances should be taken into consideration when using either of these outputs.

A 3-input multiplexer is provided, whose output is connected to the positive reference of a comparator. The multiplexer output is controlled by bits MUX2:0 of ACON1. A bandgap reference supplies the negative reference of the comparator. The output of the comparator may be used the trigger the capture input of module 4 of the PCA.

A low impedance pulldown is supplied at the XYZRAMP pin and is controlled by bit AC5 of ACON0.

The functions of the analog section are controlled by the IBM® TrackPoint™ code embedded within the Philips TPM754.

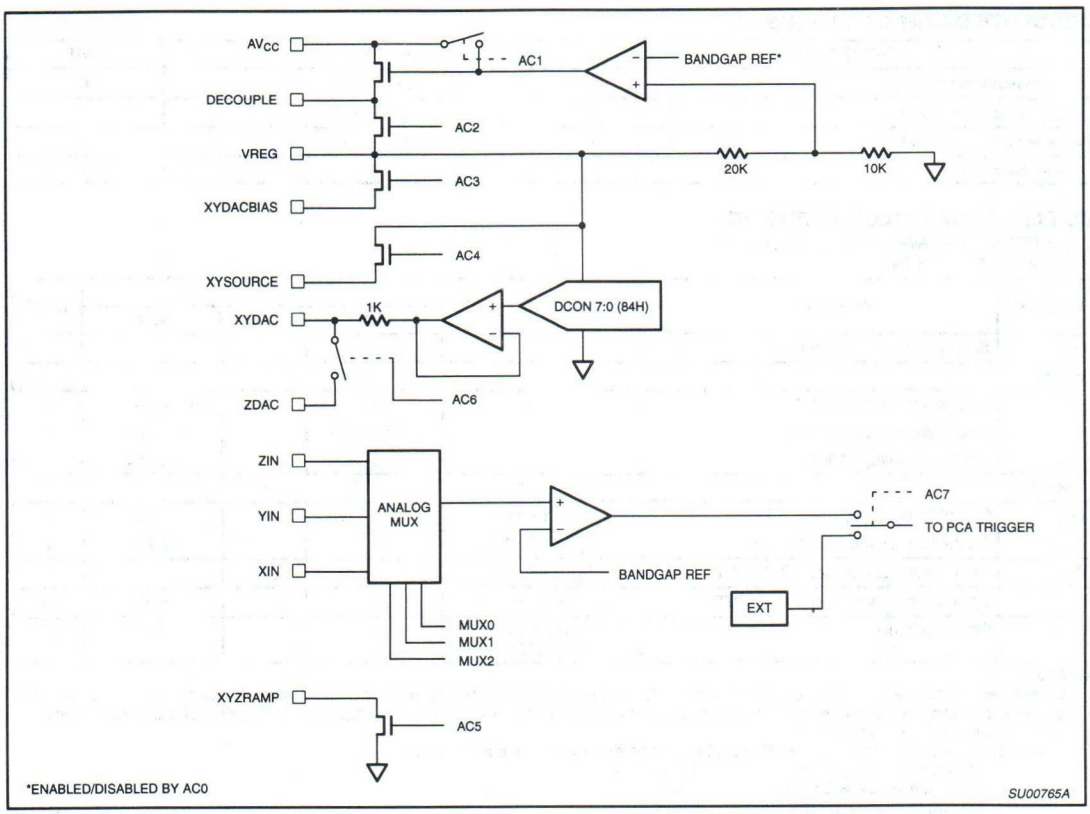



**Figure 1. Analog Section**

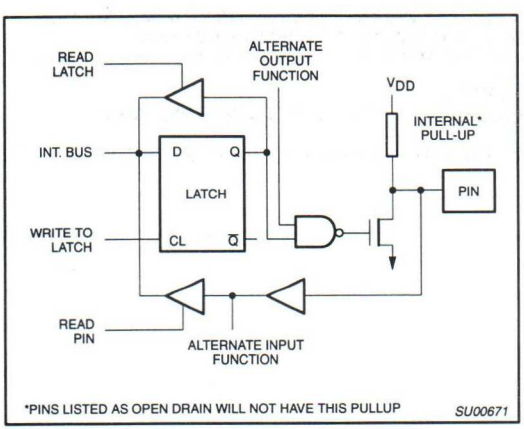



**Figure 2. Typical Port Bit Latches and I/O Buffers**

## ABSOLUTE MAXIMUM RATINGS[1, 3, 4]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage from $V_{CC}$ to $V_{SS}$ | –0.5 to +6.5 | V |
| Voltage from any pin to $V_{SS}$ | –0.5 to $V_{CC}$ + 0.5 | V |
| Power dissipation | 1.0 | W |

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, $AV_{CC}$ = 5V ±5, $AV_{SS}$ = 0V[4]
$V_{CC}$ = 5V ± 10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS[4] | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $I_{CC}$ | Supply current (see Figure 5) | | | | | |
| **Inputs** | | | | | | |
| $V_{IL}$ | Input low voltage, port 1, 3 | | –0.5 | | $0.2V_{CC}$–0.1 | V |
| $V_{IH}$ | Input high voltage, port 1, 3 | | $0.2V_{CC}$+0.9 | | $V_{CC}$+0.5 | V |
| $V_{IH1}$ | Input high voltage, X1, RST | | $0.7V_{CC}$ | | $V_{CC}$+0.5 | V |
| **Outputs** | | | | | | |
| $V_{OL}$ | Output low voltage, port 3, 1.2 | $I_{OL}$ = 1.6mA[2] | | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 1.0, 1.1 | $I_{OL}$ = 3.2mA[2] | | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 3, 1.0, 1.1 | $I_{OH}$ = –60µA, | 2.4 | | | V |
| $I_{LI}$ | Input leakage current, port 1, 3, RST | 0.45 < $V_{IN}$ < $V_{CC}$ | | | ±10 | µA |
| $C_{IO}$ | Pin capacitance | Test freq = 1MHz, $T_{amb}$ = 25°C | | | 10 | pF |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. Under steady state (non-transient) conditions, IOL must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 10mA
   Maximum $I_{OL}$ per 8-bit port: 26mA
   Maximum total $I_{OL}$ for all outputs: 67mA
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
3. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
4. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.
5. Power-down $I_{CC}$ is measured with all output pins disconnected; X2, X1 n.c.; RST = $V_{SS}$.
6. $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; RST = $V_{CC}$. $I_{CC}$ will be slightly higher if a crystal oscillator is used.
7. Idle $I_{CC}$ is measured with all output pins disconnected; X1 driven with $t_{CLCH}$, $t_{CHCL}$ = 5ns, $V_{IL}$ = $V_{SS}$ + 0.5V, $V_{IH}$ = $V_{CC}$ – 0.5V; X2 n.c.; RST = $V_{SS}$.

## ANALOG SECTION ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C; $V_{CC}$ = 5V ± 10%, $V_{SS}$ = 0V

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS[4] | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| **Analog Inputs** | | | | | | |
| $AV_{CC}$ | Analog supply voltage | | 4.5 | – | 5.5 | V |
| | Sensor resistor | | 330 | – | 3K | Ω |
| $IAV_{CC}$ | | AC0 = 0 IC only | – | 1.2 | 1.8 | mA |
| | | AC0 = 1 | – | – | 10 | µA |
| **Regulator** | | | | | | |
| VREG | | | 3.6 | 3.8 | 4.0 | V |
| IVREG | | | 13 | – | 80 | mA |
| CDECOUPLE | | Stability requirement | – | 10 | – | µF |
| RDSONQ1 | | | – | 7 | 12 | Ω |
| ILEAKAGEQ1 | | | –10 | | +10 | µA |
| ILEAKAGEQ2 | | | –10 | | +10 | µA |
| PSRR | | 100Hz | – | –40 | – | dB |
| **MUX and Comparator** | | | | | | |
| | Comparator trip point | | 1.14 | 1.26 | 1.38 | V |
| | MUX impedance | | – | 1 | 4 | kΩ |
| ILEAKAGEMUX | | | –10 | | +10 | µA |
| **Digital-to-Analog Conversion** | | | | | | |
| | ZDAC, XYDAC monotonicity | | 8 | – | – | bits |
| | ZDAC switch impedance | | – | 75 | 200 | Ω |
| | DAC output resistance | | – | 2.7 | 5 | kΩ |
| | ZDAC switch leakage | | –10 | | +10 | µA |
| **Switches** | | | | | | |
| | XYZRAMP impedance | | – | 33 | 100 | Ω |
| | XYZRAMP leakage | | –10 | | +10 | µA |
| | XYDACBIAS impedance | | – | 13 | 25 | Ω |
| | XYDACBIAS leakage | | –10 | | +10 | µA |
| | XYSOURCE impedance | | – | 200 | 400 | Ω |
| | XYSOURCE leakage | | –10 | | +70 | µA |

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C, $V_{CC}$ = 5V ±10%, $V_{SS}$ = 0V[4]

| SYMBOL | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | Oscillator frequency: | | | 3.5 | 12 | MHz |
| **External Clock (Figure 3)** | | | | | | |
| $t_{CHCX}$ | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | Fall time | | 20 | | 20 | ns |

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

C – Clock
D – Input data
H – Logic level high
L – Logic level low
Q – Output data
T – Time
V – Valid
X – No longer a valid logic level
Z – Float

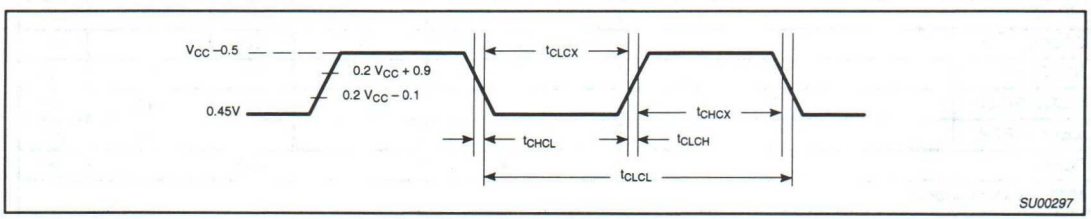



**Figure 3. External Clock Drive**

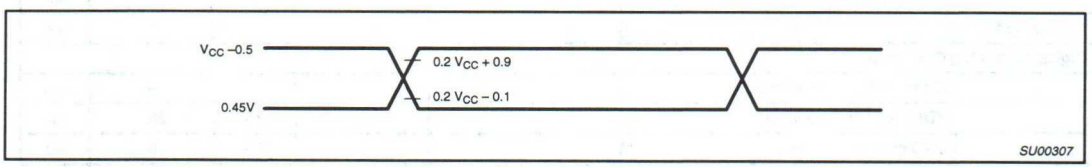



**Figure 4. AC Testing Input/Output**

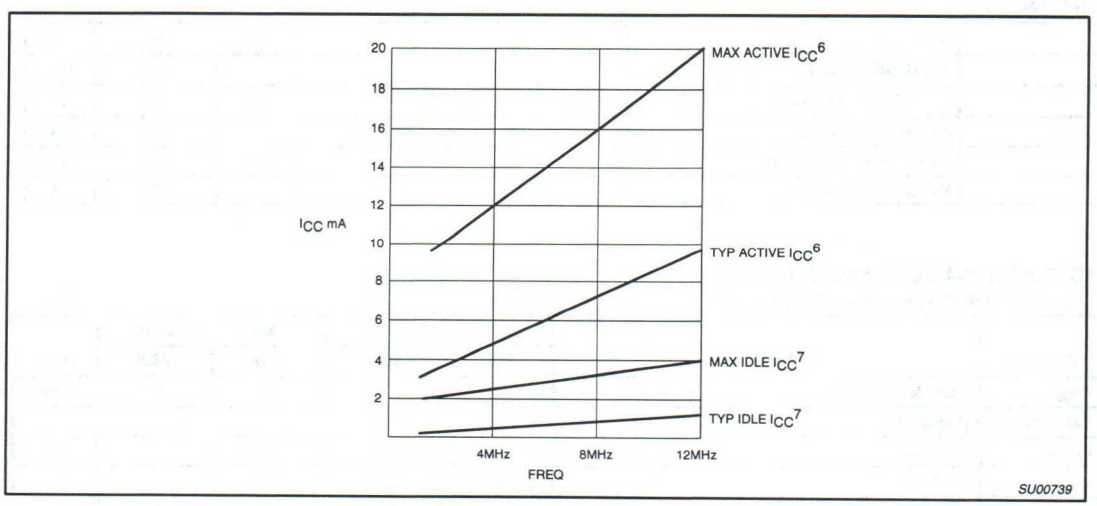



**Figure 5. $I_{CC}$ vs. FREQ**
**Maximum $I_{CC}$ values taken at $V_{CC}$ = 5.5V and worst case temperature.**
**Typical $I_{CC}$ values taken at $V_{CC}$ = 5.0V and 25°C.**
**Notes 6 and 7 refer to AC Electrical Characteristics.**

# Low voltage 8-bit microcontrollers P83CL781; P83CL782

## 1 FEATURES

- Full static 80C51 CPU
- 8-bit CPU, ROM, RAM, I/O in a 40 lead DIP or 44 lead QFP package
- 16 kbytes ROM, expandable externally to 64 kbytes
- 256 bytes RAM, expandable externally to 64 kbytes
- Four 8-bit ports, 32 I/O lines
- Three 16-bit timer/event counters
- External memory expandable up to 128 kbytes external ROM up to 64 kbytes and/or RAM up to 64 kbytes
- On-chip oscillator suitable for RC, LC, quartz crystal or ceramic resonator
- Fifteen source, fifteen vector interrupt structure with two priority levels
- Full duplex serial port (UART)
- I²C-bus interface for serial transfer on two lines
- Enhanced architecture with:
  - non-page oriented instructions
  - direct addressing
  - four 8 byte RAM register banks
  - stack depth limited only by available internal RAM (maximum 256 bytes)
  - multiply, divide, subtract and compare instructions
- Power-down and Idle modes
- Wake-up via external interrupts at Port 1
- Single supply voltage of 1.8 to 6.0 V
- Frequency range of DC to 12 MHz
- Very low current consumption
- Operating temperature:
  - 83CL781: –40 to +85 °C
  - 83CL782: –25 to +55 °C.

## 2 GENERAL DESCRIPTION

The term P83CL781 is used throughout this data sheet to refer to both the P83CL781 and P83CL782; differences between the devices are highlighted in the text.

The P83CL781 is manufactured in an advanced CMOS technology. The instruction set of the P83CL781 is based on that of the 8051. The P83CL781 is an 8-bit general purpose microcontroller especially suited for cordless telephone applications. The device has low power consumption and a wide range of supply voltage. For emulation purposes, the P85CL781 (Piggy-back version) with 256 bytes of RAM is recommended. The P83CL781 has two software selectable modes of reduced activity for further power reduction: Idle and Power-down. The P83CL781 also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 46 two-byte, and 16 three-byte.

The P83CL782 is a faster version of the P83CL781 and operates at a maximum frequency of 12 MHz at $V_{DD} \geq 3.1$ V.

## 3 ORDERING INFORMATION

| TYPE NUMBER | PACKAGE | | |
|---|---|---|---|
| | NAME | DESCRIPTION | VERSION |
| P83CL781HFP | DIP40 | plastic dual in-line package; 40 leads (600 mil) | SOT129-1 |
| P83CL782HDP | | | |
| P83CL781HFH | QFP44[1] | plastic quad flat package; 44 leads (lead length 2.35 mm); body 14 × 14 × 2.2 mm | SOT205-1 |
| P83CL782HDH | | | |
| P83CL781HFH | QFP44[1] | plastic quad flat package; 44 leads (lead length 1.3 mm); body 10 × 10 × 1.75 mm | SOT307-2 |
| P83CL781HDH | | | |

**Note**

1. When using IR reflow soldering it is recommended that the Dry Packing instructions in the *"Quality Reference Pocketbook"* (order number 9398 510 34011) are followed.

## 4 BLOCK DIAGRAM

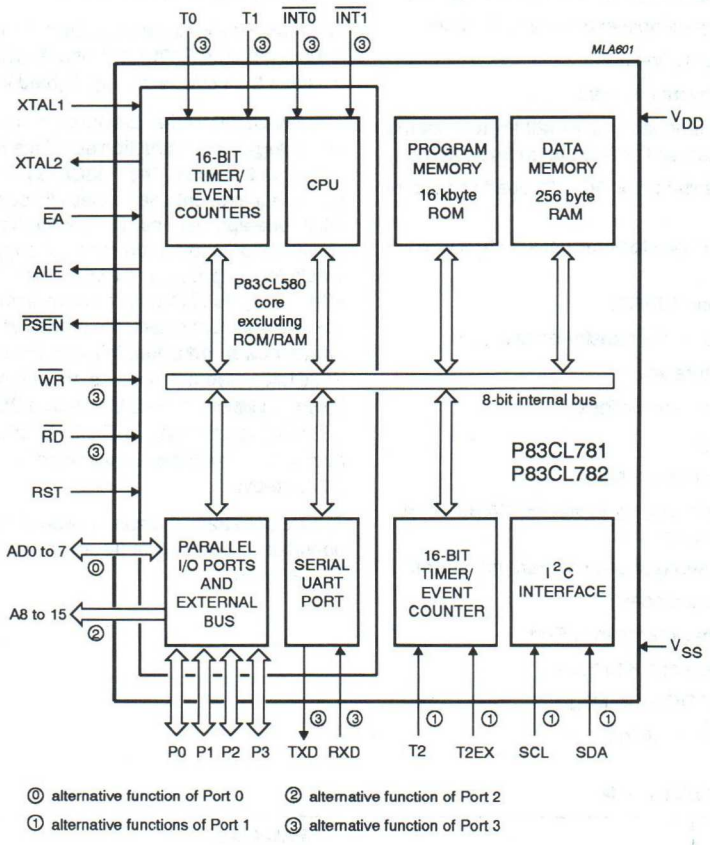



Fig.1 Block diagram.

## 5 PINNING INFORMATION

### 5.1 Pinning

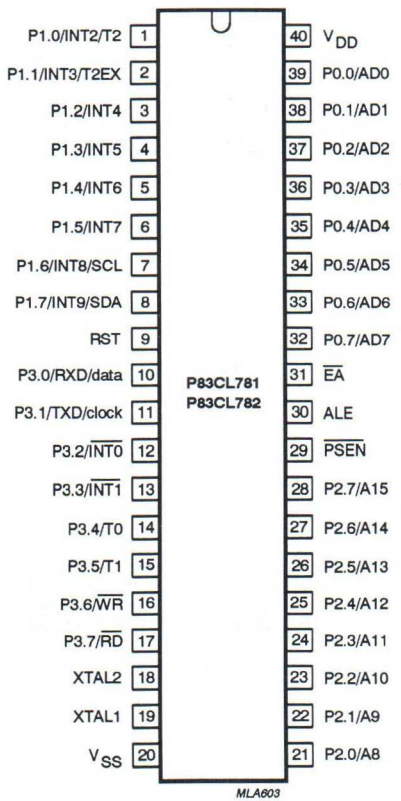



Fig.2 Pin configuration for DIP package.

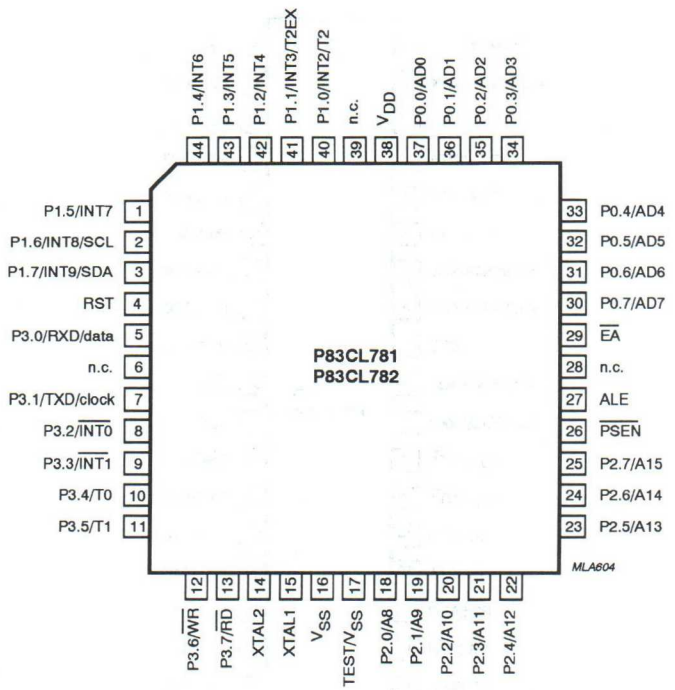



Fig.3 Pin configuration for QFP packages.

### 5.2 Pin description

**Table 1** QFP packages (SOT205 and SOT307)

| SYMBOL | PIN | DESCRIPTION |
|---|---|---|
| P1.0/INT2/T2 | 40 | **Port 1:** 8-bit bidirectional I/O port with alternative functions. Port pins that have logic 1s written to them are pulled HIGH by internal pull-ups, and in this state can be used as inputs. As inputs, Port 1 pins that are externally pulled LOW will source current ($I_{IL}$) due to the internal pull-ups. Port 1 output buffers can sink/source 4 LS TTL loads. Port 1 also serves the alternative functions INT2 to INT9 and Timer T2 external input. |
| P1.1/INT3/T2EX | 41 | |
| P1.2/INT4 | 42 | |
| P1.3/INT5 | 43 | |
| P1.4/INT6 | 44 | |
| P1.5/INT7 | 1 | |
| P1.6/INT8/SCL | 2 | |
| P1.7/INT9//SDA | 3 | |
| RST | 4 | **Reset:** A HIGH level on this pin for two machine cycles while the oscillator is running, resets the device. |
| n.c. | 6 | Not connected. |
| P3.0/RXD/data | 5 | **Port 3:** 8-bit bidirectional I/O port with alternative functions. Port pins that have logic 1s written to them are pulled HIGH by internal pull-ups, and in this state can be used as inputs. As inputs, port pins that are externally pulled LOW will source current ($I_{IL}$) due to the internal pull-ups. Port 3 output buffers can sink/source 4 LS TTL loads. RXD/data is the serial port receiver data input (asynchronous) or data I/O (synchronous). TXD/clock is the serial port transmitter data output (asynchronous) or clock output (synchronous). $\overline{INT0}$ and $\overline{INT1}$ are external interrupt lines. T0 and T1 are external inputs for Timer 0 and Timer 1 respectively. $\overline{WR}$ is the external memory write strobe and $\overline{RD}$ is the external memory read strobe. |
| P3.1/TXD/clock | 7 | |
| P3.2/$\overline{INT0}$ | 8 | |
| P3.3/$\overline{INT1}$ | 9 | |
| P3.4/T0 | 10 | |
| P3.5/T1 | 11 | |
| P3.6/$\overline{WR}$ | 12 | |
| P3.7/$\overline{RD}$ | 13 | |
| XTAL2 | 14 | **Crystal Output:** Output of the inverting amplifier that forms the oscillator. Left open-circuit when an external oscillator clock is used. |
| XTAL1 | 15 | **Crystal Input:** Input to the inverting amplifier that forms the oscillator, also the input for an externally generated clock source. |
| $V_{SS}$ | 16 | **Ground:** Circuit ground potential. |
| TEST/$V_{SS}$ | 17 | **Test Input:** Must be connected to $V_{SS}$ or left open. |
| P2.0/A8 | 18 | **Port 2:** 8-bit bidirectional I/O port with alternative functions. Port pins that have logic 1s written to them are pulled HIGH by internal pull-ups, and in this state can be used as inputs. Port 2 output buffers can sink/source 4 LS TTL loads. Port 2 emits the high order address byte during accesses to external memory that use 16-bit addresses (MOVX@DPTR). In this application it uses the strong internal pull-ups when emitting logic 1's. During accesses to external memory that use 8-bit addresses (MOVX@Ri), Port 2 emits the contents of the P2 Special Function Register. |
| P2.1/A9 | 19 | |
| P2.2/A10 | 20 | |
| P2.3/A11 | 21 | |
| P2.4/A12 | 22 | |
| P2.5/A13 | 23 | |
| P2.6/A14 | 24 | |
| P2.7/A15 | 25 | |

| SYMBOL | PIN | DESCRIPTION |
|---|---|---|
| $\overline{PSEN}$ | 26 | **Program Store Enable:** Read strobe to external program memory. When executing code out of external program memory, $\overline{PSEN}$ is activated twice each machine cycle. However, during each access to external data memory two $\overline{PSEN}$ activations are skipped. |
| ALE | 27 | **Address Latch Enable:** Latches the low byte of the address during accesses to external memory. It is activated every six oscillator periods and may be used for external timing or clocking purposes. |
| n.c. | 28 | Not connected. |
| $\overline{EA}$ | 29 | **External Access:** When $\overline{EA}$ is held HIGH, the CPU executes out of the internal program memory (unless the Program Counter exceeds 3FFFH). When $\overline{EA}$ is held LOW, the CPU executes out of external program memory regardless of the value of the program counter. |
| P0.7/AD7 | 30 | **Port 0:** 8-bit open drain bidirectional I/O port with alternative functions. P0.7 to P0.0 provide the 8-bit I/O port. As an open-drain output port it can sink/source 8 LS TTL loads. Port 0 pins that have logic 1s written to them float, and in this state will function as high-impedance inputs. AD7 to AD0 provide the multiplexed low-order address and data bus during accesses to external memory. In this application it uses the strong internal pull-ups when emitting logic 1s. |
| P0.6/AD6 | 31 | |
| P0.5/AD5 | 32 | |
| P0.4/AD4 | 33 | |
| P0.3/AD3 | 34 | |
| P0.2/AD2 | 35 | |
| P0.1/AD1 | 36 | |
| P0.0/AD0 | 37 | |
| $V_{DD}$ | 38 | **Power supply** |
| n.c. | 39 | Not connected. |

## 6 FUNCTIONAL DESCRIPTION

### 6.1 General

The P83CL781 is a stand-alone high-performance CMOS microcontroller designed for use in real-time applications such as instrumentation, industrial control, intelligent computer peripherals and consumer products. The device provides hardware features, architectural enhancements and new instructions to function as a controller for applications requiring up to 64 kbytes of program memory and/or up to 64 kbytes of data storage.

The P83CL781 contains a non-volatile 16 kbyte read-only program memory; a static 256 byte read/write data memory; 32 I/O lines; three 16-bit timer/event counters; a fifteen-source two priority-level, nested interrupt structure and on-chip oscillator and timing circuit.

The device has two software selectable modes of reduced activity for power reduction; Idle and Power-down. The Idle mode freezes the CPU while allowing the RAM, timers, serial I/O and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative.

Two serial interfaces are provided on-chip; a standard UART serial interface and an I²C-bus serial interface. The I²C-bus serial interface has byte orientated master and slave functions allowing communication with the whole family of I²C-bus compatible devices.

### 6.2 CPU timing

A machine cycle consists of a sequence of 6 states. Each state lasts for two oscillator periods, thus a machine cycle takes 12 oscillator periods or 1 μs if the oscillator frequency is 12 MHz.

### 6.3 Memory organization

The P83CL781 has a 16 kbyte Program Memory (ROM) plus 256 bytes of Data Memory (RAM) on-chip. The device has separate address spaces for Program and Data Memory (see Fig.4). Using Ports P0 and P2, the P83CL781 can address up to 64 kbytes of external memory. The CPU generates both read ($\overline{RD}$) and write ($\overline{WR}$) signals for external Data Memory accesses, and the read strobe ($\overline{PSEN}$) for external Program Memory.

#### 6.3.1 PROGRAM MEMORY

The P83CL781 contains 16 kbytes of internal ROM. After reset the CPU begins execution at location 0000H. The lower 16 kbytes of Program Memory can be implemented in either on-chip ROM or external memory. If the $\overline{EA}$ pin is strapped to $V_{DD}$, then program memory fetches from addresses 0000H through to 3FFFH are directed to the internal ROM. Fetches from addresses 4000H through to FFFFH are directed to external ROM. Program Counter values greater than 3FFFH are automatically addressed to external memory regardless of the state of the $\overline{EA}$ pin.

#### 6.3.2 DATA MEMORY

The P83CL781 contains 256 bytes of internal RAM and 34 Special Function Registers (SFRs). Figure 4 shows the internal Data Memory space divided into the lower 128 bytes the upper 128 bytes and the SFR space. Internal RAM locations 0 to 27 are directly and indirectly addressable. Internal RAM locations 128 to 255 are only indirectly addressable. The Special Function Register locations 128 to 255 bytes are only directly addressable.

#### 6.3.3 SPECIAL FUNCTION REGISTERS

The upper 128 bytes are the address locations of the Special Function Registers. Figures 6 and 7 show the Special Function Registers space. The SFRs include the port latches, timers, peripheral control, serial I/O registers, etc. These registers can only be accessed by direct addressing. There are 128 addressable locations in the SFR address space (SFRs with addresses divisible by eight).

### 6.4 Addressing

The P83CL781 has five methods for addressing source operands:

- Register
- Direct
- Register-Indirect
- Immediate
- Base-Register plus Index-Register-Indirect.

The first three methods can be used for addressing destination operands. Most instructions have a 'destination/source' field that specifies the data type, addressing methods and operands involved. For operations other than MOVs, the destination operand is also a source operand.

Access to memory addressing is as follows:

- Registers in one of the four 8-register banks through Register Direct or Register-Indirect
- 256 bytes of internal data RAM through Direct or Register-Indirect
- Special Function Registers through Direct
- External data memory through Register-Indirect
- Program memory look-up tables through Base-Register plus Index-Register-Indirect.

The P83CL781 is classified as an 8-bit device since the internal ROM, RAM, Special Function Registers, Arithmetic Logic Unit and external data bus are all 8-bits wide. It performs operations on bit, nibble, byte and double-byte data types.

Facilities are available for byte transfer, logic and integer arithmetic operations. Data transfer, logic and conditional branch operations can be performed directly on Boolean variables to provide excellent bit handling.

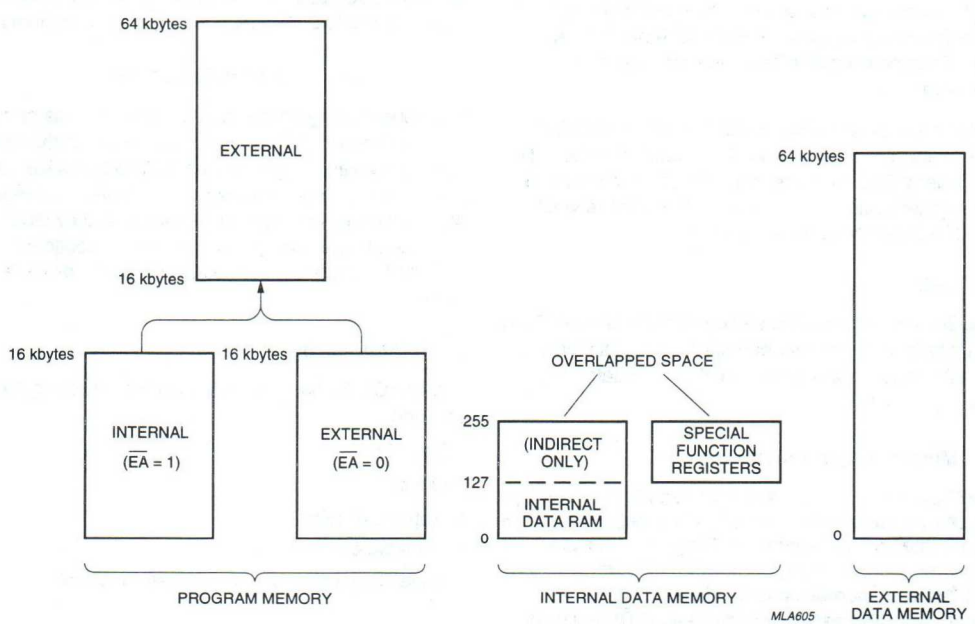



Fig.4 Memory map.

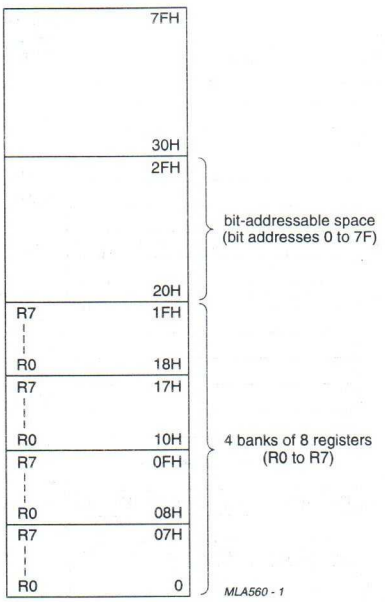



Fig.5 The lower 128 bytes of internal RAM.

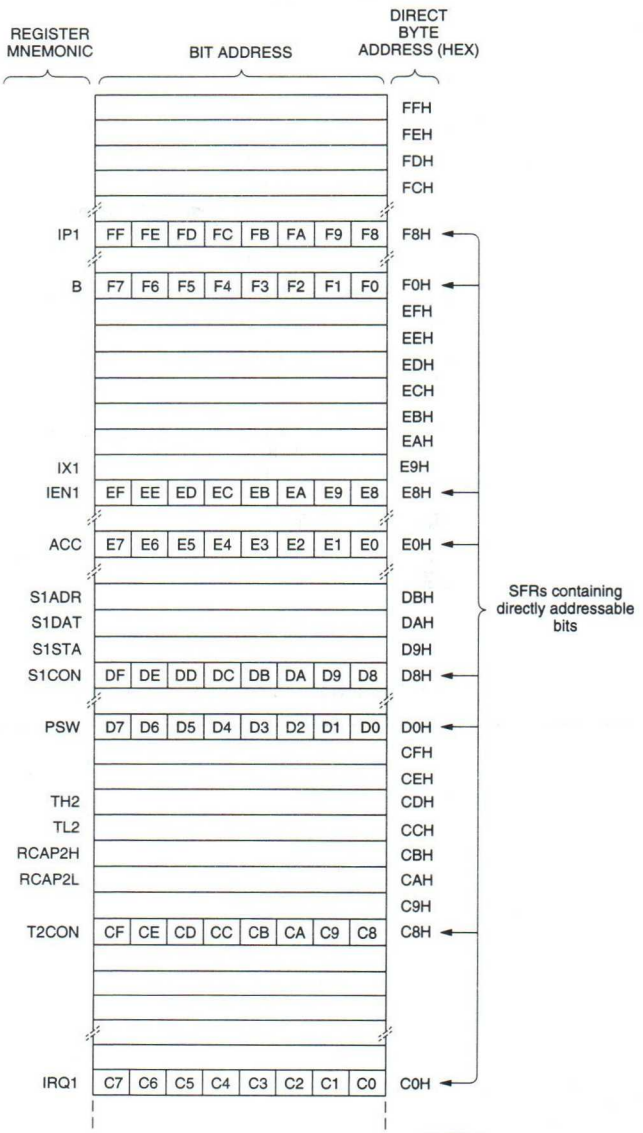



Fig.6 Special Function Register memory map (continued in Fig.7).

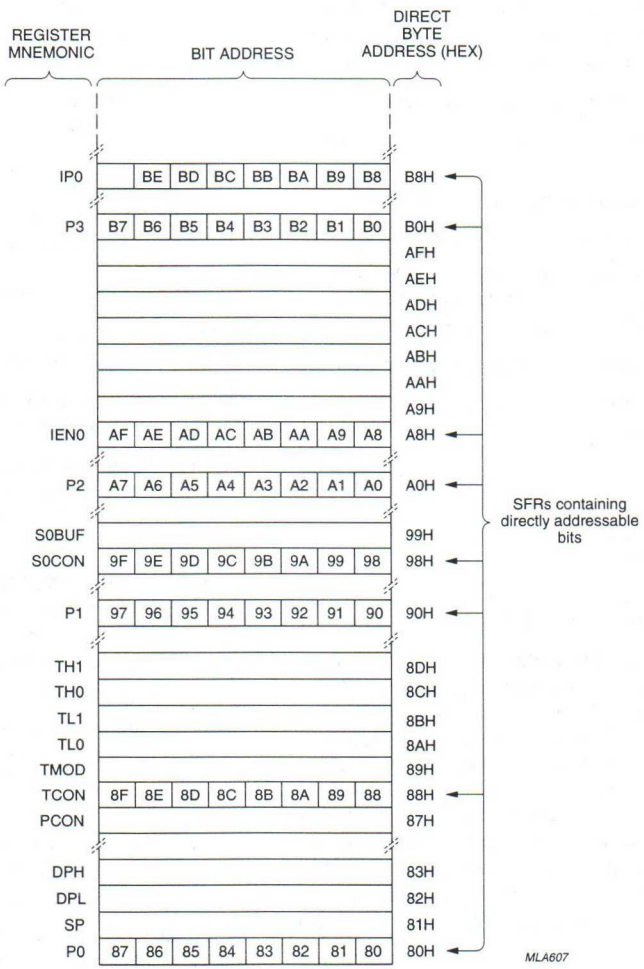



Fig.7 Special Function Register memory map (continued from Fig.6).

## 6.5 I/O facilities

### 6.5.1 PORTS

The P83CL781 has 32 I/O lines treated as 32 individually addressable bits or as four parallel 8-bit addressable ports. Ports 0, 1, 2 and 3 perform the following alternative functions:

Port 0 Provides the multiplexed low-order address and data bus for expanding the device with standard memories and peripherals.

Port 1 Used for a number of special functions:

- Provides the inputs for the external interrupts INT2 to INT9
- External counter/capture of Timer 2
- SCL and SDA for the $I^2C$-bus interface.

Port 2 Provides the high-order address bus when expanding the device with external program memory and/or external data memory.

Port 3 Pins can be configured individually to provide:

- External interrupt request inputs
- Counter inputs
- Serial port receiver input and transmitter output (UART)
- Control signals to read and write to external memories.

To enable a Port 3 pin alternative function, the Port 3 bit latch in its SFR must contain a logic 1.

Each port consists of a latch (Special Function Registers P0 to P3), an output driver and input buffer. Ports 1, 2 and 3 have internal pull-ups. Figure 8a shows that the strong transistor p1 is turned on for only 2 oscillator periods after a LOW-to-HIGH transition in the port latch. When on, it turns on p3 (a weak pull-up) through the inverter. This inverter and p3 form a latch which holds the logic 1. In Port 0 the pull-up p1 is only on when emitting logic 1s for external memory access. Writing a logic 1 to a Port 1 bit latch leaves both output transistors switched off so that the pin can be used as an high-impedance input.

### 6.5.2 PORT OPTIONS

30 of the 32 port pins (excluding P1.6 and P1.7 with option 2S only) may be individually configured with one of the following options. These options are also shown in Fig.8.

Option 1 Standard Port; quasi-bidirectional I/O with pull-up. The strong booster pull-up p1 is turned on for two oscillator periods after a LOW-to-HIGH transition in the port latch (see Fig.8a).

Option 2 Open drain; quasi-bidirectional I/O with n-channel open drain output. Use as an output requires the connection of an external pull-up resistor (see Fig.8c).

Option 3 Push-Pull; output with drive capability in both polarities. Under this option, pins can only be used as outputs (see Fig.8b).

The definition of port options for Port 0 is slightly different. Two cases are examined. First, access to external memory ($\overline{EA}$ = 0 or access above the built-in memory boundary) and second, I/O accesses.

#### *6.5.2.1 External Memory Accesses*

Option 1 True 0 and 1 are written as address to the external memory (strong pull-up to be used).

Option 2 An external pull-up resistor is required for external accesses.

Option 3 Not allowed for external memory accesses as the port can only be used as output.

#### *6.5.2.2 I/O Accesses*

Option 1 When writing a logic 1 to the port latch, the strong pull-up p1 will be on for 2 oscillator periods. No weak pull-up exists. Without an external pull-up, this option can be used as a high-impedance input.

Option 2 Open drain; quasi-directional I/O with n-channel open drain output. Use as an output requires the connection of an external pull-up resistor. See Fig.8c.

Option 3 Push-Pull; output with drive capability in both polarities. Under this option pins can only be used as outputs.

Individual mask selection of the post-reset state is available with any of the above pins. The required selection is made by appending 'R' or 'S' to options 1, 2, or 3 above.

Option R RESET, after reset this pin will be initialized LOW.

Option S SET, after reset this pin will be initialized HIGH.

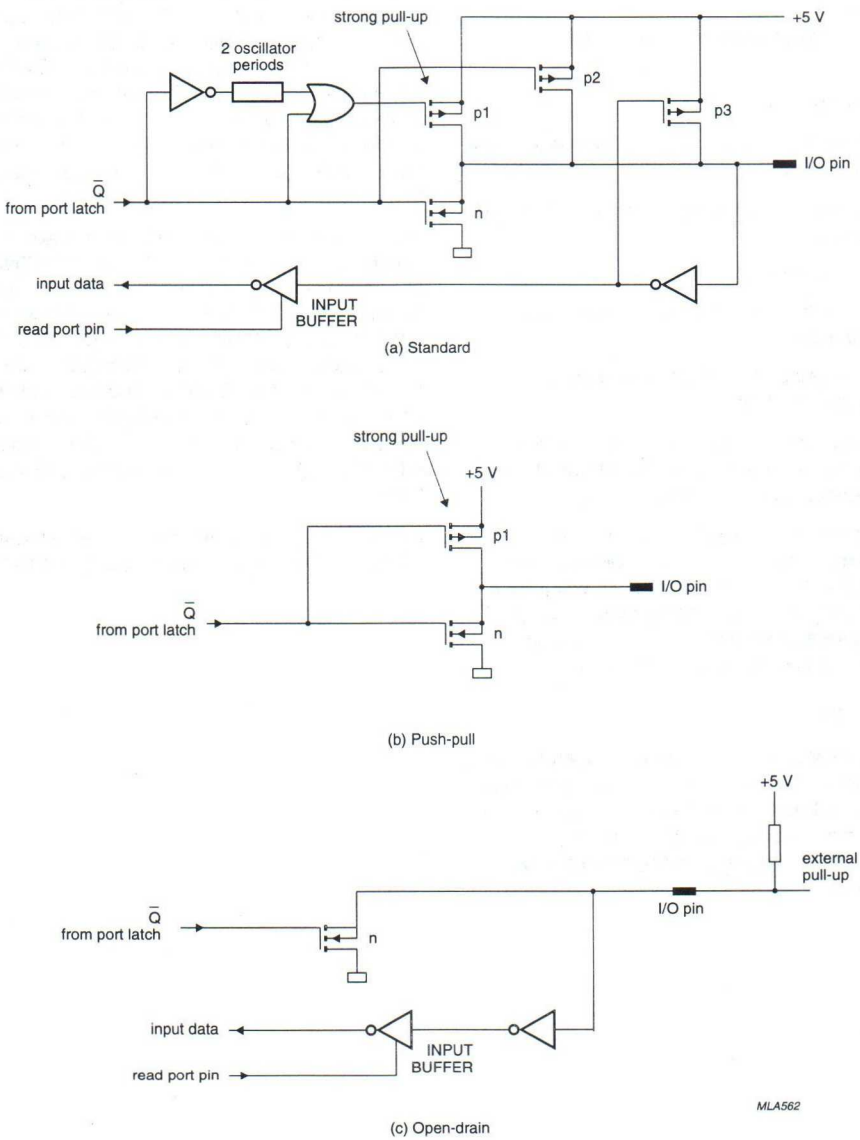



Fig.8 Port configuration options.

### 6.6 Timer/event counters

The P83CL781 contains three 16-bit timer/event counters: Timer 0, Timer 1 and Timer 2 which can perform the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests.

Timer 0 and Timer 1 can be programmed independently to operate in four modes:

Mode 0 8-bit timer or 8-bit counter each with divide-by-32 prescaler.

Mode 1 16-bit time-interval or event counter.

Mode 2 8-bit time-interval or event counter with automatic reload upon overflow.

Mode 3 Timer 0 establishes TL0 and TH0 as two separate counters.

In the 'timer' mode the register is incremented every machine cycle. Since a machine cycle consists of 12 oscillator periods, the count rate is $^1/_{12}f_{osc}$.

In the 'counter' mode, the register is incremented in response to a HIGH-to-LOW transition. Since it takes 2 machine cycles (24 oscillator periods) to recognize a HIGH-to-LOW transition, the maximum count rate is $^1/_{24}f_{osc}$. To ensure a given level is sampled, it should be held for at least one complete machine cycle.

#### 6.6.1 Timer T2

Timer T2 is a 16-bit timer/counter that can operate either as a timer or as an event counter. These functions are selected by the state of the C/$\overline{T2}$ bit in the T2CON register. Three operating modes are available Capture, Auto-Reload and Baud rate generator, these are also selected via the T2CON register.

In the Capture Mode, two options may be selected by the EXEN2 bit in T2CON. If EXEN2 = 0, then Timer 2 is a 16-bit timer or counter which upon overflowing sets the Timer 2 overflow bit TF2, this may then be used to generate an interrupt. If EXEN2 = 1, Timer 2 operates as described above but with the additional feature that a HIGH-to-LOW transition at external input T2EX causes the current value in TL2 and TH2 to be captured into registers RCAP2L and RCAP2H respectively. In addition, the transition at T2EX causes the EXF2 bit in T2CON to be set; this may also be used to generate an interrupt. The Capture Mode is shown in Fig.9.

In the Auto-Reload Mode there are also two options selected by the EXEN2 bit in T2CON. If EXEN2 = 0, then when Timer 2 rolls over, it sets the TF2 bit but also causes the Timer 2 registers to be reloaded with the 16-bit value held in registers RCAP2L and RCAP2H. The 16-bit value held in these registers is preset by software. If EXEN2 = 1, Timer 2 operates as described above but with the additional feature that a HIGH-to-LOW transition at external input T2EX will also trigger the 16-bit reload and set the EXF2 bit. The Auto-Reload Mode is shown in Fig.10.

The Baud rate generator Mode is selected when RTCLK = 1. This is described in Section 6.10.

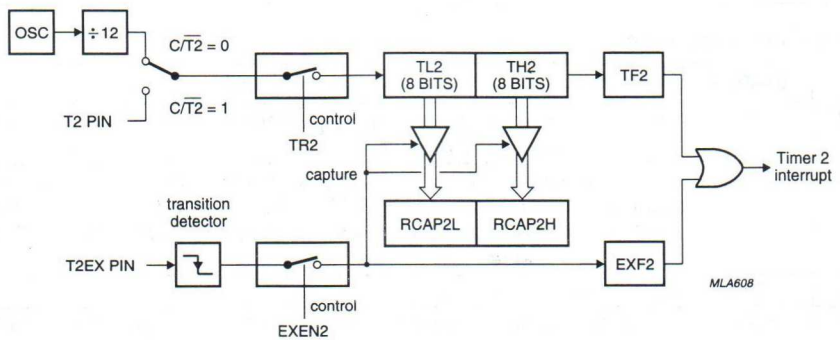



Fig.9 Timer 2 in Capture Mode.

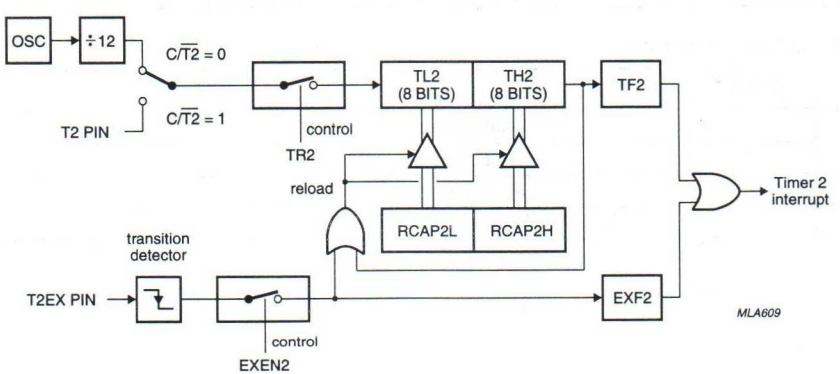



Fig.10 Timer 2 in Auto-Reload Mode.

### 6.6.2 Timer/Counter 2 Control Register (T2CON)

**Table 2** Timer/Counter 2 Control Register (SFR address C8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| TF2 | EXF2 | GF2 | RTCLK | EXEN2 | TR2 | C/$\overline{T2}$ | CP/$\overline{RL2}$ |

**Table 3** Description of T2CON bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| T2CON.7 | TF2 | Timer 2 overflow flag, set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when RTCLK = 1. |
| T2CON.6 | EXF2 | Timer 2 external flag is set when either a capture or reload is caused by a negative transition on T2EX and when EXEN2 = 1. When Timer T2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to Timer 2 interrupt routine. EXF2 must be cleared by software. |
| T2CON.5 | GF2 | General purpose flag bit. |
| T2CON.4 | RTCLK | Transmit clock flag. When set, causes the UART serial port to use Timer 2 overflow pulses for its receive and transmit clock in Modes 1 and 3. RTCLK = 0 causes Timer 1 overflows to be used for the receive and transmit clock. |
| T2CON.3 | EXEN2 | Timer 2 external enable flag. When set, allows a capture or reload to occur as a result of a negative transition on T2EX, if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX. |
| T2CON.2 | TR2 | Start/Stop control for Timer 2. TR2 = 1 starts the timer. |
| T2CON.1 | C/$\overline{T2}$ | Timer or counter select for Timer 2. C/$\overline{T2}$ = 0 selects the internal timer with a clock frequency of $1/12 f_{osc}$. C/$\overline{T2}$ = 1 selects the external event counter; negative edge-triggered. |
| T2CON.0 | CP/$\overline{RL2}$ | Capture/reload flag. When set captures will occur on negative transitions at T2EX, if EXEN2 = 1. When cleared, auto-reloads will occur either with Timer 2 overflows or negative transitions at T2EX when EXEN2 = 1. When RTCLK = 1 this bit is ignored and the timer is forced to auto-reload on a Timer 2 overflow. |

**Table 4** Timer 2 operating modes

| RTCLK | CP/$\overline{RL2}$ | TR2 | MODE |
|---|---|---|---|
| 0 | 0 | 1 | 16-bit auto-reload |
| 0 | 1 | 1 | 16-bit capture |
| 1 | X | 1 | baud rate generator |
| X | X | 0 | OFF |

## 6.7 Idle and Power-down operation

Idle mode operation permits the interrupt, serial ports and timer blocks to continue to function while the clock to the CPU is halted.

The following functions remain active during the Idle mode. These functions may generate an interrupt or reset; thus ending the Idle mode.

- Timer 0, Timer 1 and Timer 2
- SIO, $I^2C$-bus interface
- External interrupt.

The Power-down operation freezes the oscillator. The Power-down mode can only be activated by setting the PD bit in the PCON register. The Idle and Power-down clock configuration is shown in Fig.11.

### 6.7.1 IDLE MODE

The instruction that sets PCON.0 is the last instruction executed in the normal operating mode before the Idle mode is activated. Once in the Idle mode, the CPU status is preserved along with the Stack Pointer, Program Counter, Program Status Word and Accumulator. The RAM and all other registers maintain their data during Idle mode. The status of the external pins during Idle mode is shown in Table 5.

There are two ways to terminate the Idle mode:

1. Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware thus terminating the Idle mode. The interrupt is serviced, and following the RETI instruction, the next instruction to be executed will be the one following the instruction that put the device in the Idle mode. The flag bits GF0 and GF1 may be used to determine whether the interrupt was received during normal execution or during the Idle mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When the Idle mode is terminated by an interrupt, the service routine can examine the status of the flag bits.
2. The second way of terminating the Idle mode is with an external hardware reset, or an internal reset caused by an overflow of Timer T2. Since the oscillator is still running, the hardware reset is required to be active for two machine cycles (24 oscillator periods) to complete the reset operation. Reset redefines all SFRs but does not affect the on-chip RAM.

### 6.7.2 POWER-DOWN MODE

The instruction that sets PCON.1 is the last executed prior to going into the Power-down mode. Once in the Power-down mode, the oscillator is stopped. The contents of the on-chip RAM and the SFRs are preserved. The port pins output the value held by their respective SFRs. ALE and $\overline{PSEN}$ are held LOW.

In the Power-down mode, $V_{DD}$ may be reduced to minimize circuit power consumption. The supply voltage must not be reduced until the Power-down mode is entered, and must be restored before the hardware reset is applied which will free the oscillator. Reset should not be released until the oscillator has restarted and stabilized.

### 6.7.3 WAKE-UP MODE

Setting the PD flag in the PCON register forces the controller into the Power-down mode. Setting this flag enables the controller to be woken-up from the Power-down mode with either the external interrupts INT2 to INT9, or a reset operation.

#### *6.7.3.1 Wake-up using INT2 to INT9*

If any of the interrupts INT2 to INT9 are enabled, the device can be woken-up from the Power-down mode with the external interrupts. To ensure that the oscillator is stable before the controller restarts, the internal clock will remain inactive for 1536 oscillator periods. This is controlled by an on-chip delay counter.

#### *6.7.3.2 Wake-up using RST*

To wake-up the P83CL781, the RST pin must be kept HIGH for a minimum of 24 periods. The on-chip delay counter is inactive. The user must ensure that the oscillator is stable before any operation is attempted. Figure 12 illustrates the two possibilities for wake-up.

### 6.7.4 STATUS OF EXTERNAL PINS

The status of the external pins during Idle and Power-down mode is shown in Table 5. If the Power-down mode is activated whilst accessing external program memory, the port data that is held in the Special Function Register P2 is restored to Port 2. If the data is a logic 1, the port pin is held HIGH during the Power-down mode by the strong pull-up transistor p1 (see Fig.8a).

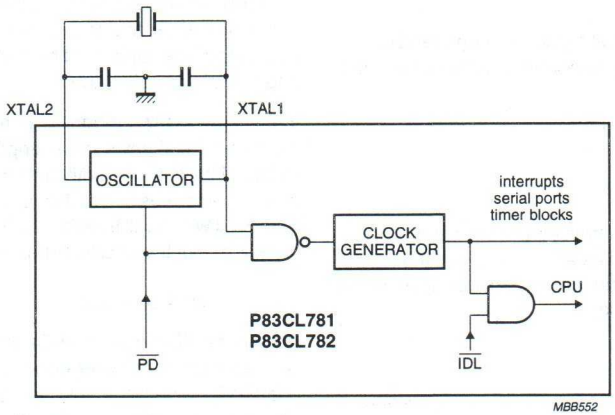



Fig.11 Internal Idle and Power-down clock configuration.

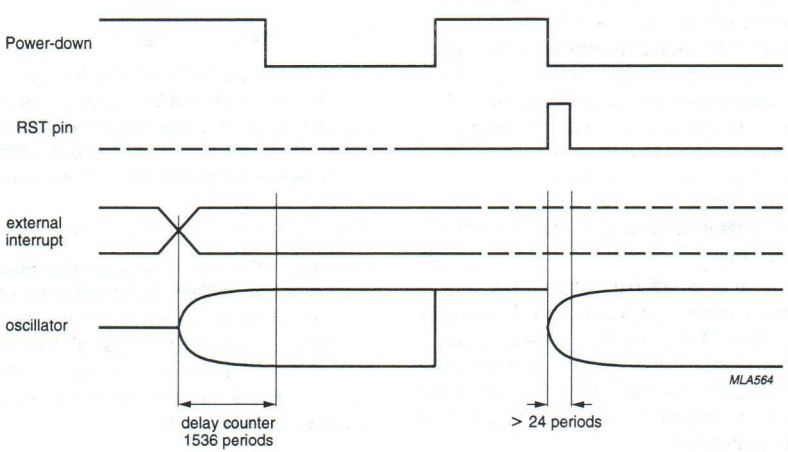



Fig.12 Wake-up operation.

**Table 5** Status of external pins during Idle and Power-down modes

| MODE | MEMORY | ALE | $\overline{\text{PSEN}}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | internal | 1 | 1 | port data | port data | port data | port data |
| Idle | external | 1 | 1 | floating | port data | address | port data |
| Power-down | internal | 0 | 0 | port data | port data | port data | port data |
| Power-down | external | 0 | 0 | floating | port data | port data | port data |

### 6.7.5 Power Control Register (PCON)

The reduced power modes are activated by software using this Special Function Register. PCON is not bit addressable.

**Table 6** Power Control Register (SFR address 87H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SMOD | – | – | – | GF1 | GF0 | PD | IDL |

**Table 7** Description of PCON bits.

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| PCON.7 | SMOD | Double Baud rate bit. When set to a logic 1 the baud rate is doubled when the serial port SIO0 is being used in modes 1, 2 or 3. |
| PCON.6 | – | Reserved |
| PCON.5 | – | Reserved |
| PCON.4 | – | Reserved |
| PCON.3 | GF1 | General purpose flag bit |
| PCON.2 | GF0 | General purpose flag bit |
| PCON.1 | PD | Power-down bit. Setting this bit activates the Power-down mode; see note 1. |
| PCON.0 | IDL | Idle mode bit. Setting this bit activates the Idle mode; see note 1. |

**Note**

1. If logic 1s are written to PD and IDL at the same time, PD takes precedence. The reset value of PCON is (0XX00000).

### 6.8 I²C-bus serial I/O

The serial port supports the twin line I²C-bus. The I²C-bus consists of two lines: a data line (SDA) and a clock line (SCL). These lines also function as the I/O port lines P1.7 and P1.6 respectively. The system is unique because data transport, clock generation, address recognition and bus control arbitration are all controlled by hardware. The I²C-bus serial I/O has complete autonomy in byte handling and operates in 4 modes:

- Master transmitter
- Master receiver
- Slave transmitter
- Slave receiver.

These functions are controlled by the S1CON register. S1STA is the Status Register whose contents may also be used as a vector to various service routines. S1DAT is the Data Shift Register and S1ADR the Slave Address Register. Slave address recognition is performed by on-chip hardware. The block diagram of the I²C-bus serial I/O is shown in Fig.13.

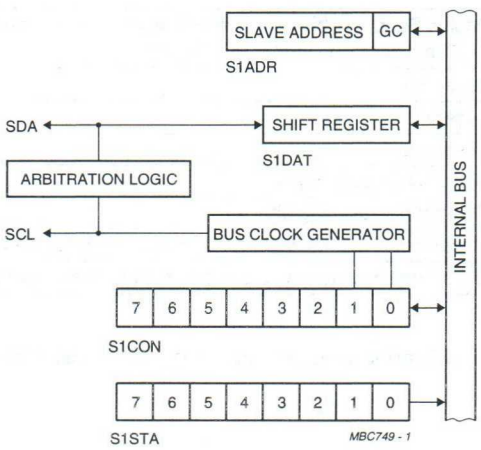



Fig.13 Block diagram of I²C-bus serial I/O.

### 6.8.1 Serial Control Register (S1CON)

**Table 8** Serial Control Register (SFR address D8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| CR2 | ENS1 | STA | STO | SI | AA | CR1 | CR0 |

**Table 9** Description of S1CON bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| S1CON.7 | CR2 | This bit along with bits CR1 and CR0 determines the serial clock frequency when SIO is in the Master Mode. See Table 10. |
| S1CON.6 | ENS1 | ENABLE serial I/O. When ENS1 = 0, the serial I/O is disabled. SDA and SCL outputs are in the high-impedance state; P1.6 and P1.7 function as open-drain ports. When ENS1 = 1, the serial I/O is enabled. Output port latches P1.6 and P1.7 must be set to logic 1. |
| S1CON.5 | STA | START flag. When this bit is set in Slave Mode, the SIO hardware checks the status of the $I^2C$-bus and generates a START condition if the bus is free or after the bus becomes free. If STA is set while the SIO is in Master Mode, SIO will generate a repeated START condition. |
| S1CON.4 | STO | STOP flag. With this bit set while in Master Mode a STOP condition is generated. When a STOP condition is detected on the $I^2C$-bus, the SIO hardware clears the STO flag. STO may also be set in Slave Mode in order to recover from an error condition. In this case no STOP condition is transmitted to the $I^2C$-bus. However, the SIO hardware behaves as if a STOP condition has been received and releases the SDA and SCL. The SIO then switches to the not addressed slave receiver mode. The STOP flag is cleared by the hardware. |
| S1CON.3 | SI | SIO interrupt flag. This flag is set, and an interrupt is generated, after any of the following events occur:<br>• A start condition is generated in Master Mode.<br>• Own slave address has been received during AA = 1.<br>• The general call address has been received while S1ADR0 and AA = 1.<br>• A data byte has been received or transmitted in Master Mode (even if arbitration is lost).<br>• A data byte has been received or transmitted as selected slave.<br>• A Stop or Start condition is received as selected slave receiver or transmitter. |
| S1CON.2 | AA | Assert Acknowledge. When this bit is set, an acknowledge (LOW level to SDA) is returned during the acknowledge clock pulse on the SCL line when:<br>• Own slave address is received.<br>• General call address is received (S1ADR.0 = 1).<br>• A data byte is received while the device is programmed to be a Master Receiver.<br>• A data byte is received while the device is a selected Slave Receiver.<br>When this bit is reset, no acknowledge is returned. Consequently, no interrupt is requested when the own slave address or general call address is received. |
| S1CON.1 | CR1 | These two bits along with the CR2 bit determines the serial clock frequency when SIO is in the Master Mode. See Table 10. |
| S1CON.0 | CR0 | |

**Table 10** Selection of the serial clock frequency in the master mode of operation

| CR2 | CR1 | CR0 | $f_{osc}$ DIVISOR | BIT RATE (kHz) at $f_{osc}$ | | |
|---|---|---|---|---|---|---|
| | | | | 3.58 MHz | 6 MHz | 12 MHz |
| 0 | 0 | 0 | 256 | 14.0 | 23.4 | 46.9 |
| 0 | 0 | 1 | 224 | 16.0 | 26.8 | 53.6 |
| 0 | 1 | 0 | 192 | 18.6 | 31.3 | 62.5 |
| 0 | 1 | 1 | 160 | 22.4 | 37.5 | 75.0 |
| 1 | 0 | 0 | 960 | 3.73 | 6.25 | 12.5 |
| 1 | 0 | 1 | 120 | 29.8 | 50.0 | 100 |
| 1 | 1 | 0 | 60 | 59.7 | 100 | – |
| 1 | 1 | 1 | not allowed | – | – | – |

### 6.8.2 Data Shift Register (S1DAT)

S1DAT contains the serial data to be transmitted or data which has just been received. Bit 7 is transmitted or received first; i.e. data shifted from left to right.

**Table 11** Data Shift Register (SFR address DAH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| S1DAT.7 | S1DAT.6 | S1DAT.5 | S1DAT.4 | S1DAT.3 | S1DAT.2 | S1DAT.1 | S1DAT.0 |

### 6.8.3 Address Register (S1ADR)

This 8-bit register may be loaded with the 7-bit slave address to which the controller will respond when programmed as a slave receiver/transmitter.

**Table 12** Address Register (SFR address DBH)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SLA6 | SLA5 | SLA4 | SLA3 | SLA2 | SLA1 | SLA0 | GC |

**Table 13** Description of S1ADR bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| S1ADR.7 to S1ADR.1 | SLA6 to 0 | Own slave address. |
| S1ADR.0 | GC | This bit is used to determine whether the general CALL address is recognized. When a logic 0, the general CALL address is not recognized. When a logic 1, the general CALL address is recognized. |

### 6.8.4 Serial Status Register (S1STA)

The contents of this register may be used as a vector to a service routine. This optimizes the response time of the software and consequently that of the I²C-bus. S1STA is a read-only register. The status codes for all possible modes of the I²C-bus interface are given in Tables 16 to 20.

**Table 14** Serial Status Register (SFR address D9H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SC4 | SC3 | SC2 | SC1 | SC0 | 0 | 0 | 0 |

**Table 15** Description of S1STA bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| S1STA.3 to S1STA.7 | SC4 to SC0 | 5-bit status code. |
| S1STA.0 to S1STA.2 | – | These three bits are held LOW. |

**Table 16** MST/TRX mode

| S1STA VALUE | DESCRIPTION |
|---|---|
| 08H | A START condition has been transmitted. |
| 10H | A repeated START condition has been transmitted. |
| 18H | SLA and W have been transmitted, ACK has been received. |
| 20H | SLA and W have been transmitted, $\overline{\text{ACK}}$ received. |
| 28H | DATA of S1DAT has been transmitted, ACK received. |
| 30H | DATA of S1DAT has been transmitted, $\overline{\text{ACK}}$ received. |
| 38H | Arbitration lost in SLA, R/W or DATA. |

**Table 17** MST/REC mode

| S1STA VALUE | DESCRIPTION |
|---|---|
| 38H | Arbitration lost while returning $\overline{\text{ACK}}$. |
| 40H | SLA and R have been transmitted, ACK received. |
| 48H | SLA and R have been transmitted, $\overline{\text{ACK}}$ received. |
| 50H | DATA has been received, ACK returned. |
| 58H | DATA has been received, $\overline{\text{ACK}}$ returned. |

**Table 18** SLV/REC mode

| S1STA VALUE | DESCRIPTION |
|---|---|
| 60H | Own SLA and W have been received, ACK returned. |
| 68H | Arbitration lost in SLA, R/W as MST. Own SLA and W have been received, ACK returned. |
| 70H | General CALL has been received, ACK returned. |
| 78H | Arbitration lost in SLA, R/W as MST. General CALL has been received. |
| 80H | Previously addressed with own SLA. DATA byte received, ACK returned. |
| 88H | Previously addressed with own SLA. DATA byte received, $\overline{\text{ACK}}$ returned. |
| 90H | Previously addressed with general CALL. DATA byte has been received, ACK has been returned. |
| 98H | Previously addressed with general CALL. DATA byte has been received, $\overline{\text{ACK}}$ has been returned. |
| A0H | A STOP condition or repeated START condition has been received while still addressed as SLV/REC or SLV/TRX. |

**Table 19** SLV/TRX mode

| S1STA VALUE | DESCRIPTION |
|---|---|
| A8H | Own SLA and R have been received, ACK returned. |
| B0H | Arbitration lost in SLA, R/W as MST. Own SLA and R have been received, ACK returned. |
| B8H | DATA byte has been transmitted, ACK received. |
| C0H | DATA byte has been transmitted, $\overline{\text{ACK}}$ received. |
| C8H | Last DATA byte has been transmitted (AA = logic 0), ACK received. |

**Table 20** Miscellaneous

| S1STA VALUE | DESCRIPTION |
|---|---|
| 00H | Bus error during MST mode or selected SLV mode, due to an erroneous START or STOP condition. |

**Table 21** Symbols used in Tables 16 to 20

| SYMBOL | DESCRIPTION |
|---|---|
| SLA | 7-bit slave address |
| R | Read bit |
| W | Write bit |
| ACK | Acknowledgement (acknowledge bit = logic 0) |
| $\overline{\text{ACK}}$ | No acknowledgement (acknowledge bit = logic 1) |
| DATA | 8-bit data byte to or from $I^2C$-bus |
| MST | Master |
| SLV | Slave |
| TRX | Transmitter |
| REC | Receiver |

### 6.9 Standard serial interface SIO0: UART

This serial port is full duplex which means that it can transmit and receive simultaneously. It is also receive-buffered and can commence reception of a second byte before a previously received byte has been read from the register. (However, if the first byte has not been read by the time the reception of the second byte is complete, one of the bytes will be lost). The serial port receive and transmit registers are both accessed via the Special Function Register S0BUF. Writing to S0BUF loads the transmit register and reading S0BUF accesses a physically separate receive register.

The serial port can operate in 4 modes:

Mode 0 Serial data enters and exits through RXD. TXD outputs the shift clock. 8 bits are transmitted/received (LSB first). The baud rate is fixed at $1/12$ the oscillator frequency.

Mode 1 10 bits are transmitted (through TXD) or received (through RXD): a start bit (0), 8 data bits (LSB first) and a stop bit (1). On receive, the stop bit goes into RB8 in Special Function Register S0CON. The baud rate is variable.

Mode 2 11 bits are transmitted (through TXD) or received (through RXD): a start bit (0), 8 data bits (LSB first), a programmable 9th data bit and a stop bit (1). On transmit, the 9th data bit (TB8 in S0CON) can be assigned the value of 0 or 1. Or, for example, the parity bit (P, in the PSW) could be moved into TB8. On receive, the 9th data bit goes into RB8 in S0CON, while the stop bit is ignored. The baud rate is programmable to either $1/32$ or $1/64$ of the oscillator frequency.

Mode 3 11 bits are transmitted (through TXD) or received (through RXD): a start bit (0), 8 data bits (LSB first), a programmable 9th data bit and a stop bit (1). In fact, Mode 3 is the same as Mode 2 in all respects except baud rate. The baud rate in Mode 3 is variable.

In all four modes, transmission is initiated by any instruction that uses S0BUF as a destination register. Reception is initiated in Mode 0 by the condition RI = 0 and REN = 1. Reception is initiated in the other modes by the incoming start bit if REN = 1.

#### 6.9.1 Multiprocessor communications

Modes 2 and 3 have a special provision for multiprocessor communications. In these modes, 9 data bits are received. The 9th bit goes into RB8. The following bit is the stop bit. The port can be programmed such that when the stop bit is received, the serial port interrupt will be activated, but only if RB8 = 1. This feature is enabled by setting bit SM2 in S0CON. One use of this feature, in multiprocessor systems, is as follows.

When the master processor wants to transmit a block of data to one of several slaves, it first sends out an address byte which identifies the target slave. An address byte differs from a data byte in that the 9th bit is HIGH in an address byte and LOW in a data byte. With SM2 = 1, no slave will be interrupted by a data byte. An address byte, however, will interrupt all slaves, so that each slave can examine the received byte and see if it is being addressed. The addressed slave will clear its SM2 bit and prepare to receive the data bytes that will be sent. The slaves that were not being addressed leave their SM2 bits set and go on about their business, ignoring the coming data bytes.

SM2 has no effect in Mode 0, and in Mode 1 can be used to check the validity of the stop bit. In a Mode 1 reception, if SM2 = 1, the receive interrupt will not be activated unless a valid stop bit is received.

### 6.9.2 Serial Port Control Register (S0CON)

The Serial Port Control and Status Register is the Special Function Register S0CON; shown in Table 22. The register contains not only the mode selection bits, but also the 9th data bit for transmit and receive (TB8 and RB8), and the serial port interrupt bits (TI and RI).

**Table 22** Serial Port Control Register (SFR address 98H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |

**Table 23** Description of S0CON bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| S0CON.7 | SM0 | These two bits are used to select the serial port mode. See Table 24. |
| S0CON.6 | SM1 | |
| S0CON.5 | SM2 | Enables the multiprocessor communication feature in Modes 2 and 3. In these modes, if SM2 = 1, then RI will not be activated if the received 9th data bit (RB8) is a logic 0. In Mode 1, if SM2 = 1, then RI will not be activated unless a valid Stop bit was received. In Mode 0, SM2 should be a logic 0. |
| S0CON.4 | REN | Enables serial reception and is set by software to enable reception, and cleared by software to disable reception. |
| S0CON.3 | TB8 | Is the 9th data bit that will be transmitted in Modes 2 and 3. Set or cleared by software as desired. |
| S0CON.2 | RB8 | In Modes 2 and 3, is the 9th data bit received. In Mode 1, if SM2 = 0 then RB8 is the stop bit that was received. In Mode 0, RB8 is not used. |
| S0CON.1 | TI | The transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit time in the other modes, in any serial transmission. Must be cleared by software. |
| S0CON.0 | RI | The receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial transmission (for exception see SM2). Must be cleared by software. |

**Table 24** Selection of the serial port modes

| SM0 | SM1 | MODE | DESCRIPTION | BAUD RATE |
|---|---|---|---|---|
| 0 | 0 | 0 | shift register | $^1/_{12}f_{osc}$ |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $^1/_{64}f_{osc}$ or $^1/_{32}f_{osc}$ |
| 1 | 1 | 3 | 9-bit UART | variable |

### 6.10 Baud rates

The baud rate in Mode 0 is fixed and may be calculated as shown below:

$$\text{Baud rate} = \frac{f_{osc}}{12}$$

The baud rate in Mode 2 depends on the value of the SMOD bit in Special Function Register PCON. If SMOD = 0, (which is the value on reset), the baud rate is $1/64$ the oscillator frequency. If SMOD = 1, the baud rate is $1/32$ the oscillator frequency. The baud rate in Mode 2 may be calculated as shown below:

$$\text{Baud rate} = \frac{2^{SMOD}}{64} \times f_{osc}$$

The baud rates in Modes 1 and 3 are determined by the Timer 1 or Timer 2 overflow rate.

#### 6.10.1 Using Timer 1 to generate baud rates

When Timer 1 is used as the baud rate generator, the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate and the value of the SMOD bit as follows:

$$\text{Baud rate} = \frac{2^{SMOD}}{32} \times \text{Timer 1 overflow rate}$$

The Timer 1 interrupt should be disabled in this application. The Timer itself can be configured for either 'timer' or 'counter' operation in any of its 3 running modes. In most typical applications, it is configured for 'timer' operation, in the Auto-Reload mode (high nibble of TMOD = 0010B). In this case the baud rate is given by the formula:

$$\text{Baud rate} = \frac{2^{SMOD}}{32} \times f_{osc} \times \frac{1}{[12 \times (256 - TH1)]}$$

By configuring Timer 1 to run as a 16-bit timer (high nibble of TMOD = 0001B), and using the Timer 1 interrupt to do a 16-bit software reload, very low baud rates can be achieved. Table 25 lists various commonly used baud rates and how they can be obtained from Timer 1.

**Table 25** Timer 1 generated commonly used baud rates

| BAUD RATE | $f_{osc}$ (MHz) | SMOD | C/T | TIMER 1 MODE | RELOAD VALUE |
|---|---|---|---|---|---|
| Mode 0 max: 1 Mbits/s | 12 | X | X | X | X |
| Mode 2 max: 375 kbits/s | 12 | 1 | X | X | X |
| Modes 1 and 3: 62.5 kbit/s | 12 | 1 | 0 | 2 | FFH |
| 19.2 kbits/s | 11.059 | 1 | 0 | 2 | FDH |
| 9.6 kbits/s | 11.059 | 0 | 0 | 2 | FDH |
| 4.8 kbits/s | 11.059 | 0 | 0 | 2 | FAH |
| 2.4 kbits/s | 11.059 | 0 | 0 | 2 | F4H |
| 1.2 kbits/s | 11.059 | 0 | 0 | 2 | E8H |
| 137.5 kbits/s | 11.986 | 0 | 0 | 2 | 1DH |
| 110 | 6 | 0 | 0 | 2 | 72H |
| 110 | 12 | 0 | 0 | 1 | FEEBH |

### 6.10.2 Using Timer 2 to generate baud rates

Timer 2 is selected as a baud rate generator by setting the RTCLK bit in T2CON. The baud rate generator mode is similar to the Auto-Reload mode, in that a roll-over in TH2 causes Timer 2 registers to be reloaded with the 16-bit value held in the registers RCAP2H and RCAP2L, which are preset by software. Baud rates in Modes 1 and 3 are determined by Timer 2's overflow rate as specified below.

$$\text{Baud rate} = \frac{\text{Timer 2 overflow rate}}{16}$$

The timer can be configured for either 'timer' or 'counter' operation. In typical applications it is configured for timer operation ($C/\overline{T2} = 0$). Timer operation is slightly different for Timer 2 when it is being used as a baud rate generator. Normally, as a timer it would increment every machine cycle at a frequency of $^1/_{12}f_{osc}$. However, as a baud rate generator it increments every state time at a frequency of $^1/_2f_{osc}$. In this case the baud rate is determined as specified below.

$$\text{Baud rate} = \frac{f_{osc}}{32 \times [65536 - (\text{RCAP2H;RCAP2L})]}$$

Where (RCAP2H;RCAP2L) is the content of registers RCAP2H and RCAP2L taken as a 16-bit unsigned integer.

The baud rate generator mode for Timer 2 is shown in Fig.14. This figure is only valid if RTCLK = 1. At roll-over TH2 does not set the TF2 bit in T2CON and therefore, will not generate an interrupt. Consequently, the Timer 2 interrupt does not need to be disabled when in the baud rate generator mode. If EXEN2 is set, a HIGH-to-LOW transition on T2EX will set the EXF2 bit, also in T2CON, but will not cause a reload from (RCAP2H; RCAP2L) to (TH2; TL2). Therefore, in this mode T2EX may be used as an additional external interrupt.

When Timer 2 is operating as a timer (TR2 = 1), in the baud rate generator mode, registers TH2 and TL2 should not be accessed. Under these conditions the timer is being incremented every state time and therefore the results of a read or write may not be accurate. The RCAP registers however, may be read but not written to. A write might overlap a reload and cause write and/or reload errors. If a write operation is required Timer 2 should first be turned off by clearing the TR2 bit.

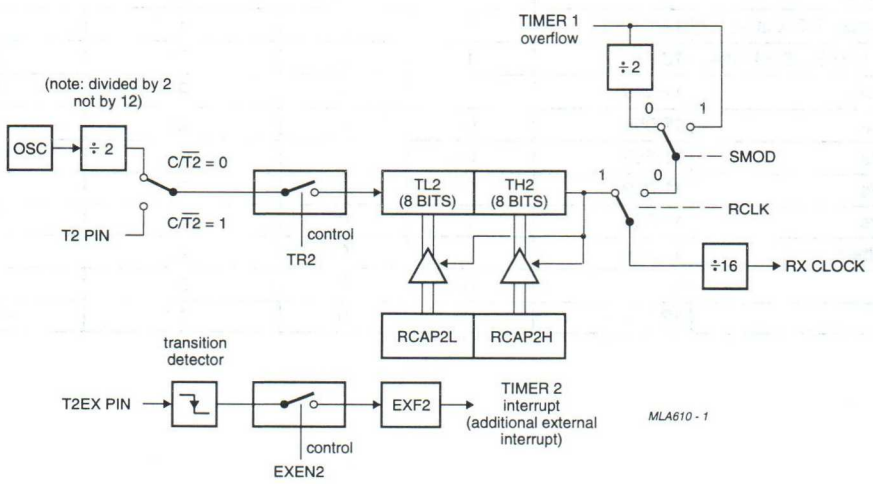



Fig.14 Timer 2 in Baud Rate Generator Mode.

## 6.11 Interrupt system

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronously to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. The interrupt system is shown in Fig.15. The P83CL781 acknowledges interrupt requests from fifteen sources as follows:

- INT0 to INT9
- Timer 0, Timer 1 and Timer 2
- $I^2C$-bus serial I/O
- UART.

Each interrupt vectors to a separate location in program memory for its service routine. Each source can be individually enabled or disabled by its corresponding bit in the Interrupt Enable Registers (1EN0 and 1EN1). The priority level is selected via the Interrupt Priority Registers (IP0 and IP1). All enabled sources can be globally disabled or enabled.

### 6.11.1 External interrupts INT2 to INT9

Port 1 lines serve an alternative purpose as eight additional interrupts INT2 to INT9. When enabled, each of these lines may wake-up the device from the Power-down mode. Using the Interrupt Polarity Register (IX1), each pin may be initialized to be either active HIGH or active LOW. IRQ1 is the Interrupt Request Flag Register. If the interrupt is enabled, each flag will be set on an interrupt request but must be cleared by software, i.e. via the interrupt software or when the interrupt is disabled.

Port 1 interrupts are level sensitive. A Port 1 interrupt will be recognized when a level (HIGH or LOW depending on the Interrupt Polarity Register) on P1.n is held active for at least one machine cycle. The interrupt request is not serviced until the next machine cycle. The external interrupt configuration is shown in Fig.15.

### 6.11.2 Interrupt priority

Each interrupt source can be set to either a high priority or to a low priority. If both priorities are requested simultaneously, the processor will branch to the high priority vector. A low priority interrupt can only be interrupted by a high priority interrupt. A high priority interrupt routine can not be interrupted.

Table 26 shows the interrupt vectors in order of priority. X0 having the highest priority; X9 the lowest. The vector indicates the ROM location where the appropriate interrupt service routine starts.

**Table 26** Interrupt vectors

| SOURCE | SYMBOL | VECTOR |
|---|---|---|
| External 0 | X0 | 0003H |
| $I^2C$-bus port | S1 | 002BH |
| External 5 | X5 | 0053H |
| Timer 0 | T0 | 000BH |
| Timer 2 | T2 | 0033H |
| External 6 | X6 | 005BH |
| External 1 | X1 | 0013H |
| External 2 | X2 | 003BH |
| External 7 | X7 | 0063H |
| Timer 1 | T1 | 001BH |
| External 3 | X3 | 0043H |
| External 8 | X8 | 006BH |
| UART | SO | 0023H |
| External 4 | X4 | 004BH |
| External 9 | X9 | 0073H |

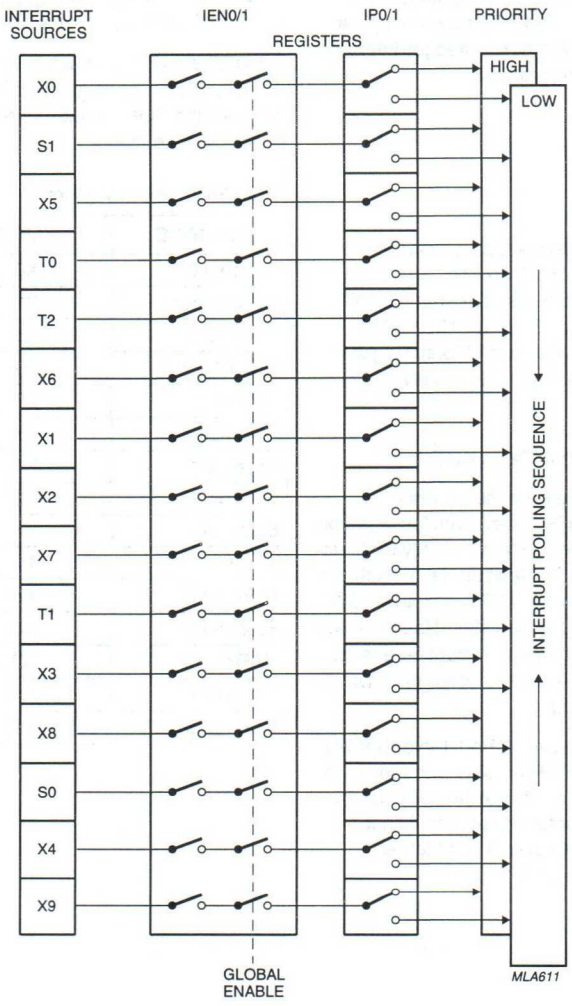



Fig.15 Interrupt system.

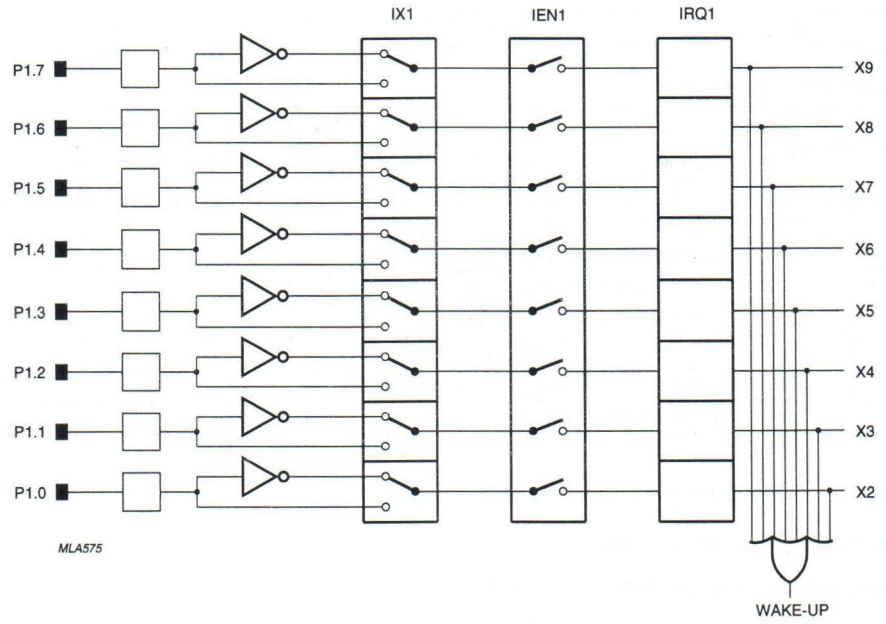



Fig.16 External interrupt configuration.

### 6.11.3 Interrupt Enable Register (IEN0)

**Table 27** Interrupt Enable Register (SFR address A8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| EA | ET2 | ES1 | ES0 | ET1 | EX1 | ET0 | EX0 |

**Table 28** Description of IEN0 bits

| BIT[1] | SYMBOL | FUNCTION |
|---|---|---|
| IEN0.7 | EA | General enable/disable control. If EA = 0, no interrupt is enabled. If EA = 1, any individually enabled interrupt will be accepted. |
| IEN0.6 | ET2 | enable T2 interrupt |
| IEN0.5 | ES1 | enable $I^2C$-bus interrupt |
| IEN0.4 | ES0 | enable UART SIO interrupt |
| IEN0.3 | ET1 | enable Timer 1 interrupt (T1) |
| IEN0.2 | EX1 | enable external interrupt 1 |
| IEN0.1 | ET0 | enable Timer 0 interrupt (T0) |
| IEN0.0 | EX0 | enable external interrupt 0 |

**Note**

1. Where: logic 0 = interrupt disabled; logic 1 = interrupt enabled.

### 6.11.4 Interrupt Enable Register (IEN1)

**Table 29** Interrupt Enable Register (SFR address E8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| EX9 | EX8 | EX7 | EX6 | EX5 | EX4 | EX3 | EX2 |

**Table 30** Description of IEN1 bits

| BIT[1] | SYMBOL | FUNCTION |
|---|---|---|
| IEN1.7 | EX9 | enable external interrupt 9 |
| IEN1.6 | EX8 | enable external interrupt 8 |
| IEN1.5 | EX7 | enable external interrupt 7 |
| IEN1.4 | EX6 | enable external interrupt 6 |
| IEN1.3 | EX5 | enable external interrupt 5 |
| IEN1.2 | EX4 | enable external interrupt 4 |
| IEN1.1 | EX3 | enable external interrupt 3 |
| IEN1.0 | EX2 | enable external interrupt 2 |

**Note**

1. Where: logic 0 = interrupt disabled; logic 1 = interrupt enabled.

### 6.11.5 Interrupt Priority Register (IP0)

**Table 31** Interrupt Priority Register (SFR address B8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | PT2 | PS1 | PS0 | PT1 | PX1 | PT0 | PX0 |

**Table 32** Description of IP0 bits

| BIT[1] | SYMBOL | FUNCTION |
|---|---|---|
| IP0.7 | – | Reserved |
| IP0.6 | PT2 | Timer 2 interrupt priority level |
| IP0.5 | PS1 | $I^2C$-bus interrupt priority level |
| IP0.4 | PS0 | UART SIO interrupt priority level |
| IP0.3 | PT1 | Timer 1 interrupt priority level |
| IP0.2 | PX1 | External interrupt 1 priority level |
| IP0.1 | PT0 | Timer 0 interrupt priority level |
| IP0.0 | PX0 | External interrupt 0 priority level |

**Note**

1. Where: logic 0 = low priority; logic 1 = high priority.

### 6.11.6 Interrupt Priority Register (IP1)

**Table 33** Interrupt Priority Register (SFR address F8H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| PX9 | PX8 | PX7 | PX6 | PX5 | PX4 | PX3 | PX2 |

**Table 34** Description of IP1 bits

| BIT[1] | SYMBOL | FUNCTION |
|---|---|---|
| IP1.7 | PX9 | external interrupt 9 priority level |
| IP1.6 | PX8 | external interrupt 8 priority level |
| IP1.5 | PX7 | external interrupt 7 priority level |
| IP1.4 | PX6 | external interrupt 6 priority level |
| IP1.3 | PX5 | external interrupt 5 priority level |
| IP1.2 | PX4 | external interrupt 4 priority level |
| IP1.1 | PX3 | external interrupt 3 priority level |
| IP1.0 | PX2 | external interrupt 2 priority level |

**Note**

1. Where: logic 0 = low priority; logic 1 = high priority.

### 6.11.7 Interrupt Polarity Register (IX1)

Writing either a logic 1 or logic 0logic 0 to any Interrupt Polarity Register bit sets the polarity level of the corresponding external interrupt to an active HIGH or active LOW respectively.

**Table 35** Interrupt Polarity Register (SFR address E9H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| IL9 | IL8 | IL7 | IL6 | IL5 | IL4 | IL3 | IL2 |

**Table 36** Description of IX1 bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IX1.7 | IL9 | external interrupt 9 polarity level |
| IX1.6 | IL8 | external interrupt 8 polarity level |
| IX1.5 | IL7 | external interrupt 7 polarity level |
| IX1.4 | IL6 | external interrupt 6 polarity level |
| IX1.3 | IL5 | external interrupt 5 polarity level |
| IX1.2 | IL4 | external interrupt 4 polarity level |
| IX1.1 | IL3 | external interrupt 3 polarity level |
| IX1.0 | IL2 | external interrupt 2 polarity level |

### 6.11.8 Interrupt Request Flag Register (IRQ1)

**Table 37** Interrupt Request Flag Register (SFR address C0H)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| IQ9 | IQ8 | IQ7 | IQ6 | IQ5 | IQ4 | IQ3 | IQ2 |

**Table 38** Description of IRQ1 bits

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IRQ1.7 | IQ9 | external interrupt 9 request flag |
| IRQ1.6 | IQ8 | external interrupt 8 request flag |
| IRQ1.5 | IQ7 | external interrupt 7 request flag |
| IRQ1.4 | IQ6 | external interrupt 6 request flag |
| IRQ1.3 | IQ5 | external interrupt 5 request flag |
| IRQ1.2 | IQ4 | external interrupt 4 request flag |
| IRQ1.1 | IQ3 | external interrupt 3 request flag |
| IRQ1.0 | IQ2 | external interrupt 2 request flag |

6.11.9 RELATED REGISTERS

The following registers are used in conjunction with the interrupt system.

**Table 39** Related registers

| REGISTER | FUNCTION | SFR ADDRESS |
|---|---|---|
| IX1 | Interrupt Polarity Register | E9H |
| IRQ1 | Interrupt Request Flag Register | C0H |
| IEN0 | Interrupt Enable Register | A8H |
| IEN1 | Interrupt Enable Register (INT2 to INT9) | E8H |
| IP0 | Interrupt Priority Register | B8H |
| IP1 | Interrupt Priority Register (INT2 to INT9) | F8H |

## 6.12 Oscillator circuitry

The on-chip oscillator circuitry of the P83CL781 is a single-stage inverting amplifier biased by an internal feedback resistor. The oscillator circuit is shown in Fig.17. For operation as a standard quartz oscillator, no external components are needed (except at 32 kHz). When using external capacitors, ceramic resonators, coils and RC networks to drive the oscillator, five different configurations are supported (see Table 40 and Fig.18).

In the Power-down mode the oscillator is stopped and XTAL1 is pulled HIGH. The oscillator invertor is switched off to ensure no current will flow regardless of the voltage at XTAL1. To drive the device with an external clock source, apply the external clock signal to XTAL1, and leave XTAL2 to float, as shown in Fig.18f. There are no requirements on the duty cycle of the external clock, since the input to the internal clocking circuitry is buffered by a flip-flop.

Various oscillator options are provided for optimum on-chip oscillator performance; these are specified in Table 40 and shown in Fig.18. The required option should be stated when ordering.

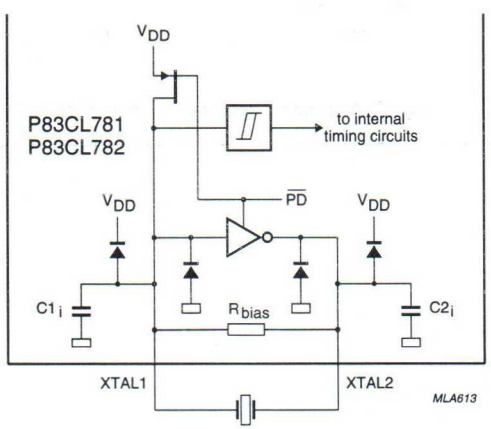



Fig.17 Oscillator.

**Table 40** Oscillator options

| OSCILLATOR | APPLICATION |
|---|---|
| Oscillator 1 | For 32 kHz clock applications with external trimmer for frequency adjustment. A 4.7 MΩ bias resistor is needed for use in parallel with the crystal. See Fig.18c. |
| Oscillator 2 | For low-power, low-frequency operations using LC components. See Fig.18e. |
| Oscillator 3 | For medium frequency range applications. |
| Oscillator 4 | For high frequency range applications. |
| RC | RC oscillator configuration. See Fig.18g. |

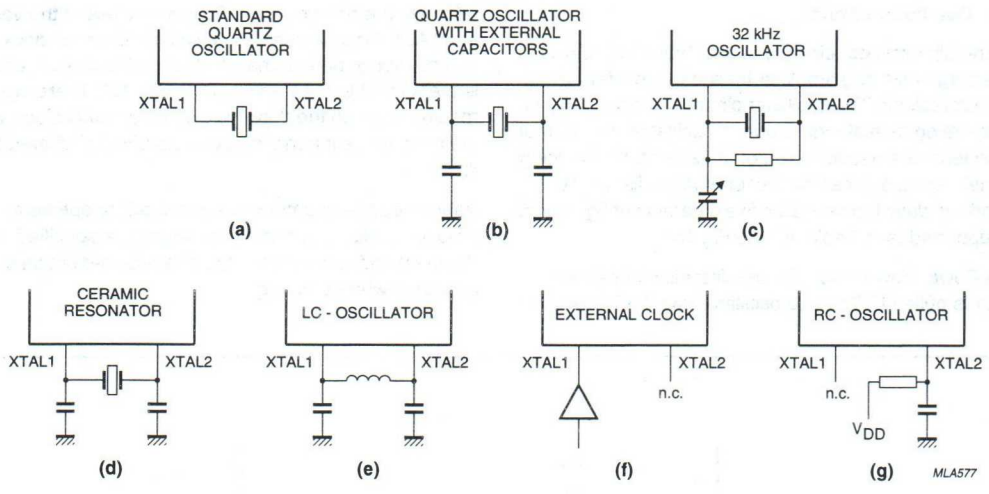



Fig.18 Alternative oscillator configurations.

### 6.12.1 OSCILLATOR TYPE SELECTION GUIDE

**Table 41** Oscillator type selection guide

| RESONATOR | FREQUENCY (MHz) | OPTION | C1 ext. (pF) | | C2 ext. (pF) | | RESONATOR MAX. SERIES RESISTANCE |
|---|---|---|---|---|---|---|---|
| | | | MIN. | MAX. | MIN. | MAX. | |
| Quartz | 0.032 | Osc. 1 | 0 | 0 | 5 | 15 | 15 kΩ; note 1 |
| | 1.0 | Osc. 2 | 0 | 30 | 0 | 30 | 600 Ω |
| | 3.58 | Osc. 2 | 0 | 15 | 0 | 15 | 100 Ω |
| | 4.0 | Osc. 2 | 0 | 20 | 0 | 20 | 75 Ω |
| | 6.0 | Osc. 3 | 0 | 10 | 0 | 10 | 60 Ω |
| | 10.0 | Osc. 4 | 0 | 15 | 0 | 15 | 60 Ω |
| | 12.0 | Osc. 4 | 0 | 10 | 0 | 10 | 40 Ω |
| | 16.0 | Osc. 4 | 0 | 15 | 0 | 15 | 20 Ω |
| PXE | 0.455 | Osc. 2 | 40 | 50 | 40 | 50 | 10 Ω |
| | 1.0 | Osc. 2 | 15 | 50 | 15 | 50 | 100 Ω |
| | 3.58 | Osc. 2 | 0 | 40 | 0 | 40 | 10 Ω |
| | 4.0 | Osc. 2 | 0 | 40 | 0 | 40 | 10 Ω |
| | 6.0 | Osc. 2 | 0 | 20 | 0 | 20 | 5 Ω |
| | 10.0 | Osc. 3 | 0 | 15 | 0 | 15 | 6 Ω |
| | 12.0 | Osc. 4 | 10 | 40 | 10 | 40 | 6 Ω |
| LC | | Osc. 2 | 20 | 90 | 20 | 90 | 10 µH = 1 Ω;<br>100 µH = 5 Ω;<br>1 mH = 75 Ω |

**Note**

1. 32 kHz quartz crystals with a series resistance higher than 15 kΩ will reduce the guaranteed supply voltage range to 2.5 to 3.5 V.

**Table 42** Oscillator equivalent circuit parameters (see note 1)

| SYMBOL | PARAMETER | OPTION | CONDITIONS | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|---|---|
| $g_m$ | transconductance | Osc. 1 | $T_{amb}$ = +25 °C; $V_{DD}$ = 4.5 V | – | 15 | – | µS |
| $g_{m1}$ | | Osc. 2 | | 200 | 600 | 1000 | µS |
| $g_{m2}$ | | Osc. 3 | | 400 | 1500 | 4000 | µS |
| $g_{m3}$ | | Osc. 4 | | 1000 | 4000 | 10000 | µS |
| $C1_i$ | input capacitance | Osc. 1 | | – | 3.0 | – | pF |
| $C1_{i1}$ | | Osc. 2 | | – | 8.0 | – | pF |
| $C1_{i2}$ | | Osc. 3 | | – | 8.0 | – | pF |
| $C1_{i3}$ | | Osc. 4 | | – | 8.0 | – | pF |
| $C2_i$ | output capacitance | Osc. 1 | | – | 23 | – | pF |
| $C2_{i1}$ | | Osc. 2 | | – | 8.0 | – | pF |
| $C2_{i2}$ | | Osc. 3 | | – | 8.0 | – | pF |
| $C2_{i3}$ | | Osc. 4 | | – | 8.0 | – | pF |
| R2 | output resistance | Osc. 1 | | – | 3800 | – | kΩ |
| $R2_1$ | | Osc. 2 | | – | 65 | – | kΩ |
| $R2_2$ | | Osc. 3 | | – | 18 | – | kΩ |
| $R2_3$ | | Osc. 4 | | – | 5.0 | – | kΩ |

**Note**

1. The equivalent circuit data of the internal oscillator compares with that of matched crystals.

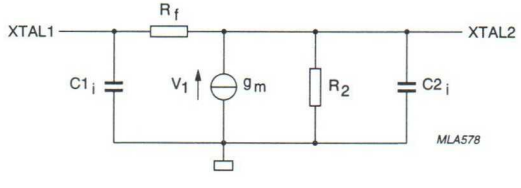



Fig.19 Equivalent circuit diagram.

### 6.12.2 RC OSCILLATOR

The externally adjustable RC oscillator has a frequency range from 100 to 500 kHz.

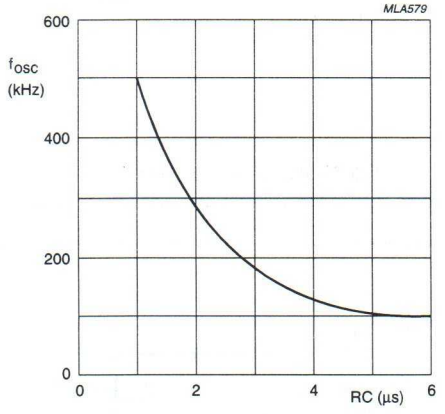



Fig.20 Oscillator frequency as a function of RC.

## 6.13 Reset

To initialize the P83CL781 a reset is performed by either of two methods:

- Applying an external signal to the RST pin
- Via Power-on reset circuitry.

The reset state of the port pins is mask-programmable and can be defined by the user. The standard reset value for Ports 0 to 3 is FFH. A reset leaves the internal registers as shown in Table 43.

### 6.13.1 EXTERNAL RESET USING THE RST PIN

The reset input for the P83CL781 is RST; pin 15. A Schmitt-trigger is used at the input for noise rejection. The output of the Schmitt-trigger is sampled by the reset circuitry every machine cycle. A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods) while the oscillator is running. The CPU responds by executing an internal reset. Port pins adopt their reset state immediately after the RST goes HIGH. During reset, ALE and $\overline{PSEN}$ are held HIGH.

The external reset is asynchronous to the internal clock. The RST pin is sampled during state 5, phase 2 of every machine cycle. After a HIGH is detected at the RST pin, an internal reset is repeated until RST goes LOW. The internal RAM is not affected by reset. When $V_{DD}$ is turned on, the RAM contents are indeterminate.

### 6.13.2 POWER-ON RESET

The device contains on-chip circuitry which switches the port pins to the customer defined logic level as soon as $V_{DD}$ exceeds 1.3 V; if the mask option 'ON' has been chosen. As soon as the minimum supply voltage is reached, the oscillator will start up. However, to ensure that the oscillator is stable before the controller starts, the clock signals are gated away from the CPU for a further 1536 oscillator periods. A hysteresis of approximately 50 mV at a typical power-on switching level of 1.3 V will ensure correct operation. See Fig.23.

The on-chip Power-on reset circuitry can also be switched off via the mask option 'OFF'. This option reduces the Power-down current to typically 800 nA and can be chosen if external reset circuitry is used.

An automatic reset can be obtained by connecting the RST pin to $V_{DD}$ via a 10 µF capacitor. At power-on, the voltage on the RST pin is equal to $V_{DD}$ minus the capacitor voltage, and decreases from $V_{DD}$ as the capacitor charges through the internal resistor ($R_{RST}$) to ground. The larger the capacitor, the more slowly $V_{RST}$ decreases. $V_{RST}$ must remain above the lower threshold of the Schmitt-trigger long enough to effect a complete reset. The time required is the oscillator start-up time, plus 2 machine cycles. The Power-on reset circuitry is shown in Fig.22.

**Table 43** State of internal registers after a reset

| REGISTER | CONTENTS[1] |
|---|---|
| ACC | 0000 0000 |
| B | 0000 0000 |
| DPL | 0000 0000 |
| DPH | 0000 0000 |
| IEN0 | 0000 0000 |
| IEN1 | 0000 0000 |
| IP0 | XX00 0000 |
| IP1 | 0000 0000 |
| IX1 | 0000 0000 |
| IRQ1 | 0000 0000 |
| PCH | 0000 0000 |
| PCL | 0000 0000 |
| PCON | 0XX0 0000 |
| P0 to P3 | 1111 1111 |
| S0BUF | XXXX XXXX |
| S0CON | 0000 0000 |
| S1ADR | 0000 0000 |
| S1CON | 0000 0000 |
| S1DAT | 0000 0000 |
| S1STA | 1111 1000 |
| SP | 0000 0111 |
| TCON | 0000 0000 |
| T2CON | 0000 0000 |
| T3 | 0000 0000 |
| TH0, TH1, TH2 | 0000 0000 |
| TL0, TL1, TL2 | 0000 0000 |
| TMOD | 0000 0000 |
| PSW | 0000 0000 |
| RCAP2L | 0000 0000 |
| RCAP2H | 0000 0000 |

**Note**

1. Where: X = undefined state.

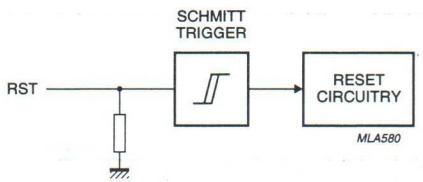



Fig.21 Reset configuration.

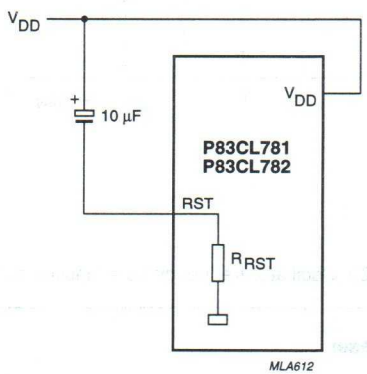



Fig.22 Power-on reset circuitry.

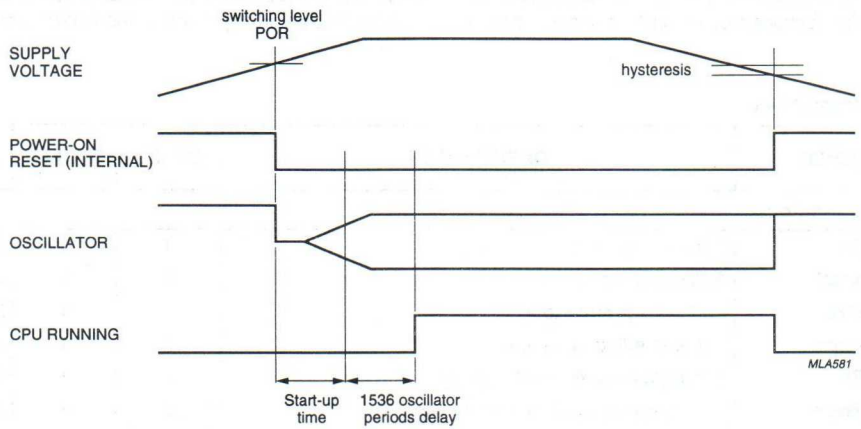



Fig.23 Power-on reset switching level.

## 7 INSTRUCTION SET

The P83CL781 uses a powerful instruction set which permits the expansion of on-chip CPU peripherals and optimizes byte efficiency and execution speed. Assigned opcodes add new high-power operation and permit new addressing modes. The instruction set consists of 49 single-byte, 46 two-byte and 16 three-byte instructions. When using a 12 MHz oscillator, 64 instructions execute in 1 μs and 45 instructions execute in 2 μs. Multiply and divide instructions execute in 4 μs.

**Table 44** Instruction Set

| MNEMONIC | | DESCRIPTION | BYTES | CYCLES | OPCODE (HEX) |
|---|---|---|---|---|---|
| **Arithmetic operations** | | | | | |
| ADD | A,Rr | Add register to A | 1 | 1 | 2* |
| ADD | A,direct | Add direct byte to A | 2 | 1 | 25 |
| ADD | A,@Ri | Add indirect RAM to A | 1 | 1 | 26, 27 |
| ADD | A,#data | Add immediate data to A | 2 | 1 | 24 |
| ADDC | A,Rr | Add register to A with carry flag | 1 | 1 | 3* |
| ADDC | A,direct | Add direct byte to A with carry flag | 2 | 1 | 35 |
| ADDC | A,@Ri | Add indirect RAM to A with carry flag | 1 | 1 | 36, 37 |
| ADDC | A,#data | Add immediate data to A with carry flag | 2 | 1 | 34 |
| SUBB | A,Rr | Subtract register from A with borrow | 1 | 1 | 9* |
| SUBB | A,direct | Subtract direct byte from A with borrow | 2 | 1 | 95 |
| SUBB | A,@Ri | Subtract indirect RAM from A with borrow | 1 | 1 | 96, 97 |
| SUBB | A,#data | Subtract immediate data from A with borrow | 2 | 1 | 94 |
| INC | A | Increment A | 1 | 1 | 04 |
| INC | Rr | Increment register | 1 | 1 | 0* |
| INC | direct | Increment direct byte | 2 | 1 | 05 |
| INC | @Ri | Increment indirect RAM | 1 | 1 | 06, 07 |
| DEC | A | Decrement A | 1 | 1 | 14 |
| DEC | Rr | Decrement register | 1 | 1 | 1* |
| DEC | direct | Decrement direct byte | 2 | 1 | 15 |
| DEC | @Ri | Decrement indirect RAM | 1 | 1 | 16, 17 |
| INC | DPTR | Increment data pointer | 1 | 2 | A3 |
| MUL | AB | Multiply A & B | 1 | 4 | A4 |
| DIV | AB | Divide A by B | 1 | 4 | 84 |
| DA | A | Decimal adjust A | 1 | 1 | D4 |

| MNEMONIC | | DESCRIPTION | BYTES | CYCLES | OPCODE (HEX) |
|---|---|---|---|---|---|
| **Logic operations** | | | | | |
| ANL | A,Rr | AND register to A | 1 | 1 | 5* |
| ANL | A,direct | AND direct byte to A | 2 | 1 | 55 |
| ANL | A,@Ri | AND indirect RAM to A | 1 | 1 | 56, 57 |
| ANL | A,#data | AND immediate data to A | 2 | 1 | 54 |
| ANL | direct,A | AND A to direct byte | 2 | 1 | 52 |
| ANL | direct,#data | AND immediate data to direct byte | 3 | 2 | 53 |
| ORL | A,Rr | OR register to A | 1 | 1 | 4* |
| ORL | A,direct | OR direct byte to A | 2 | 1 | 45 |
| ORL | A,@Ri | OR indirect RAM to A | 1 | 1 | 46, 47 |
| ORL | A,#data | OR immediate data to A | 2 | 1 | 44 |
| ORL | direct,A | OR A to direct byte | 2 | 1 | 42 |
| ORL | direct,#data | OR immediate data to direct byte | 3 | 2 | 43 |
| XRL | A,Rr | Exclusive-OR register to A | 1 | 1 | 6* |
| XRL | A,direct | Exclusive-OR direct byte to A | 2 | 1 | 65 |
| XRL | A,@Ri | Exclusive-OR indirect RAM to A | 1 | 1 | 66, 67 |
| XRL | A,#data | Exclusive-OR immediate data to A | 2 | 1 | 64 |
| XRL | direct,A | Exclusive-OR A to direct byte | 2 | 1 | 62 |
| XRL | direct,#data | Exclusive-OR immediate data to direct byte | 3 | 2 | 63 |
| CLR | A | Clear A | 1 | 1 | E4 |
| CPL | A | Complement A | 1 | 1 | F4 |
| RL | A | Rotate A left | 1 | 1 | 23 |
| RLC | A | Rotate A left through the carry flag | 1 | 1 | 33 |
| RR | A | Rotate A right | 1 | 1 | 03 |
| RRC | A | Rotate A right through the carry flag | 1 | 1 | 13 |
| SWAP | A | Swap nibbles within A | 1 | 1 | C4 |

| MNEMONIC | | DESCRIPTION | BYTES | CYCLES | OPCODE (HEX) |
|---|---|---|---|---|---|
| **Data transfer** | | | | | |
| MOV | A,Rr | Move register to A | 1 | 1 | E* |
| MOV | A,direct** | Move direct byte to A | 2 | 1 | E5 |
| MOV | A,@Ri | Move indirect RAM to A | 1 | 1 | E6, E7 |
| MOV | A,#data | Move immediate data to A | 2 | 1 | 74 |
| MOV | Rr,A | Move A to register | 1 | 1 | F* |
| MOV | Rr,direct | Move direct byte to register | 2 | 2 | A* |
| MOV | Rr,#data | Move immediate data to register | 2 | 1 | 7* |
| MOV | direct,A | Move A to direct byte | 2 | 1 | F5 |
| MOV | direct,Rr | Move register to direct byte | 2 | 2 | 8* |
| MOV | direct,direct | Move direct byte to direct byte | 3 | 2 | 85 |
| MOV | direct,@Ri | Move indirect RAM to direct byte | 2 | 2 | 86, 87 |
| MOV | direct,#data | Move immediate data to direct byte | 3 | 2 | 75 |
| MOV | @RI,A | Move A to indirect RAM | 1 | 1 | F6, F7 |
| MOV | @Ri,direct | Move direct byte to indirect RAM | 2 | 2 | A6, A7 |
| MOV | @Ri,#data | Move immediate data to indirect RAM | 3 | 1 | 76, 77 |
| MOV | DPTR,#data 16 | Load data pointer with a 16-bit constant | 3 | 2 | 90 |
| MOVC | A,@A+DPTR | Move code byte relative to DPTR to A | 1 | 2 | 93 |
| MOVC | A,@A+PC | Move code byte relative to PC to A | 1 | 2 | 83 |
| MOVX | A,@Ri | Move external RAM (8-bit address) to A | 1 | 2 | E2, E3 |
| MOVX | A,@DPTR | Move external RAM (16-bit address) to A | 1 | 2 | E0 |
| MOVX | @Ri,A | Move A to external RAM (8-bit address) | 1 | 2 | F2, F3 |
| MOVX | @DPTR,A | Move A to external RAM (16-bit address) | 1 | 2 | F0 |
| PUSH | direct | Push direct byte onto stack | 2 | 2 | C0 |
| POP | direct | Pop direct byte from stack | 2 | 2 | D0 |
| XCH | A,Rr | Exchange register with A | 1 | 1 | C* |
| XCH | A,direct | Exchange direct byte with A | 2 | 1 | C5 |
| XCH | A,@Ri | Exchange indirect RAM with A | 1 | 1 | C6, C7 |
| XCHD | A,@Ri | Exchange LOW-order nibble indirect RAM with A | 1 | 1 | D6, D7 |

| MNEMONIC | | DESCRIPTION | BYTES | CYCLES | OPCODE (HEX) |
|---|---|---|---|---|---|
| **Boolean variable manipulation** | | | | | |
| CLR | C | Clear carry flag | 1 | 1 | C3 |
| CLR | bit | Clear direct bit | 2 | 1 | C2 |
| SETB | C | Set carry flag | 1 | 1 | D3 |
| SETB | bit | Set direct bit | 2 | 1 | D2 |
| CPL | C | Complement carry flag | 1 | 1 | B3 |
| CPL | bit | Complement direct bit | 2 | 1 | B2 |
| ANL | C,bit | AND direct bit to carry flag | 2 | 2 | 82 |
| ANL | C,/bit | AND complement of direct bit to carry flag | 2 | 2 | B0 |
| ORL | C,bit | OR direct bit to carry flag | 2 | 2 | 72 |
| ORL | C,/bit | OR complement of direct bit to carry flag | 2 | 2 | A0 |
| MOV | C,bit | Move direct bit to carry flag | 2 | 1 | A2 |
| MOV | bit,C | Move carry flag to direct bit | 2 | 2 | 92 |
| **Program and machine control** | | | | | |
| ACALL | addr11 | Absolute subroutine call | 2 | 2 | •1 addr |
| LCALL | addr16 | Long subroutine call | 3 | 2 | 12 |
| RET | | Return from subroutine | 1 | 2 | 22 |
| RETI | | Return from interrupt | 1 | 2 | 32 |
| AJMP | addr11 | Absolute jump | 2 | 2 | ♦1 addr |
| LJMP | addr16 | Long jump | 3 | 2 | 02 |
| SJMP | rel | Short jump (relative address) | 2 | 2 | 80 |
| JMP | @A+DPTR | Jump indirect relative to the DPTR | 1 | 2 | 73 |
| JZ | rel | Jump if A is zero | 2 | 2 | 60 |
| JNZ | rel | Jump if A is not zero | 2 | 2 | 70 |
| JC | rel | Jump if carry flag is set | 2 | 2 | 40 |
| JNC | rel | Jump if carry flag is not set | 2 | 2 | 50 |
| JB | bit,rel | Jump if direct bit is set | 3 | 2 | 20 |
| JNB | bit,rel | Jump if direct bit is not set | 3 | 2 | 30 |
| JBC | bit,rel | Jump if direct bit is set and clear bit | 3 | 2 | 10 |
| CJNE | A,direct,rel | Compare direct to A and jump if not equal | 3 | 2 | B5 |
| CJNE | A,#data,rel | Compare immediate to A and jump if not equal | 3 | 2 | B4 |
| CJNE | Rr,#data,rel | Compare immediate to register and jump if not equal | 3 | 2 | B* |
| CJNE | @Ri,#data,rel | Compare immediate to indirect and jump if not equal | 3 | 2 | B6, B7 |
| DJNZ | Rr,rel | Decrement register and jump if not zero | 2 | 2 | D* |
| DJNZ | direct,rel | Decrement direct and jump if not zero | 3 | 2 | D5 |
| NOP | | No operation | 1 | 1 | 00 |

**Table 45** Notation for data addressing modes

| SYMBOL | DESCRIPTION |
|---|---|
| Rr | Working registers R0 to R7. |
| direct | 128 internal RAM locations and any special function register (SFR). |
| @Ri | Indirect internal RAM location addressed by register R0 or R1. |
| #data | 8-bit constant included in instruction. |
| #data 16 | 16-bit constant included as bytes 2 and 3 of instruction. |
| bit | Direct addressed bit in internal RAM or SFR. |
| addr16 | 16-bit destination address. Used by LCALL and LJMP. The branch will be anywhere within the 64 kbyte program memory address space. |
| addr11 | 11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 kbyte page of program memory as the first byte of the following instruction. |
| rel | Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is −128 to +127 bytes relative to first byte of the following instruction. |

**Table 46** Hexadecimal opcode cross-reference

| SYMBOL | DESCRIPTION |
|---|---|
| * | 8, 9, A, B, C, D, E, F. |
| • | 11, 31, 51, 71, 91, B1, D1, F1. |
| ♦ | 01, 21, 41, 61, 81, A1, C1, E1. |

first hexadecimal character of opcode

second hexadecimal character of opcode

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | NOP | AJMP addr 11 | LJMP addr 16 | RR A | INC A | INC dir | INC @ Ri | | INC Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | JBC bit, rel | ACALL addr 11 | LCALL addr 16 | RRC A | DEC A | DEC dir | DEC @ Ri | | DEC Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 2 | JB bit, rel | AJMP addr 11 | RET | RL A | ADD A, # data | ADD A, dir | ADD A, @ Ri | | ADD A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 3 | JNB bit, rel | ACALL addr 11 | RETI | RLC A | ADDC A, # data | ADDC A, dir | ADDC A, @ Ri | | ADDC A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 4 | JC rel | AJMP addr 11 | ORL dir, A | ORL dir, # data | ORL A, # data | ORL A, dir | ORL A, @ Ri | | ORL A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 5 | JNC rel | ACALL addr 11 | ANL dir, A | ANL dir, # data | ANL A, # data | ANL A, dir | ANL A, @ Ri | | ANL A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 6 | JZ rel | AJMP addr 11 | XRL dir, A | XRL dir, # data | XRL A, # data | XRL A, dir | XRL A, @ Ri | | XRL A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 7 | JNZ rel | ACALL addr 11 | ORL C, bit | JMP @ A+DPTR | MOV A, # data | MOV dir, # data | MOV @ Ri, # data | | MOV Rr, # data | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 8 | SJMP rel | AJMP addr 11 | ANL C, bit | MOVC A, @ A+PC | DIV AB | MOV dir, dir | MOV dir, @ Ri | | MOV dir, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 9 | MOV DPTR, # data 16 | ACALL addr 11 | MOV bit, C | MOVC A, @ A+DPTR | SUBB A, # data | SUBB A, dir | SUBB A, @ Ri | | SUBB A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| A | ORL C, / bit | AJMP addr 11 | MOV bit, C | INC DPTR | MUL AB | | MOV @ Ri, dir | | MOV Rr, dir | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| B | ANL C, / bit | ACALL addr 11 | CPL bit | CPL C | CJNE A, # data, rel | CJNE A, dir, rel | CJNE @ Ri, # data, rel | | CJNE Rr, # data, rel | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| C | PUSH dir | AJMP addr 11 | CLR bit | CLR C | SWAP A | XCH A, dir | XCH A, @ Ri | | XCH A, Rr | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| D | POP dir | ACALL addr 11 | SETB bit | SETB C | DA A | DJNZ dir, rel | XCHD A, @ Ri | | DJNZ Rr, rel | | | | | | | |
| | | | | | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| E | MOVX A, @ DPTR | AJMP addr 11 | MOVX A, @ Ri | | CLR A | MOV * A, dir | MOV A, @ Ri | | MOV A, Rr | | | | | | | |
| | | | 0 | 1 | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| F | MOVX @ DPTR, A | ACALL addr 11 | MOVX @ Ri, A | | CPL A | MOV dir, A | MOV @ Ri, A | | MOV Rr, A | | | | | | | |
| | | | 0 | 1 | | | 0 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |

MLA632 - 1

* MOV A, ACC is not a valid instruction.

## 8 LIMITING VALUES

In accordance with the Absolute Maximum Rating System (IEC 134)

| SYMBOL | PARAMETER | MIN. | MAX. | UNIT |
|---|---|---|---|---|
| $V_{DD}$ | supply voltage | –0.5 | +6.5 | V |
| $V_I$ | input voltage on any pin with respect to ground ($V_{SS}$) | –0.5 | $V_{DD}$ + 0.5 | V |
| $I_I$ | DC current on any input | – | +5.0 | mA |
| $I_O$ | DC current on any output | – | –5.0 | mA |
| $P_{tot}$ | total power dissipation | – | 300 | mW |
| $T_{stg}$ | storage temperature | –65 | +150 | °C |
| $T_{amb}$ | operating ambient temperature - P83CL781 | –40 | +85 | °C |
| | operating ambient temperature - P83CL782 | –25 | +55 | °C |
| $T_j$ | operating junction temperature | – | +125 | °C |

## 9 DC CHARACTERISTICS

The DC characteristics apply to both the P83CL781 and the P83CL782 unless otherwise stated. $V_{DD}$ = 1.8 to 6 V; $V_{SS}$ = 0 V; $T_{amb}$ = –40 to +85 °C for the P83CL781 and –25 to +55 °C for the P83CL782; all voltages with respect to $V_{SS}$ unless otherwise specified. See notes 1, 2 and 3.

| SYMBOL | PARAMETER | CONDITIONS | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|---|
| **Supply** | | | | | | |
| $V_{DD}$ | supply voltage | | 1.8 | – | 6.0 | V |
| $V_{DD}$ | RAM retention voltage in Power-down mode | | 1.0 | – | 6.0 | V |
| $I_{DD}$ | supply current operating; P83CL781 | $V_{DD}$ = 5 V; $f_{CLK}$ = 12 MHz; note 4 | – | 17 | 25 | mA |
| | | $V_{DD}$ = 3 V; $f_{CLK}$ = 3.58 MHz; note 4 | – | 2.4 | 5 | mA |
| $I_{DD}$ | supply current operating; P83CL782 | $V_{DD}$ = 3.1 V; $f_{CLK}$ = 12 MHz; note 4 | – | 8.4 | 12 | mA |
| | | $V_{DD}$ = 3 V; $f_{CLK}$ = 3.58 MHz; note 4 | – | 2.4 | 5 | µA |
| $I_{IDD(ID)}$ | supply current Idle mode; P83CL781 | $V_{DD}$ = 5 V; $f_{CLK}$ = 12 MHz; note 5 | – | 5.1 | 12 | mA |
| | | $V_{DD}$ = 3 V; $f_{CLK}$ = 3.58 MHz; note 5 | – | 0.75 | 3 | mA |
| $I_{IDD(ID)}$ | supply current Idle mode; P83CL782 | $V_{DD}$ = 5 V; $f_{CLK}$ = 12 MHz; note 5 | – | 2.7 | 5 | mA |
| | | $V_{DD}$ = 3 V; $f_{CLK}$ = 3.58 MHz; note 5 | – | 0.75 | 3 | mA |
| $I_{IDD(PD)}$ | supply current Power-down mode | $V_{DD}$ = 1.8 V; $T_{amb}$ = 25 °C; note 6 | – | – | 10 | µA |

| SYMBOL | PARAMETER | CONDITIONS | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|---|
| **Inputs** | | | | | | |
| $V_{IL}$ | LOW level input voltage | note 7 | $V_{SS}$ | – | $0.3V_{DD}$ | V |
| $V_{IH}$ | HIGH level input voltage | note 7 | $0.7V_{DD}$ | – | $V_{DD}$ | V |
| $I_{IL}$ | LOW level input current | $V_{DD}$ = 5 V; $V_{IN}$ = 0.4 V; note 7 | – | – | –100 | µA |
| | | $V_{DD}$ = 2.5 V; $V_{IN}$ = 0.4 V; note 7 | – | – | –50 | µA |
| $I_{IL(T)}$ | LOW level input current (HIGH-to-LOW transition) | $V_{DD}$ = 5 V; $V_{IN}$ = $0.5V_{DD}$; note 7 | – | – | –1.0 | mA |
| | | $V_{DD}$ = 2.5 V; $V_{IN}$ = $0.5V_{DD}$; note 7 | – | – | –500 | µA |
| $I_{LI}$ | input leakage current | $V_{SS} < V_I < V_{DD}$; note 7 | – | – | ±10 | µA |
| **Outputs** | | | | | | |
| $I_{OL}$ | LOW level output current; except SDA and SCL | $V_{DD}$ = 5 V; $V_{OL}$ = 0.4 V | 1.6 | – | – | mA |
| | | $V_{DD}$ = 2.5 V; $V_{OL}$ = 0.4 V | 0.7 | – | – | mA |
| $I_{OL1}$ | LOW level output current; SDA and SCL | $V_{DD}$ = 5 V; $V_{OL}$ = 0.4 V | 3.0 | – | – | mA |
| $I_{OH}$ | HIGH level output current (push-pull options only) | $V_{DD}$ = 5 V; $V_{OH} = V_{DD} - 0.4$ V | –1.6 | – | – | mA |
| | | $V_{DD}$ = 2.5 V; $V_{OH} = V_{DD} - 0.4$ V | –0.7 | – | – | mA |
| $R_{RST}$ | RST pull-down resistor | | 10 | – | 200 | kΩ |

**Notes**

1. Capacitive loading on Ports 0 and 2 may cause spurious noise pulses to be superimposed on the LOW level output voltage of ALE, Port 1 and Port 3 pins when these make a HIGH-to-LOW transition during bus operations. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make HIGH-to-LOW transitions during bus operations. In the most adverse conditions (capacitive loading >100 pF), the noise pulse on the ALE line may exceed 0.8 V. In such events it may be required to qualify ALE with a Schmitt-trigger, or use an address latch with a Schmitt-trigger strobe input.
2. Capacitive loading on Ports 0 and 2 may cause the HIGH level output voltage on ALE and $\overline{PSEN}$ to momentarily fall below the 0.9% of $V_{DD}$ specification when the address bits are stabilizing.
3. Circuits with Power-on reset option 'OFF' are tested at $V_{DDmin}$ = 1.8 V; with the 'ON' option (typically 1.3 V) they are tested at $V_{DDmin}$ = 2.3 V.
4. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ =10 ns; $V_{IL} = V_{SS}$; $V_{IH} = V_{DD}$; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = $V_{DD}$.
5. The Idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$; $V_{IH} = V_{DD}$; XTAL2 not connected; $\overline{EA}$ = Port 0 = $V_{DD}$.
6. The Power-down current is measured with all output pins disconnected; XTAL1 not connected; $\overline{EA}$ = Port 0 = $V_{DD}$; RST = $V_{SS}$.
7. The input threshold voltage of P1.6/SCL and P1.7/SDA meet the $I^2C$-bus specification. Therefore, an input voltage below $0.3V_{DD}$ will be recognized as a logic 0 and an input voltage above $0.7V_{DD}$ will be recognized as a logic 1.

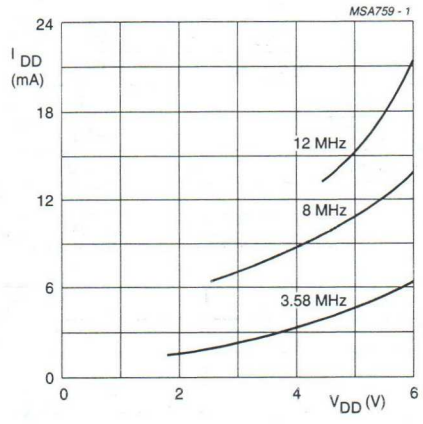



Fig.24 P83CL781: typical operating current as a function of frequency and $V_{DD}$, $T_{amb}$ = 25 °C, oscillator option Osc.3.

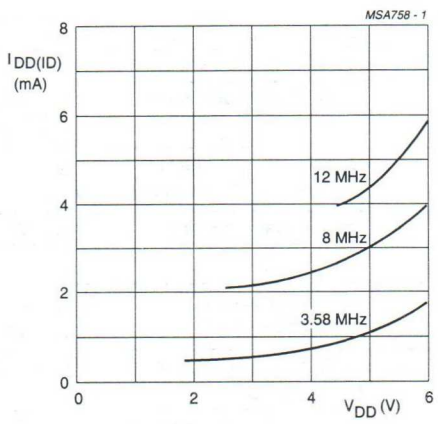



Fig.25 P83CL781: typical Idle current as a function of frequency and $V_{DD}$, $T_{amb}$ = 25 °C, oscillator option Osc.3.

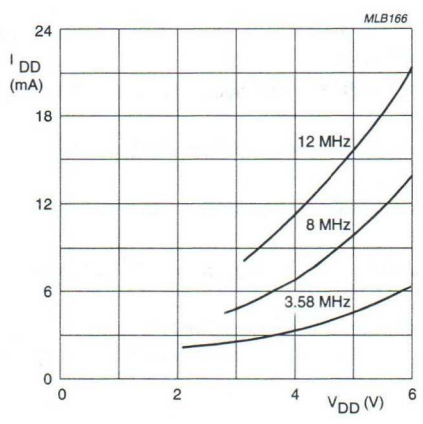



Fig.26 P83CL782: typical operating current as a function of frequency and $V_{DD}$, $T_{amb}$ = 25 °C, oscillator option Osc.3.

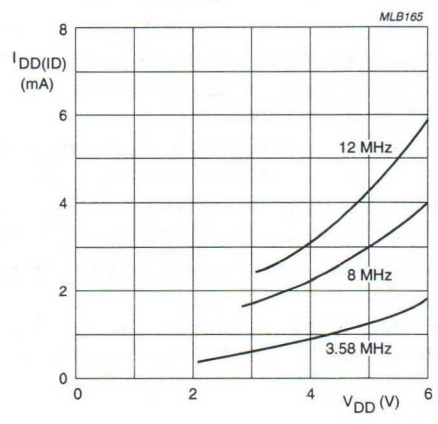



Fig.27 P83CL782: typical Idle current as a function of frequency and $V_{DD}$, $T_{amb}$ = 25 °C, oscillator option Osc.3.

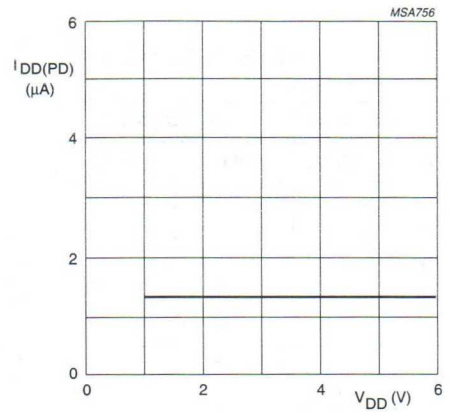



Fig.28 Typical Power-down current as a function of $V_{DD}$, $T_{amb}$ = 25 °C.

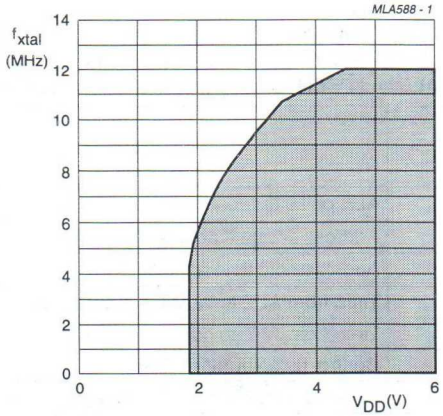



Fig.29 Frequency operating range - P83CL781.

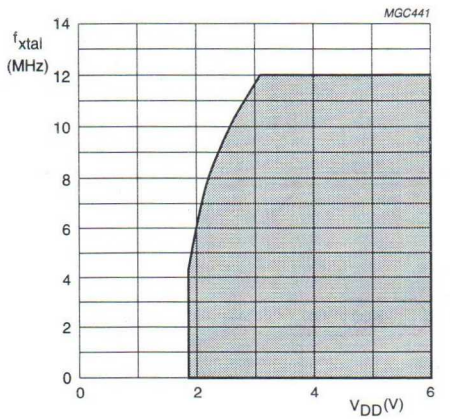



Fig.30 Frequency operating range - P83CL782.

## 10 AC CHARACTERISTICS

The following AC characteristics apply to both the P83CL781 and P83CL782 unless otherwise stated.

### 10.1 Program memory

$V_{DD}$ = 5 V; $V_{SS}$ = 0 V; $T_{amb}$ = −40 to +85 °C for the P83CL781 and −25 to +55 °C for the P83CL782; $C_L$ = 50 pF for Port 0, ALE and $\overline{PSEN}$; $C_L$ = 80 pF for all other outputs unless specified. See Fig.31.

| SYMBOL | PARAMETER | $f_{osc}$ = 12 MHz | | $f_{osc}$ = VARIABLE | | UNIT |
|---|---|---|---|---|---|---|
| | | MIN. | MAX. | MIN. | MAX. | |
| $t_{LL}$ | ALE pulse duration | 127 | – | $2t_{CK} - 40$ | – | ns |
| $t_{AL}$ | Address set-up time to ALE | 43 | – | $t_{CK} - 40$ | – | ns |
| $t_{LA}$ | Address hold time after ALE | 48 | – | $t_{CK} - 35$ | – | ns |
| $t_{LIV}$ | Time from ALE to valid instruction input | – | 233 | – | $4t_{CK} - 100$ | ns |
| $t_{LC}$ | Time from ALE to control pulse $\overline{PSEN}$ | 58 | – | $t_{CK} - 25$ | – | ns |
| $t_{CC}$ | Control pulse duration $\overline{PSEN}$ | 215 | – | $3t_{CK} - 35$ | – | ns |
| $t_{CIV}$ | Time from $\overline{PSEN}$ to valid instruction input | – | 125 | – | $3t_{CK} - 125$ | ns |
| $t_{CI}$ | Input instruction hold time after $\overline{PSEN}$ | 0 | – | 0 | – | ns |
| $t_{CIF}$ | Input instruction float delay after $\overline{PSEN}$ | – | 63 | – | $t_{CK} - 20$ | ns |
| $t_{AC}$ | Address valid after $\overline{PSEN}$ | 75 | – | $t_{CK} - 8$ | – | ns |
| $t_{AIV}$ | Address to valid instruction input | – | 302 | – | $5t_{CK} - 115$ | ns |
| $t_{AFC}$ | Address float delay after $\overline{PSEN}$ | 12 | – | 0 | – | ns |

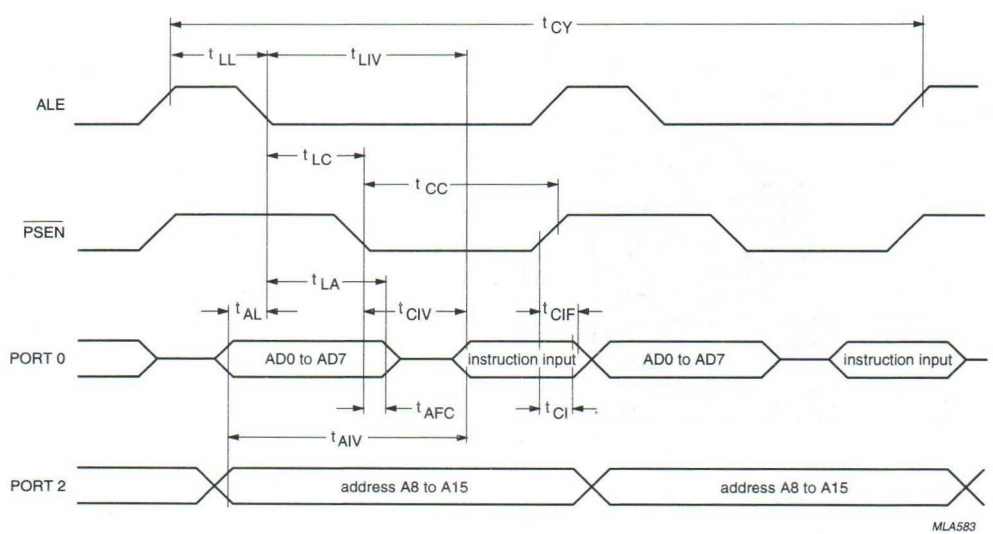



Fig.31 Read from program memory.

### 10.2 External data memory

$V_{DD}$ = 5 V; $V_{SS}$ = 0 V; $T_{amb}$ = −40 to +85 °C for P83CL781 and −25 to +55 °C for the P83CL782; $C_L$ = 50 pF for Port 0, ALE and $\overline{PSEN}$; $C_L$ = 40 pF for all other outputs unless specified. See note 1 and Figs 32 and 33.

| SYMBOL | PARAMETER | $f_{osc}$ = 12 MHz | | $f_{osc}$ = VARIABLE | | UNIT |
|---|---|---|---|---|---|---|
| | | MIN. | MAX. | MIN. | MAX. | |
| $t_{RR}$ | RD pulse duration | 400 | – | $6t_{CK} - 100$ | – | ns |
| $t_{WW}$ | WR pulse duration | 400 | – | $6t_{CK} - 100$ | – | ns |
| $t_{LA}$ | Address hold time after ALE | 48 | – | $t_{CK} - 35$ | – | ns |
| $t_{RD}$ | RD to valid data input | – | 150 | – | $5t_{CK} - 165$ | ns |
| $t_{DFR}$ | Data float delay after $\overline{RD}$ | – | 97 | – | $2t_{CK} - 70$ | ns |
| $t_{LD}$ | Time from ALE to valid data input | – | 517 | – | $8t_{CK} - 150$ | ns |
| $t_{AD}$ | Address to valid data input | – | 585 | – | $9t_{CK} - 165$ | ns |
| $t_{LW}$ | Time from ALE to $\overline{RD}$ or $\overline{WR}$ | 200 | 300 | $3t_{CK} - 50$ | $3t_{CK} + 50$ | ns |
| $t_{AW}$ | Time from address to $\overline{RD}$ or $\overline{WR}$ | 203 | – | 4 | – | ns |
| $t_{WHLH}$ | Time from $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH | 43 | 123 | $t_{CK} - 40$ | $t_{CK} + 40$ | ns |
| $t_{DWX}$ | Data valid to $\overline{WR}$ transition | 23 | – | $t_{CK} - 60$ | – | ns |
| $t_{DW}$ | Data set-up time before $\overline{WR}$ | 433 | – | $7t_{CK} - 150$ | – | ns |
| $t_{WD}$ | Data hold time after $\overline{WR}$ | 33 | – | $t_{CK} - 50$ | – | ns |
| $t_{AFR}$ | Address float delay after $\overline{RD}$ | – | 12 | – | 12 | ns |

**Note**

1. Interfacing the P83CL781 or the P83CL782 to devices with float times up to 75 ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.

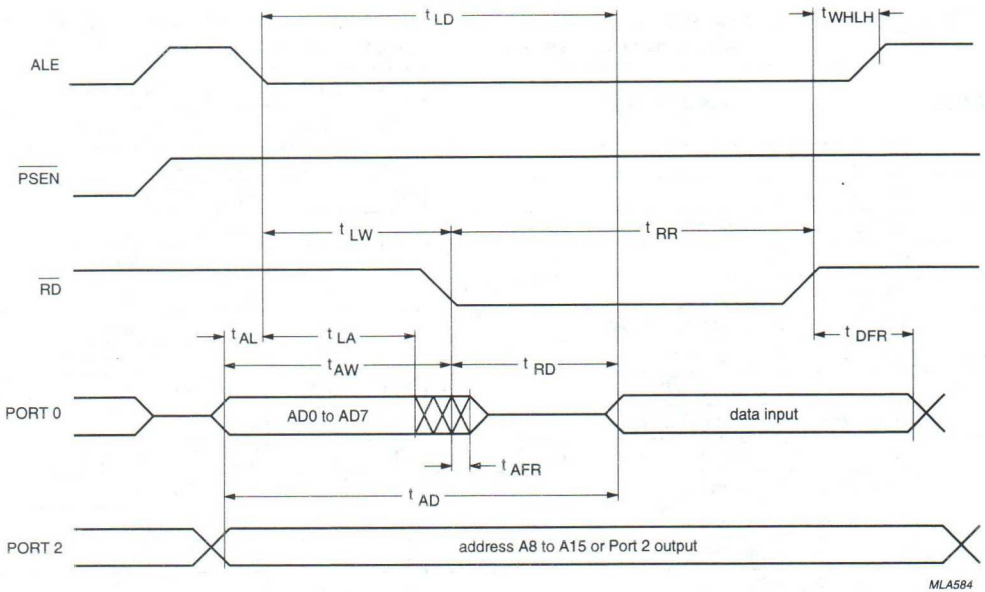



Fig.32 Read from data memory.

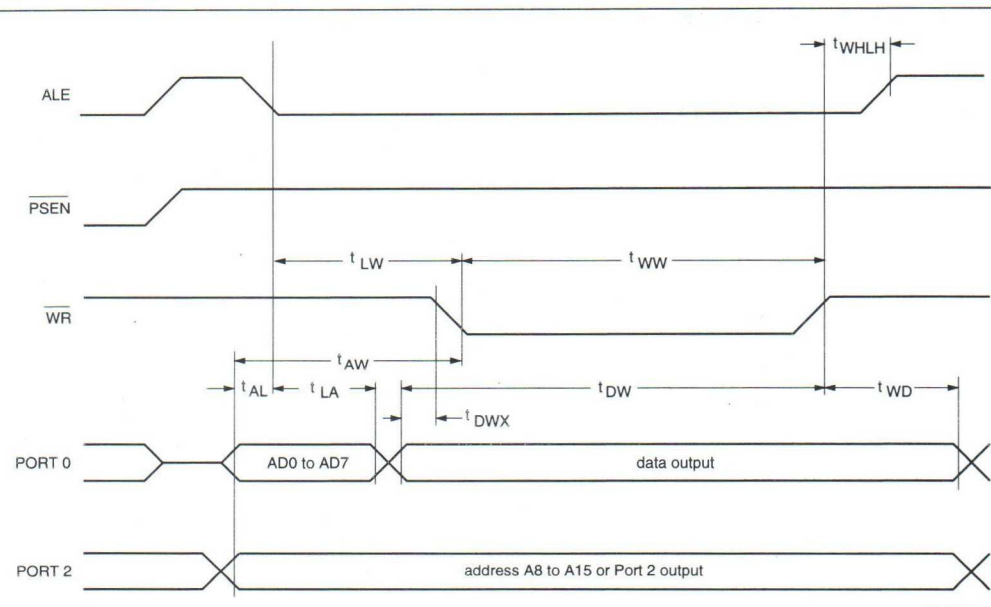



Fig.33 Write to data memory.

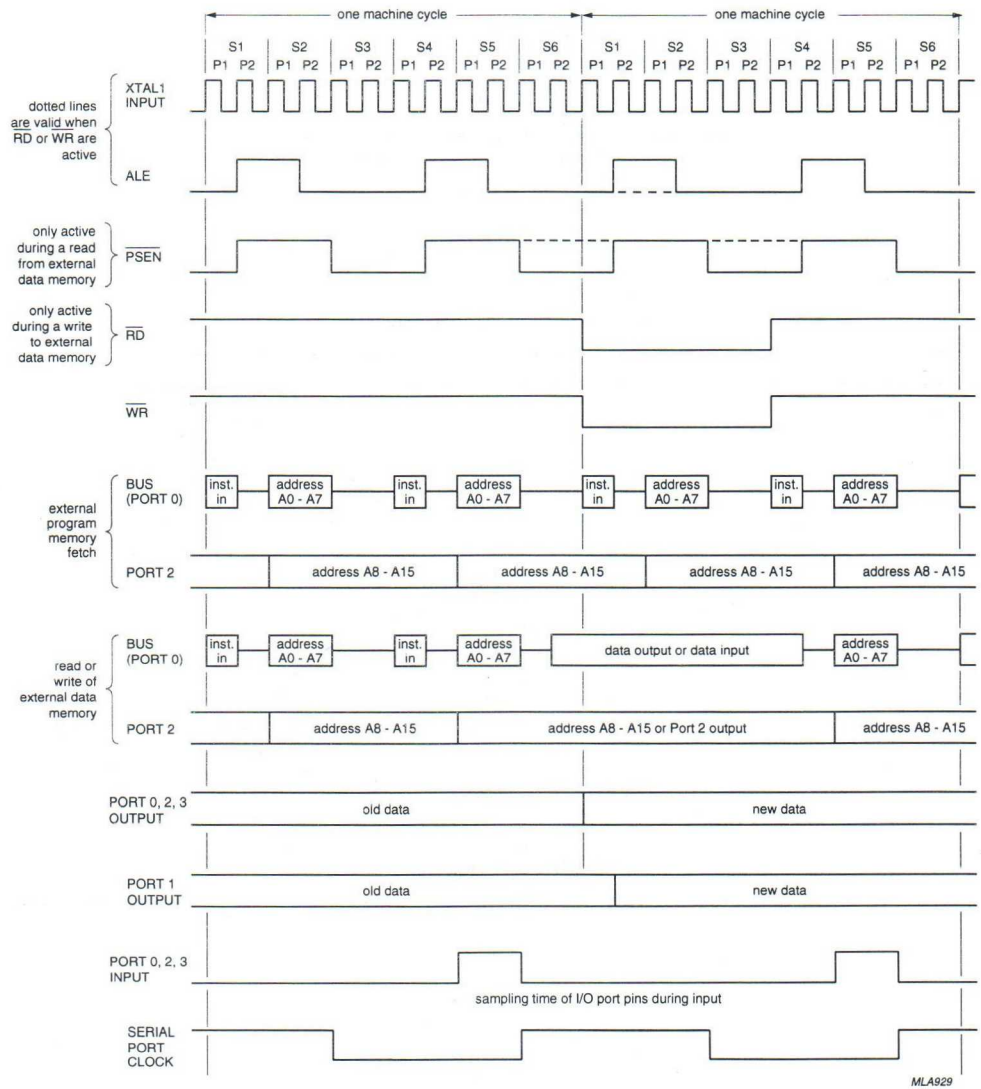



Fig.34 Instruction cycle timing.

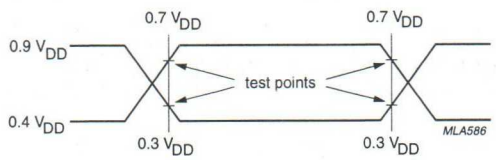



Fig.35 AC testing input waveform.

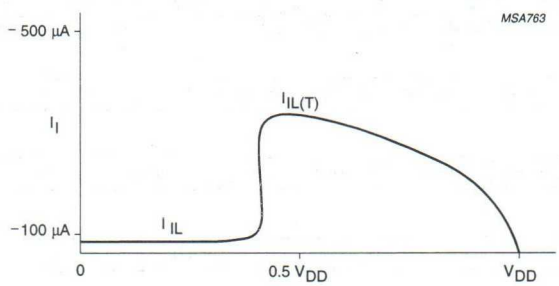



Fig.36 Input current.

# CMOS single-chip 8-bit microcontroller with on-chip EEPROM

# 80C851/83C851

## DESCRIPTION

The Philips 80C851/83C851 is a high-performance microcontroller fabricated with Philips high-density CMOS technology. The 80C851/83C851 has the same instruction set as the 80C51. The Philips CMOS technology combines the high speed and density characteristics of HMOS with the low power attributes of CMOS. The Philips epitaxial substrate minimizes latch-up sensitivity.

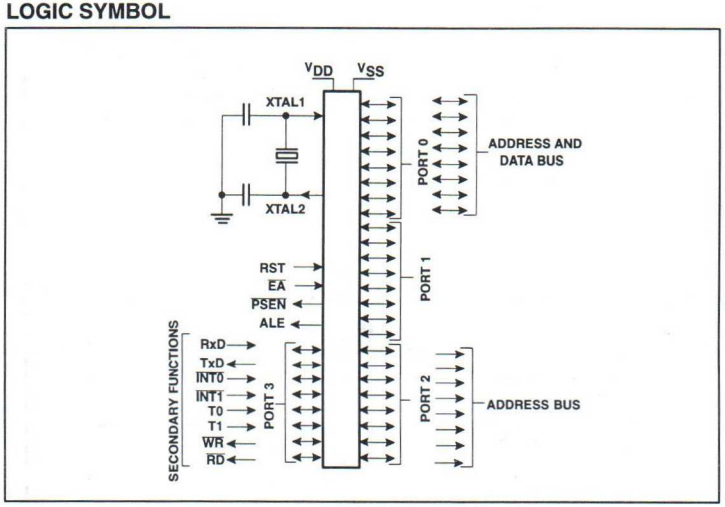
The 80C851/83C851 contains a 4k × 8 ROM with mask-programmable ROM code protection, a 128 × 8 RAM, 256 × 8 EEPROM, 32 I/O lines, two 16-bit counter/timers, a seven-source, five vector, two-priority level nested interrupt structure, a serial I/O port for either multi-processor communications, I/O expansion or full duplex UART, and on-chip oscillator and clock circuits.

In addition, the 80C851/83C851 has two software selectable modes of power reduction — idle mode and power-down mode. The idle mode freezes the CPU while allowing the RAM, timers, serial port, and interrupt system to continue functioning. The power-down mode saves the RAM and EEPROM contents but freezes the oscillator, causing all other chip functions to be inoperative.

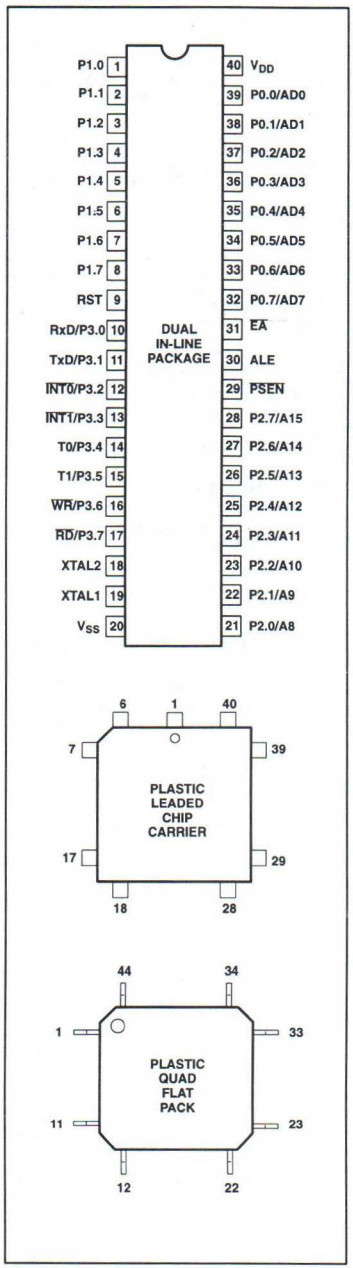
## FEATURES

- 80C51 based architecture
  - 4k × 8 ROM
  - 128 × 8 RAM
  - Two 16-bit counter/timers
  - Full duplex serial channel
  - Boolean processor
- Non-volatile 256 × 8-bit EEPROM (electrically erasable programmable read only memory)
  - On-chip voltage multiplier for erase/write
  - 50,000 erase/write cycles per byte
  - 10 years non-volatile data retention
  - Infinite number of read cycles
  - User selectable security mode
  - Block erase capability
- Mask-programmable ROM code protection
- Memory addressing capability
  - 64k ROM and 64k RAM
- Power control modes:
  - Idle mode
  - Power-down mode
- CMOS and TTL compatible
- 1.2 to 16MHz
- Three package styles
- Three temperature ranges
- ROM code protection

## LOGIC SYMBOL




## PIN CONFIGURATIONS

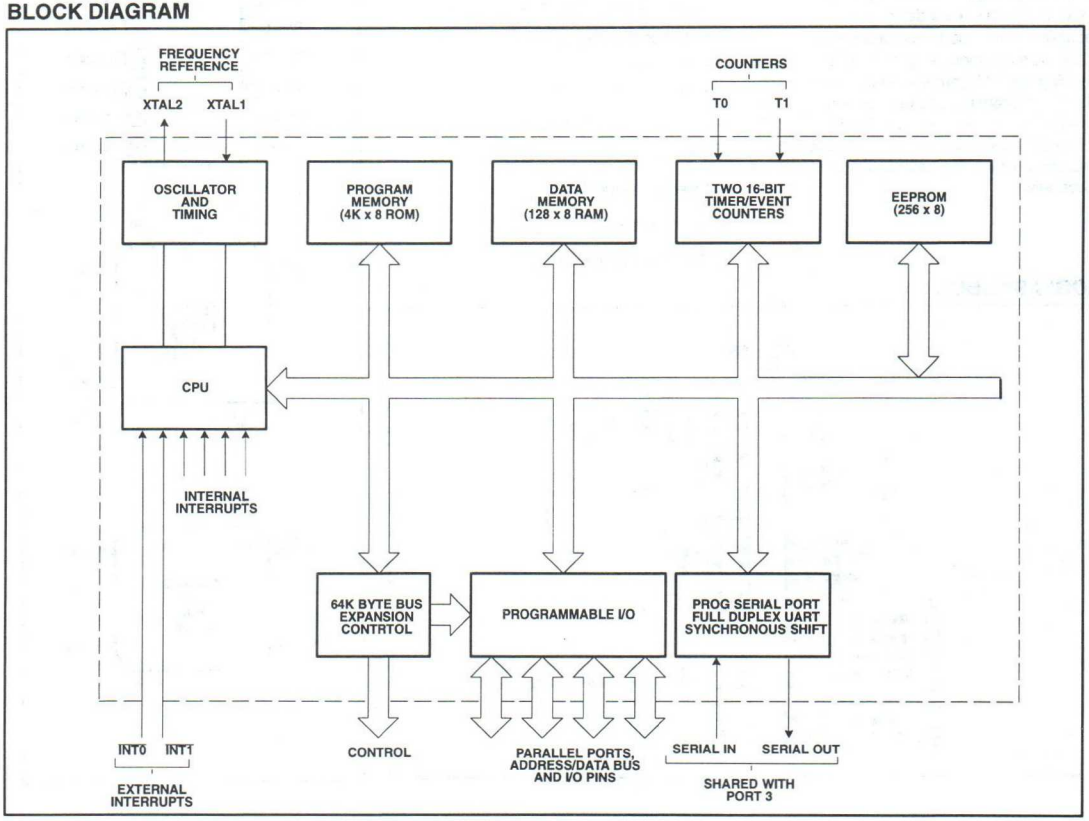


## ORDERING INFORMATION

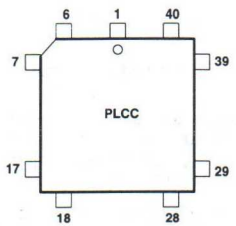
| PHILIPS PART ORDER NUMBER PART MARKING | | NORTH AMERICA PHILIPS PART ORDER NUMBER | | TEMPERATURE RANGE °C AND PACKAGE | FREQ. (MHz) | DRAWING NUMBER |
|---|---|---|---|---|---|---|
| ROMless Version | ROM Version | ROMless Version | ROM Version | | | |
| P80C851 FBP | P83C851 FBP | S80C851-4N40 | S83C851-4N40 | 0 to +70, Plastic Dual In-line Package | 1.2 to 16 | SOT129-1 |
| P80C851 FBA | P83C851 FBA | S80C851-4A44 | S83C851-4A44 | 0 to +70, Plastic Leaded Chip Carrier | 1.2 to 16 | SOT187-1 |
| P80C851 FBB | P83C851 FBB | S80C851-4B44 | S83C851-4B44 | 0 to +70, Plastic Quad Flat Pack | 1.2 to 16 | SOT307-2 [1] |
| P80C851 FFP | P83C851 FFP | S80C851-5N40 | S83C851-5N40 | –40 to +85, Plastic Dual In-line Package | 1.2 to 16 | SOT129-1 |
| P80C851 FFA | P83C851 FFA | S80C851-5A44 | S83C851-5A44 | –40 to +85, Plastic Leaded Chip Carrier | 1.2 to 16 | SOT187-1 |
| P80C851 FFB | P83C851 FFB | S80C851-5B44 | S83C851-5B44 | –40 to +85, Plastic Quad Flat Pack | 1.2 to 16 | SOT307-2 [1] |
| P80C851 FHP | P83C851 FHP | S80C851-6N40 | S83C851-6N40 | –40 to +125, Plastic Dual In-line Package | 1.2 to 16 | SOT129-1 |
| P80C851 FHA | P83C851 FHA | S80C851-6A44 | S83C851-6A44 | –40 to +125, Plastic Leaded Chip Carrier | 1.2 to 16 | SOT187-1 |
| P80C851 FHB | P83C851 FHB | S80C851-6B44 | S83C851-6B44 | –40 to +125, Plastic Quad Flat Pack | 1.2 to 16 | SOT307-2 [1] |

**NOTE:**
1. SOT311 replaced by SOT307-2.

## BLOCK DIAGRAM



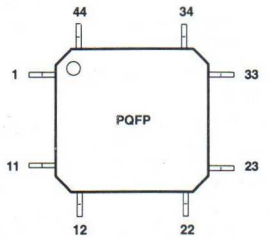
## PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC* | 23 | NC* |
| 2 | P1.0 | 24 | P2.0/A8 |
| 3 | P1.1 | 25 | P2.1/A9 |
| 4 | P1.2 | 26 | P2.2/A10 |
| 5 | P1.3 | 27 | P2.3/A11 |
| 6 | P1.4 | 28 | P2.4/A12 |
| 7 | P1.5 | 29 | P2.5/A13 |
| 8 | P1.6 | 30 | P2.6/A14 |
| 9 | P1.7 | 31 | P2.7/A15 |
| 10 | RST | 32 | $\overline{PSEN}$ |
| 11 | P3.0/RxD | 33 | ALE |
| 12 | NC* | 34 | NC* |
| 13 | P3.1/TxD | 35 | $\overline{EA}$ |
| 14 | P3.2/$\overline{INT0}$ | 36 | P0.7/AD7 |
| 15 | P3.3/$\overline{INT1}$ | 37 | P0.6/AD6 |
| 16 | P3.4/T0 | 38 | P0.5/AD5 |
| 17 | P3.5/T1 | 39 | P0.4/AD4 |
| 18 | P3.6/$\overline{WR}$ | 40 | P0.3/AD3 |
| 19 | P3.7/$\overline{RD}$ | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | $V_{SS}$ | 44 | $V_{DD}$ |

* DO NOT CONNECT

## PLASTIC QUAD FLAT PACK PIN FUNCTIONS




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5 | 23 | P2.5/A13 |
| 2 | P1.6 | 24 | P2.6/A14 |
| 3 | P1.7 | 25 | P2.7/A15 |
| 4 | RST | 26 | $\overline{PSEN}$ |
| 5 | P3.0/RxD | 27 | ALE |
| 6 | NC* | 28 | NC* |
| 7 | P3.1/TxD | 29 | $\overline{EA}$ |
| 8 | P3.2/$\overline{INT0}$ | 30 | P0.7/AD7 |
| 9 | P3.3/$\overline{INT1}$ | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/$\overline{WR}$ | 34 | P0.3/AD3 |
| 13 | P3.7$\overline{RD}$ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | $V_{SS}$ | 38 | $V_{DD}$ |
| 17 | NC* | 39 | $V_{SS}$ |
| 18 | P2.0/A8 | 40 | P1.0 |
| 19 | P2.1/A9 | 41 | P1.1 |
| 20 | P2.2/A10 | 42 | P1.2 |
| 21 | P2.3/A11 | 43 | P1.3 |
| 22 | P2.4/A12 | 44 | P1.4 |

* DO NOT CONNECT

## PIN DESCRIPTION

| MNEMONIC | PIN NO. DIP | PIN NO. LCC | PIN NO. QFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16, 39 | I | **Ground:** 0V reference. |
| $V_{DD}$ | 40 | 44 | 38 | I | **Power Supply:** This is the power supply voltage for normal, idle, and power-down operation. |
| P0.0–0.7 | 39–32 | 43–36 | 37–30 | I/O | **Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. |
| P1.0–P1.7 | 1–8 | 2–9 | 40–44, 1–3 | I/O | **Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). |
| P2.0–P2.7 | 21–28 | 24–31 | 18–25 | I/O | **Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register. |
| P3.0–P3.7 | 10–17 | 11, 13–19 | 5, 7–13 | I/O | **Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: $I_{IL}$). Port 3 also serves the special features of the SC80C51 family, as listed below: |
| | 10 | 11 | 5 | I | **RxD (P3.0):** Serial input port |
| | 11 | 13 | 7 | O | **TxD (P3.1):** Serial output port |
| | 12 | 14 | 8 | I | **$\overline{INT0}$ (P3.2):** External interrupt |
| | 13 | 15 | 9 | I | **$\overline{INT1}$ (P3.3):** External interrupt |
| | 14 | 16 | 10 | I | **T0 (P3.4):** Timer 0 external input |
| | 15 | 17 | 11 | I | **T1 (P3.5):** Timer 1 external input |
| | 16 | 18 | 12 | O | **$\overline{WR}$ (P3.6):** External data memory write strobe |
| | 17 | 19 | 13 | O | **$\overline{RD}$ (P3.7):** External data memory read strobe |
| RST | 9 | 10 | 4 | I | **Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to $V_{SS}$ permits a power-on reset using only an external capacitor to $V_{DD}$. |
| ALE | 30 | 33 | 27 | I/O | **Address Latch Enable:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. |
| $\overline{PSEN}$ | 29 | 32 | 26 | O | **Program Store Enable:** The read strobe to external program memory. When the device is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. |
| $\overline{EA}$ | 31 | 35 | 29 | I | **External Access Enable:** If during a RESET, $\overline{EA}$ is held at TTL, level HIGH, the CPU executes out of the internal program memory ROM provided the Program Counter is less than 4096. If during a RESET, $\overline{EA}$ is held a TTL LOW level, the CPU executes out of external program memory. $\overline{EA}$ is not allowed to float. |
| XTAL1 | 19 | 21 | 15 | I | **Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. |
| XTAL2 | 18 | 20 | 14 | O | **Crystal 2:** Output from the inverting oscillator amplifier. |

**Table 1. 8XC851 Special Function Registers**

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB | | | | | | | LSB | RESET VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACC* | Accumulator | E0H | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | 00H |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| | | | EF | EE | ED | EC | EB | EA | E9 | E8 | |
| DPTR: | Data pointer (2 bytes): | | | | | | | | | | |
| DPH | High byte | 83H | | | | | | | | | 00H |
| DPL | Low byte | 82H | | | | | | | | | 00H |
| EADRH# | EEPROM addr reg-high | F3H | | | | | | | | | 80H |
| EADRL# | EEPROM addr reg-low | F2H | | | | | | | | | 00H |
| ECNTRL# | EEPROM control reg | F6H | IFE | EEINT | EWP | – | ECNTR L3 | ECNTR L2 | ECNTR L1 | ECNTR L0 | 00H |
| EDAT# | EEPROM data register | F4H | | | | | | | | | xxH |
| ETIM# | EEPROM timer register | F5H | | | | | | | | | 08H |
| | | | BF | BE | BD | BC | BB | BA | B9 | B8 | |
| IP* | Interrupt priority | B8H | – | – | – | PS | PT1 | PX1 | PT0 | PX0 | xxx00000B |
| | | | AF | AE | AD | AC | AB | AA | A9 | A8 | |
| IE* | Interrupt enable | A8H | EA | – | – | ES | ET1 | EX1 | ET0 | EX0 | 0xx00000B |
| P0* | Port 0 | 80H | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | FFH |
| P1* | Port 1 | 90H | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | FFH |
| P2* | Port 2 | A0H | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | FFH |
| P3* | Port 3 | B0H | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | FFH |
| PCON | Power control | 87H | SMOD | – | – | – | GF1 | GF0 | PD | IDL | 0xxx0000B |
| | | | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| PSW* | Program status word | D0H | CY | AC | F0 | RS1 | RS0 | OV | – | P | 00H |
| SBUF | Serial data buffer | 99H | | | | | | | | | xxxxxxxxB |
| | | | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | |
| SCON* | Serial port control | 98H | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | 00H |
| SP | Stack pointer | 81H | | | | | | | | | 07H |
| | | | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | 00H |
| TCON* | Timer/counter control | 88H | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 00H |
| TMOD | Timer/counter mode | 89H | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 | 00H |
| TH0 | Timer 0 high byte | 8CH | | | | | | | | | 00H |
| TH1 | Timer 1 high byte | 8DH | | | | | | | | | 00H |
| TL0 | Timer 0 low byte | 8AH | | | | | | | | | 00H |
| TL1 | Timer 1 low byte | 8BH | | | | | | | | | 00H |

* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.

## EEPROM

Communications between the CPU and the EEPROM is accomplished via 5 special function registers; 2 address registers (high and low byte), 1 data register for read and write operations, 1 control register, and 1 timer register to adapt the erase/write time to the clock frequency. All registers can be read and written. Figure 1 shows a block diagram of the CPU, the EEPROM and the interface.

## Register and Functional Description

**Address Register (EADRH, EADRL)**
The lower byte contains the address of one of the 256 bytes. The higher byte (EADRH) is for future extensions and for addressing the security bits (see Security Facilities). The EADRH register address is F3H. The EADRL register address is F2H.

**Data Register (EDAT)**
This register is required for read and write operations and also for row/block erase. In write mode, its contents are written to the addressed byte (for "row erase" and "block erase" the contents are don't care). The write pulse starts all operations, except read. In read mode, EDAT contains the data of the addressed byte. The EDAT register address is F4H.

**Timer Register (ETIM)**
The timer register is required to adapt the erase/write time to the oscillator frequency. The user has to ensure that the erase or write (program) time is neither too short or too long.

The ETIM register address is F5H. Table 2 contains the values which must be written to the ETIM register by software for various oscillator frequencies (the default value is 08H after RESET).

The general formula is:

2ms Write time:

$$\text{Value (decimal, to be rounded up)} = \frac{f_{XTAL1}\ [kHz]}{512} - 2$$

10ms Write time:

$$\text{Value (decimal)} = \frac{f_{XTAL1}\ [kHz]}{96} - 2$$

**Control Register (ECNTRL)**
See Figure 2 for a description of this register. The ECNTRL register address is F6H.

**Table 2. Values for the Timer Register (ETIM)**

| $f_{XTAL1}$ | VALUES FOR ETIM | | | |
|---|---|---|---|---|
| | 2ms WRITE TIME | | 10ms WRITE TIME | |
| | HEX | DEC | HEX | DEC |
| 1.0MHz | – | – | 08 | 8 |
| 2.0MHz | 02 | 2 | 13 | 19 |
| 3.0MHz | 04 | 4 | 1D | 29 |
| 4.0MHz | 06 | 6 | 28 | 40 |
| 5.0MHz | 08 | 8 | 32 | 50 |
| 6.0MHz | 0A | 10 | 3C | 60 |
| 7.0MHz | 0C | 12 | 47 | 71 |
| 8.0MHz | 0E | 14 | 51 | 81 |
| 9.0MHz | 10 | 16 | 5C | 92 |
| 10.0MHz | 12 | 18 | 66 | 102 |
| 11.0MHz | 14 | 20 | 71 | 113 |
| 12.0MHz | 16 | 22 | 7B | 123 |
| 13.0MHz | 18 | 24 | | |
| 14.0MHz | 1A | 26 | | |
| 15.0MHz | 1C | 28 | | |
| 16.0MHz | 1E | 30 | | |

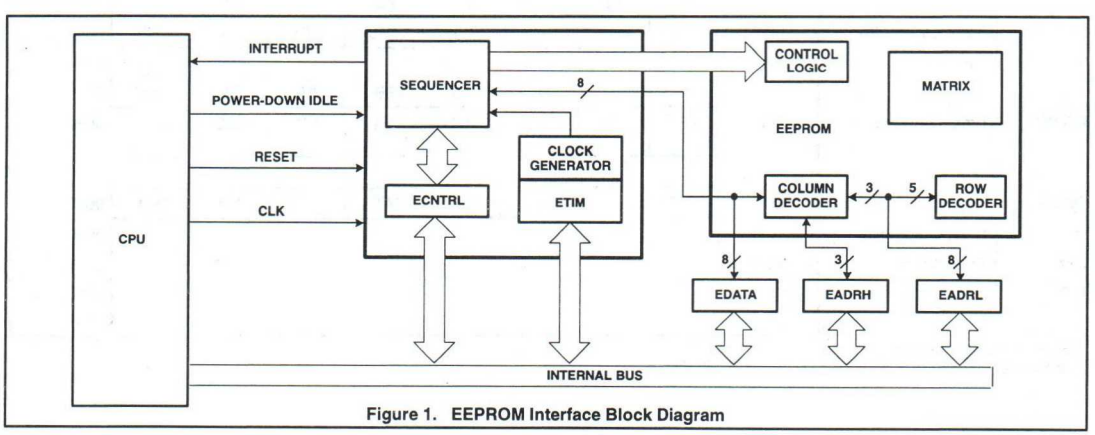



**Figure 1. EEPROM Interface Block Diagram**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| IFE | EEINT | EWP | – – | ECNTRL3 | ECNTRL2 | ECNTRL1 | ECNTRL0 |

| Bit | Symbol | Function |
|---|---|---|
| ECNTRL.7 | IFE | Active high EEPROM interrupt flag: set by the sequencer or by software; reset by software.<br>When set and enabled, this flag forces an interrupt to the same vector as the serial port interrupt (see Interrupt section). |
| ECNTRL.6 | EEINT | EEPROM interrupt enable: set and reset by software (active high). |
| ECNTRL.5 | EWP | Erase/write in progress flag: set and reset by the sequencer (active high).<br>When EWP is set, access to the EEPROM is not possible. EWP cannot be set or reset by software. |
| ECNTRL.4 | | Reserved. |
| ECCTRL.3–<br>ECNTRL.0 | | See table below. |

| Operation | ECNTRL.3 | ECNTRL.2 | ECNTRL.1 | ECNTRL.0 |
|---|---|---|---|---|
| Byte mode | 0 | 0 | 0 | 0 |
| Row erase | 1 | 1 | 0 | 0 |
| Page write* | – | – | – | – |
| Page erase/write* | – | – | – | – |
| block erase | 1 | 0 | 1 | 0 |

*Future products.

Byte mode: Normal EEPROM mode, default mode after reset. In this mode, data can be read and written to one byte at a time.

Read mode: This is the default mode when byte mode is selected. This means that the contents of the addressed byte are available in the data register.

Write mode: This mode is activated by writing to the data register. The address register must be loaded first. Since the old contents are read first (by default), this allows the sequencer to decide whether an erase/write or write cycle only (data = 00H) is required.

Row erase: In this mode, the addressed row is cleared. The three LSBs of EADRL are not significant, i.e. the 8 bytes addressed by EADRL are cleared in the same time normally needed to clear one byte ($t_{ROWERASE} = t_E = t_W$). For the following write modes, only the write and not the erase/write cycle is required. For example, using the row erase mode, programming 8 bytes takes $t_{TOTOAL} = t_E + 8 \times t_W$ compared to $t_{TOTAL} = 8 \times t_E + 8 \times t_W$ ($t_E = t_{ERASE} \cdot t_W = t_{WRITE}$).

Page write: For future products.

Page erase/write: For future products.

Block erase: In this mode all 256 bytes are cleared. The byte containing the security bits is also cleared. $t_{BLOCKERASE} = t_E$. The contents of EADRH, EADRL and EDAT are insignificant.

**Program Sequences and Register Contents after Reset**

The contents of the EEPROM registers after a Reset are the default values:

| | | |
|---|---|---|
| EADRH | = 1xxxxxxxB | (security bit address) |
| EADRL | = 00H | (security bit address) |
| ETIM | = 08H | (minimum erase time with the lowest permissible oscillator frequency) |
| ECNTRL | = 00H | (Byte mode, read) |
| EDAT | = xxH | (security bit) |

```
Initialize:   MOV ETIM, ..
              MOV EADRH, ..

Read:         MOV EADRL, ..
              MOV .., EDAT

Write:        MOV EADRL, ..
              MOV EDAT, ..

Erase row:    MOV EADRL, ..          Row address. 3LSBs don't care
              MOV ECNTRL, #0CH       Erase row mode
              MOV EDAT, ..  (EDAT) don't care

Erase block:  MOV ECNTRL, 0AH        Erase block mode
              MOV EDAT, ..  (EDAT) don't care
```

If the security bit is to be altered, the program generally starts as follows:

```
MOV EADRH, #80H
MOV EADRL, #00H
```

**Figure 2. Control Register (ECNTRL)**

## Security Facilities

**EEPROM Protection**

The EEPROM is protected using four security bits which are contained in an extra EEPROM byte at address 8000H (EADRH/EADRL). They can be set or cleared by software. To activate the EEPROM protection, the program sequence in byte mode must be as follows:

```
MOV   EADRH, #80H
MOV   EADRL, #00H
MOV   EDAT, #FFH
```

If two or more of these bits are reset, SB = 0, the security mode is disabled and the EEPROM is not protected. If three or four bits are set, SB = 1 and the $\overline{EA}$ mode differs from the internal access mode.

In this case, access to the EEPROM is only possible in one mode regardless of how the external access mode is reached (by pulling the $\overline{EA}$ pin low or by passing the 4K boundary). For SB = 1 and "external access" only, the "block erase" mode is enabled. The program sequence has to be as follows:

```
MOV   EADRH, #80H (security byte address)
MOV   EADRL, #00H (security byte address)
MOV   ECNTRL, #0AH (block erase mode)
MOV   EDAT, #xxH     (start block erase)
```

All 256 data bytes, the security bits, and SB will be cleared after completing this mode (EWP = 0). SB will also be affected in byte mode when writing to the security byte (not for SB = 1 and "external access"). Figure 3 illustrates the access to SB.

**ROM Code Protection**

Since the external access mode can only be selected by pulling the $\overline{EA}$ pin low during reset, it is not possible to read the internal program memory using the MOVC instruction while executing external program memory. Furthermore, it is not possible to change this mode to internal access within the MOVC cycle.

Additionally, a mask-programmable ROM code protection facility is available. When the program memory passes the 4K boundary using both the internal and external ROMs, it is not possible to access the internal ROM from the external program memory if the mask-programmable ROM security bit is set. An access to the lower 4K bytes of program memory using the MOVC instruction is only possible while executing internal program memory.

Also the verification mode (test-mode which writes the ROM contents to a port for comparison with a reference code) is not implemented for security reasons. A different test-mode is implemented for test purposes. This mode allows every bit to be tested. However, the internal code cannot be accessed via a port.

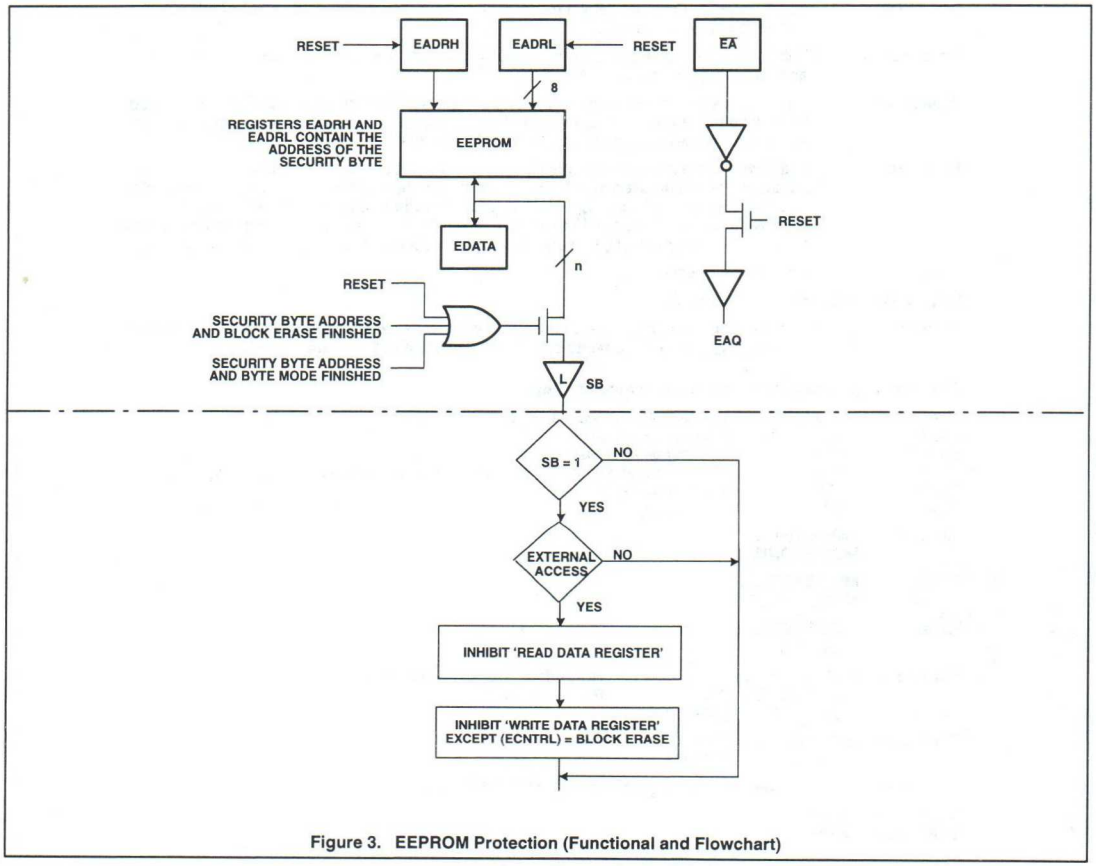



**Figure 3. EEPROM Protection (Functional and Flowchart)**

## OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the logic symbol, page 3-1251.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

## RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-up reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-up, the voltage on $V_{DD}$ and RST must come up at the same time for a proper start-up.

**Note:** Before entering the idle or power-down modes, the user has to ensure that there is no EEPROM erase/write cycle in progress (i.e., the EWP bit has to be reset before activating the idle or power-down modes; otherwise EEPROM accesses will be aborted).

## IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

## POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM and EEPROM are preserved. A hardware reset is the only way to terminate the power-down mode. The control bits for the reduced power modes are in the special function register PCON. Table 3 shows the state of the I/O ports during low current operating modes.

## INTERRUPT SYSTEM

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution, a multiple-source, two-priority-level, nested interrupt system is provided. Interrupt response latency is from 3µs to 7µs when using a 12MHz crystal. The S83C851 acknowledges interrupt requests from 7 sources as follows:

- $\overline{INT0}$ and $\overline{INT1}$: externally via pins 12 and 13, respectively,
- Timer 0 and timer 1: from the two internal counters,
- Serial port: from the internal serial I/O port or EEPROM (1 vector).

Each interrupt vectors to a separate location in program memory for its service program. Each source can be individually enabled (the EEPROM interrupt can only be enabled when the serial port interrupt is enabled) or disabled and can be programmed to a high or low priority level. All enabled sources can also be globally disabled or enabled. Both external interrupts can be programmed to be level-activated and are active low to allow "wire-ORing" of several interrupt sources to one input pin.

**Note:** The serial port and EEPROM interrupt flags must be cleared by software; all other flags are cleared by hardware.

**Table 3. External Pin Status During Idle and Power-Down Modes**

| MODE | PROGRAM MEMORY | ALE | $\overline{PSEN}$ | PORT 0 | PORT 1 | PORT 2 | PORT 3 |
|---|---|---|---|---|---|---|---|
| Idle | Internal | 1 | 1 | Data | Data | Data | Data |
| Idle | External | 1 | 1 | Float | Data | Address | Data |
| Power-down | Internal | 0 | 0 | Data | Data | Data | Data |
| Power-down | External | 0 | 0 | Float | Data | Data | Data |

## ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

| PARAMETER | RATING | UNIT |
|---|---|---|
| Storage temperature range | –65 to +150 | °C |
| Voltage on any other pin to $V_{SS}$ | –0.5 to +6.5 | V |
| Input or output DC current on any single I/O pin | ±5 | mA |
| Power dissipation (based on package heat transfer limitations, not device power consumption) | 1.0 | W |

**NOTES:**

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C ($V_{DD}$ = 5V ±10%), –40°C to +85°C ($V_{DD}$ = 5V ±10%), or –40°C to +125°C ($V_{DD}$ = 5V ±10%), $V_{SS}$ = 0V

| SYMBOL | PARAMETER | PART TYPE | TEST CONDITIONS | LIMITS MIN | LIMITS MAX | UNIT |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input low voltage, except $\overline{EA}$ | 0 to +70°C<br>–40 to +85°C<br>–40 to +125°C | | –0.5<br>–0.5<br>–0.5 | $0.2V_{DD}$–0.1<br>$0.2V_{DD}$–0.15<br>$0.2V_{DD}$–0.25 | V<br>V<br>V |
| $V_{IL1}$ | Input low voltage to $\overline{EA}$ | 0 to +70°C<br>–40 to +85°C<br>–40 to +125°C | | –0.5<br>–0.5<br>–0.5 | $0.2V_{DD}$–0.3<br>$0.2V_{DD}$–0.35<br>$0.2V_{DD}$–0.45 | V<br>V<br>V |
| $V_{IH}$ | Input high voltage, except XTAL1, RST | 0 to +70°C<br>–40 to +85°C<br>–40 to +125°C | | $0.2V_{DD}$+0.9<br>$0.2V_{DD}$+1.0<br>$0.2V_{DD}$+1.0 | $V_{DD}$+0.5<br>$V_{DD}$+0.5<br>$V_{DD}$+0.5 | V<br>V<br>V |
| $V_{IH1}$ | Input high voltage, XTAL1, RST | 0 to +70°C<br>–40 to +85°C<br>–40 to +125°C | | $0.7V_{DD}$<br>$0.7V_{DD}$+0.1<br>$0.7V_{D}$+0.1 | $V_{DD}$+0.5<br>$V_{DD}$+0.5<br>$V_{DD}$+0.5 | |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3 [6] | | $I_{OL}$ = 1.6mA[4] | | 0.45 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$ [6] | | $I_{OL}$ = 3.2mA[4] | | 0.45 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3, ALE, $\overline{PSEN}$ | | $I_{OH}$ = –60µA,<br>$I_{OH}$ = –25µA,<br>$I_{OH}$ = –10µA | 2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | V<br>V<br>V |
| $V_{OH1}$ | Output high voltage, port 0 in external bus mode[5] | | $I_{OH}$ = –800µA,<br>$I_{OH}$ = –300µA,<br>$I_{OH}$ = –80µA | 2.4<br>$0.75V_{DD}$<br>$0.9V_{DD}$ | | V<br>V<br>V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3 | 0 to +70°C<br>–40 to +85°C<br>–40 to +125°C | $V_{IN}$ = 0.45V | | –50<br>–75<br>–75 | µA<br>µA<br>µA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3 | 0 to +70°C<br>–40 to +85°C<br>–40 to +125°C | $V_{IN}$ = 2.0V | | –650<br>–750<br>–750 | µA<br>µA<br>µA |
| $I_{L1}$ | Input leakage current, port 0, $\overline{EA}$ | | 0.45V<$V_i$<$V_{DD}$ | | ±10 | µA |
| $I_{DD}$ | Power supply current:<br>Active mode @ 16MHz [1]<br>Idle mode @ 16MHz [2]<br>Power down mode [3] | | See note 7 | | <br>19<br>3.7<br>50 | <br>mA<br>mA<br>µA |
| $R_{RST}$ | Internal reset pull-down resistor | | | 50 | 150 | kΩ |
| $C_{IO}$ | Pin capacitance | | f = 1MHz | | 10 | pF |

**NOTES:**

1. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 5ns; $V_{IL} = V_{SS}$ +0.5V; $V_{IH} = V_{DD}$ – 0.5V; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = $V_{DD}$.
2. The idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 5ns; $V_{IL} = V_{SS}$ +0.5V; $V_{IH} = V_{DD}$ – 0.5V; XTAL2 not connected; $\overline{EA}$ = Port 0 = $V_{DD}$; RST = $V_{SS}$.
3. The power-down current is measured with all output pins disconnected; XTAL2 not connected; $\overline{EA}$ = Port 0 = $V_{DD}$; RST = XTAL1 = $V_{SS}$.
4. Capacitive loading on Port 0 and Port 2 may cause spurious noise pulses to be superimposed on the LOW level output voltage of ALE, Port 1 and Port 3. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make a 1-to-0 transition during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE line may exceed 0.8V. In such cases it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.
5. Capacitive loading on Port 0 and Port 2 may cause the HIGH level output voltage on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{DD}$ specification when the address bits are stabilizing.
6. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per Port pin: 10mA
   Maximum $I_{OL}$ per 8-bit port –
   Port 0: 26mA
   Ports 1, 2, and 3: 15mA
   Maximum total $I_{OL}$ for all output pins: 71mA.
   If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
7. See Figures 11 through 14 for $I_{DD}$ test conditions.

## AC ELECTRICAL CHARACTERISTICS[1, 2]

| SYMBOL | FIGURE | PARAMETER | 16MHz CLOCK | | VARIABLE CLOCK | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 4 | Oscillator frequency | | | 1.2 | 16 | MHz |
| $t_{LHLL}$ | 4 | ALE pulse width | 85 | | $2t_{CLCL}-40$ | | ns |
| $t_{AVLL}$ | 4 | Address valid to ALE low | 8 | | $t_{CLCL}-55$ | | ns |
| $t_{LLAX}$ | 4 | Address hold after ALE low | 28 | | $t_{CLCL}-35$ | | ns |
| $t_{LLIV}$ | 4 | ALE low to valid instruction in | | 150 | | $4t_{CLCL}-100$ | ns |
| $t_{LLPL}$ | 4 | ALE low to $\overline{PSEN}$ low | 23 | | $t_{CLCL}-40$ | | ns |
| $t_{PLPH}$ | 4 | $\overline{PSEN}$ pulse width | 143 | | $3t_{CLCL}-45$ | | ns |
| $t_{PLIV}$ | 4 | $\overline{PSEN}$ low to valid instruction in | | 83 | | $3t_{CLCL}-105$ | ns |
| $t_{PXIX}$ | 4 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 4 | Input instruction float after $\overline{PSEN}$ | | 38 | | $t_{CLCL}-25$ | ns |
| $t_{AVIV}$ | 4 | Address to valid instruction in | | 208 | | $5t_{CLCL}-105$ | ns |
| $t_{PLAZ}$ | 4 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 5, 6 | $\overline{RD}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{WLWH}$ | 5, 6 | $\overline{WR}$ pulse width | 275 | | $6t_{CLCL}-100$ | | ns |
| $t_{RLDV}$ | 5, 6 | $\overline{RD}$ low to valid data in | | 148 | | $5t_{CLCL}-165$ | ns |
| $t_{RHDX}$ | 5, 6 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 5, 6 | Data float after $\overline{RD}$ | | 55 | | $2t_{CLCL}-70$ | ns |
| $t_{LLDV}$ | 5, 6 | ALE low to valid data in | | 350 | | $8t_{CLCL}-150$ | ns |
| $t_{AVDV}$ | 5, 6 | Address to valid data in | | 398 | | $9t_{CLCL}-165$ | ns |
| $t_{LLWL}$ | 5, 6 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 138 | 238 | $3t_{CLCL}-50$ | $3t_{CLCL}+50$ | ns |
| $t_{AW}$ | 5, 6 | Address to $\overline{RD}$ or $\overline{WR}$ | 120 | | $4t_{CLCL}-130$ | | ns |
| $t_{QW}$ | 5, 6 | Data setup time before $\overline{WR}$ | 288 | | $7t_{CLCL}-150$ | | ns |
| $t_{QVWX}$ | 5, 6 | Data valid to $\overline{WR}$ transition | 3 | | $t_{CLCL}-60$ | | ns |
| $t_{WHQX}$ | 5, 6 | Data hold after $\overline{WR}$ | 13 | | $t_{CLCL}-50$ | | ns |
| $t_{RLAZ}$ | 5, 6 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 5, 6 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 23 | 103 | $t_{CLCL}-40$ | $t_{CLCL}+40$ | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 8 | High time | 20 | | 20 | | ns |
| $t_{CLCX}$ | 8 | Low time | 20 | | 20 | | ns |
| $t_{CLCH}$ | 8 | Rise time | | 20 | | 20 | ns |
| $t_{CHCL}$ | 8 | Fall time | | 20 | | 20 | ns |
| **Erase/write timer constant[3]** | | | | | | | |
| $t_{E/W}$ | | Erase/write cycle time | 4 | 20 | 4 | 20 | ms |
| $t_E$ | | Erase time | 2 | 10 | 2 | 10 | ms |
| $t_W$ | | Write time | 2 | 10 | 2 | 10 | ms |
| $t_S$ | | Data retention time[4] | 10 | | 10 | | years |
| NE/W | | Erase/write cycles[5] | 50,000 | | 50,000 | | cycles |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.
3. The power-off fall-time of $V_{DD}$ must be less than 1ms to prevent an overwrite pulse from being generated in the EEPROM which can cause spurious parasitic writing to EEPROM cells. If the $V_{DD}$ power-off full-time is greater than 1ms, a power-off reset signal should be generated to prevent this condition from occurring.
4. Test condition: $T_{amb}$ = +55°C.
5. Number of erase/write cycles for each EEPROM byte.

## EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{\text{PSEN}}$
Q – Output data
R – $\overline{\text{RD}}$ signal
t – Time
V – Valid
W – $\overline{\text{WR}}$ signal
X – No longer a valid logic level
Z – Float

**Examples:** $t_{AVLL}$ = Time for address valid to ALE low.
$t_{LLPL}$ = Time for ALE low to $\overline{\text{PSEN}}$ low.

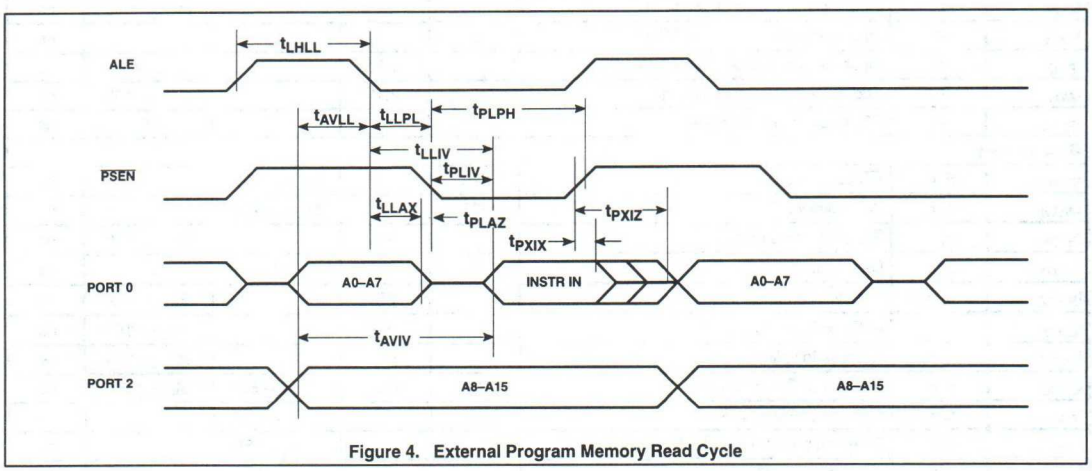



**Figure 4. External Program Memory Read Cycle**

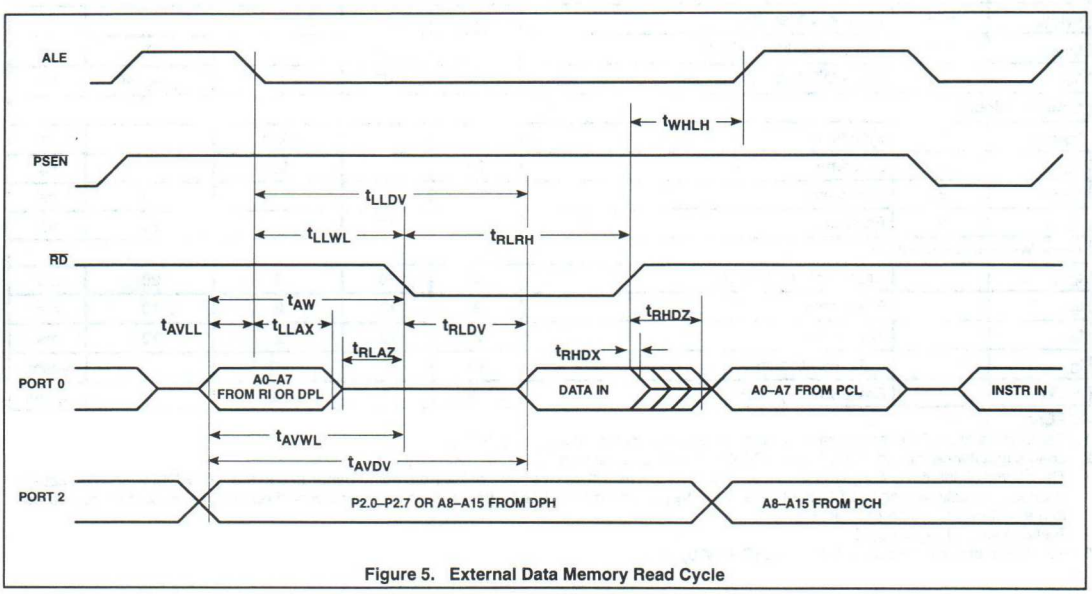



**Figure 5. External Data Memory Read Cycle**

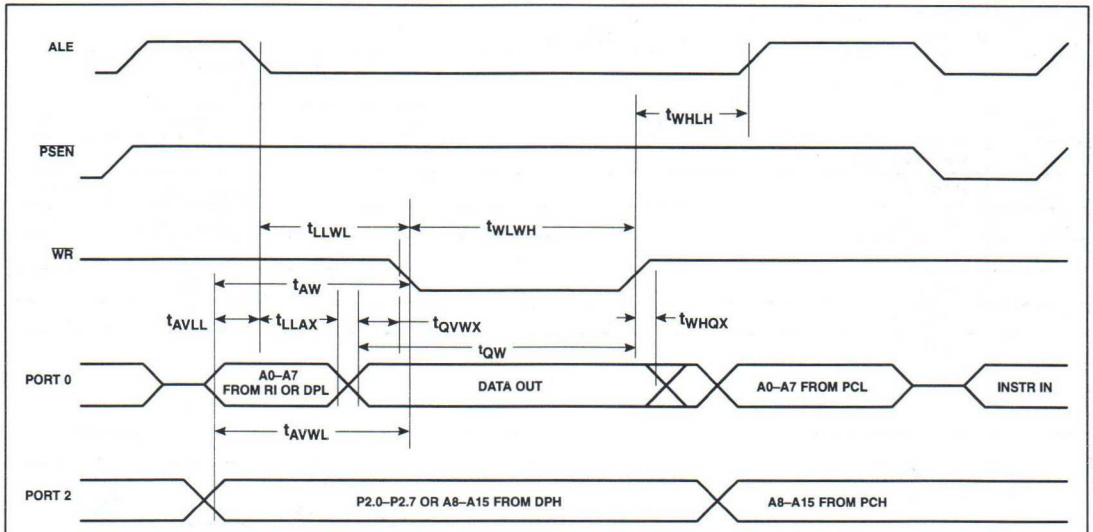



**Figure 6. External Data Memory Write Cycle**

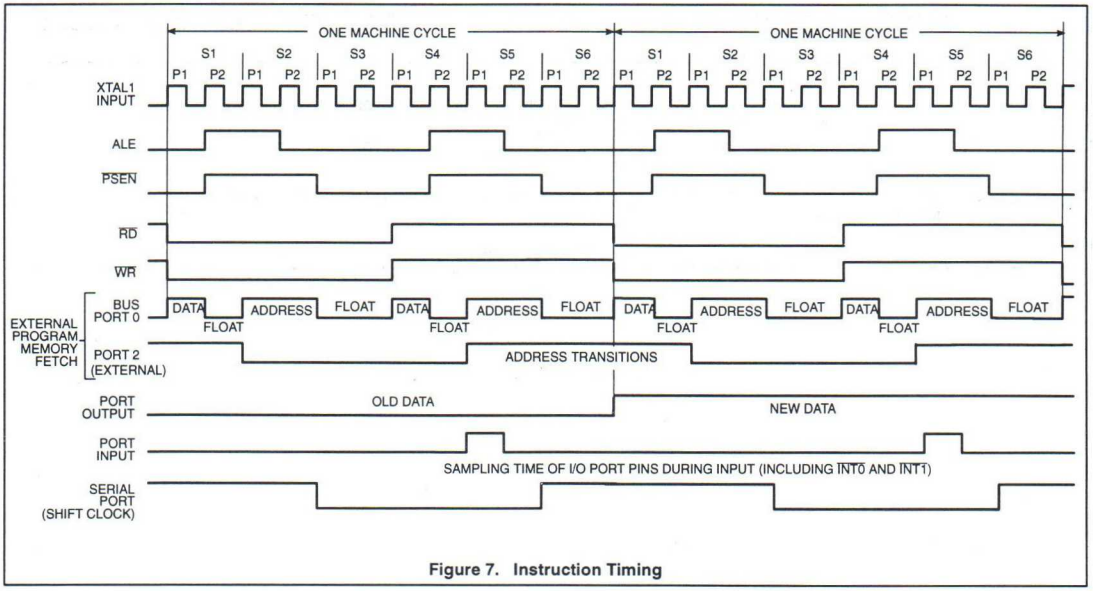



**Figure 7. Instruction Timing**

## Table 4. External Clock Drive XTAL1

Oscillator circuitry: The capacities connected to the crystal should be: C1 = C2 = tbf.

| SYMBOL | PARAMETER | VARIABLE CLOCK f = 1.2 – 16MHz | | UNIT |
|---|---|---|---|---|
| | | MIN | MAX | |
| $t_{CLCL}$ | Oscillator clock period | 63 | 833 | ns |
| $t_{HIGH}$ | HIGH time | 20 | $t_{CLCL} - t_{LOW}$ | ns |
| $t_{LOW}$ | LOW time | 20 | $t_{CLCL} - t_{HIGH}$ | ns |
| $t_r$ | Rise time | – | 20 | ns |
| $t_f$ | Fall time | – | 20 | ns |
| $t_{CY}$ | Cycle time[1] | 0.75 | 10 | ns |

**NOTE:**
1. $t_{CY} = 12\ t_{CLCL}$.

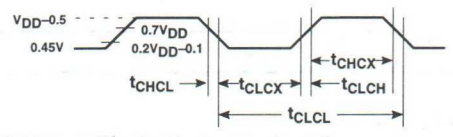
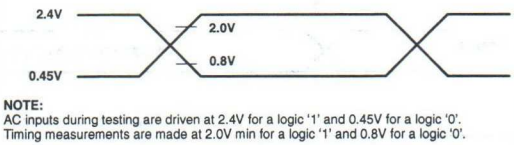
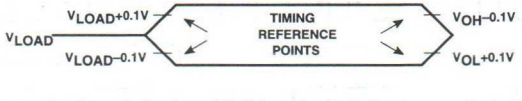



**Figure 8. External Clock Drive**




**NOTE:**
AC inputs during testing are driven at 2.4V for a logic '1' and 0.45V for a logic '0'. Timing measurements are made at 2.0V min for a logic '1' and 0.8V for a logic '0'.

**Figure 9. AC Testing Input/Output**




**NOTE:**
For timing purposes, a port is no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded $V_{OH}/V_{OL}$ level occurs. $I_{OH}/I_{OL} \geq \pm 20$mA.

**Figure 10. Float Waveform**

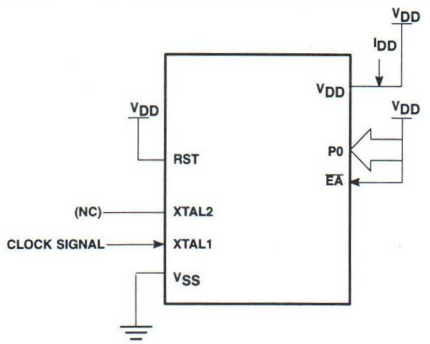



**Figure 11. $I_{DD}$ Test Condition, Active Mode**
**All other pins are disconnected**

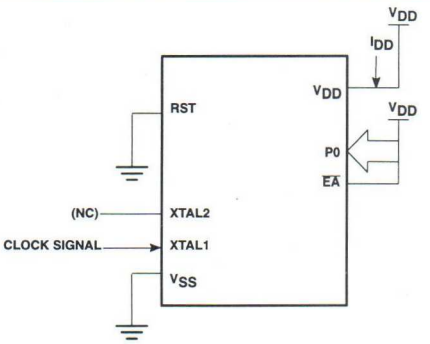



**Figure 12. $I_{DD}$ Test Condition, Idle Mode**
**All other pins are disconnected**

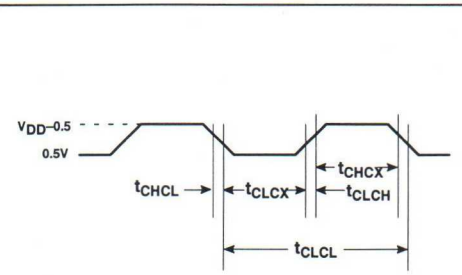



**Figure 13. Clock Signal Waveform for $I_{DD}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

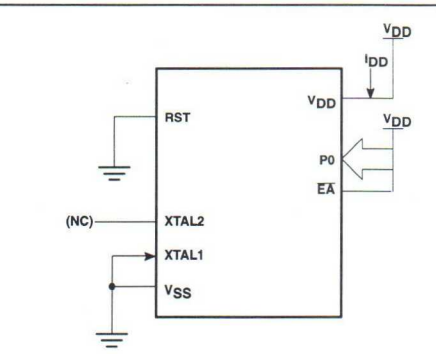



**Figure 14. $I_{DD}$ Test Condition, Power Down Mode**
**All other pins are disconnected.**
**$V_{DD}$ = 2V to 5.5V**

# Section 4
## High Performance 16-bit 80C51 XA (eXtended Architecture)

**80C51-Based 8-Bit Microcontrollers**

**CONTENTS**


80C51XA Architectural overview ..... 4-3

XA-G1 CMOS single-chip 16-bit microcontroller ..... 4-13

XA-G2 CMOS single-chip 16-bit microcontroller ..... 4-14

XA-G3 CMOS single-chip 16-bit microcontroller ..... 4-15

XA-C3 CMOS single-chip 16-bit microcontroller with CAN/DeviceNet controller ... 4-16

XA-S3 Single-chip 16-bit microcontroller ..... 4-17

The Philips Semiconductors 80C51XA (eXtended Architecture) family of 16-bit single-chip microcontrollers is powerful enough to easily handle the requirements of high performance embedded applications, yet inexpensive enough to compete in the market for high-volume, low-cost applications.

The XA family provides an upwards migration path for 80C51 users who need higher performance or more than 64k of program memory. Existing 80C51 code can easily be translated to run on XA microcontrollers.

The performance of the XA architecture has not been compromised because of its compatibility with the 80C51. This new architecture supports the comprehensive bit-oriented operations of the 80C51 while incorporating support for multi-tasking operating systems and high-level languages such as C. The speed of the XA architecture, at 10 to 100 times that of the 80C51, gives designers an easy path to truly high-performance embedded control.

Philips Semiconductors will develop a full family of XA derivatives comparable to the existing 80C51 family. Although the XA derivatives will not maintain pin-for-pin compatibility with Philips' 80C51 microcontrollers, the I/O structures will be similar. So, in many cases, system hardware designs may require only minor changes to upgrade to the higher performance XA.

The 80C51XA architecture supports:
- Upward compatibility with the 80C51 architecture
- 16-bit CPU operation
- 24 bit address range (16 megabytes) for both program and data memory
- Eight 16-bit CPU registers, each capable of performing all arithmetic and logic operations
- Both 8-bit and 16-bit operations
- An enhanced instruction set that includes bit intensive logic operations as well as fast 16 × 16 multiply and 32/16 divide
- Multi-tasking and real-time operations systems that include up to 32 vectored interrupts, up to 16 software traps, and banked program memory to support context switching
- Low power operation, which is intrinsic to the XA architecture and will include power-down and idle modes.

## XA CPU ARCHITECTURE

The XA core is partially pipelined and performs some CPU functions in parallel. For instance, instruction fetch and decode, and in some cases write back, are done in parallel with instruction execution. This partial pipelining gives very fast instruction execution at a very low cost. For instance, the instruction execution time for most register-to-register operations on the XA is 3 CPU clocks, or 100 nanoseconds with a 30MHz clock.

Figure 1 presents major functional blocks within the XA core. Specific functional blocks outside of the core will vary from derivative to derivative.

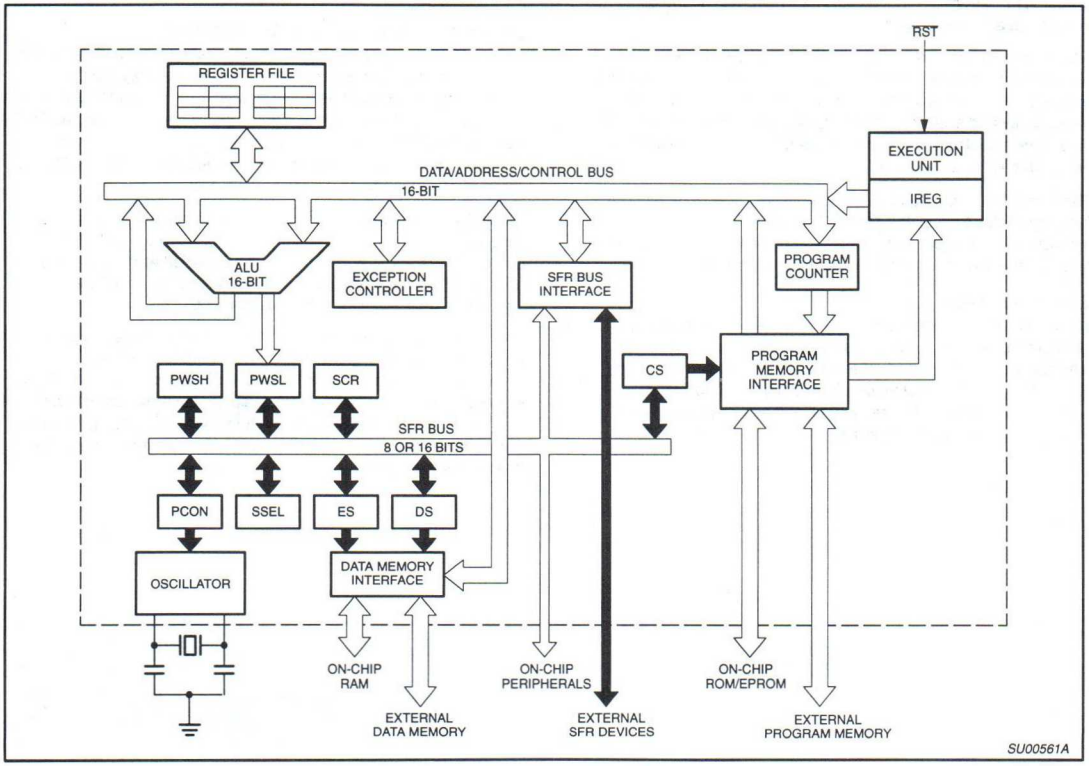



**Figure 1. XA Functional Block Diagram**

## ALU

Data operations in the XA core are accomplished with a 16-bit ALU, providing both 8-bit and 16-bit functions. Special circuitry is added to allow some 32-bit functions, such as shifts, and divides.

## Core Registers

A number of registers comprise a part of the XA core. These registers perform special functions in the XA and are, in fact, accessed by programs as Special Function Registers (SFRs).

The System Configuration Register (SCR) sets up basic operating modes for the XA. The Program Status Word (PSW) contains status flags based on the result of ALU operations. This register contains bits for the four register file banks, the interrupt mask bit, and other system flags. The Data Segment (DS), Extra Segment (ES), and Code Segment (CS) registers contain the segment numbers of active data and code memory segments. The Segment Select register (SSEL), contains bits that determine which segment register is used by each pointer register in the register file. The Power Control register (PCON) holds the bits that control the reduced power modes of the processor (see Power-Down and Idle Modes section).

## Instruction Fetch and Decode

This block controls the fetching of instructions from the code memory and the decoding of instructions prior to execution. The XA normally attempts to fetch instructions from the code memory ahead of what is immediately needed by the execution unit. These pre-fetched instructions are stored in a seven byte queue contained in the fetch and decode unit.

If the fetch unit can keep instructions in the queue, the execution unit will not have to wait for a fetch to occur when it is ready to begin execution of a new instruction. If a program branch is taken, the queue is flushed and instructions are fetched from the new location. This block also decides whether to attempt instruction fetches form on- or off-chip code memory.

The instruction at the head of the queue is decoded into separate functional fields that tell the other CPU blocks what to do when the instruction is executed. These fields are stored in staging registers that hold the information until the next instruction begins executing.

## Execution Unit

The execution unit controls many of the other CPU blocks during instruction execution. It routes addressing information, and sends read and write commands to the register file and memory control blocks. It also tells the fetch and decode unit when to branch, controls the stack, and insures that all of these operations are performed in the proper sequence.

## Interrupt Controller

The interrupt controller receives interrupt requests from all of the sources on a particular XA derivative. It prioritizes these based on user-programmable registers containing a priority for each interrupt source. It then compares the priority of the highest pending interrupt (if any) to the interrupt mask bits from the PSW. If the interrupt has a higher priority than the currently running code, then the interrupt controller issues a request to the execution unit.

The interrupt controller contains an extra register for processing software interrupts. Seven software interrupt request bits, each with an enable bit, can request interrupts entirely under program control. Software interrupts have fixed priority, which is always lower than hardware interrupt priorities. The primary purpose of software interrupts is to allow an interrupt service routine (ISR) to be broken into high and low priority portions and have control transferred between the 2 (or more) portions.

## Exception Controller

The exception controller is similar to the interrupt controller, except that it processes CPU exceptions rather than hardware and software interrupt requests. Sources of exceptions are stack overflow, divide by zero, user execution of an RETI instruction; hardware breakpoint; trace mode; and non-maskable interrupt (NMI).

Exceptions are processed with a fixed priority scheme. Generally they must be serviced immediately since they represent some important event or problem that must be dealt with before normal operation can resume.

## Interrupt and Exception Processing

Interrupt and exception processing both make use of a vector table that resides in the low addresses of the code memory. Each interrupt and exception has a four byte entry in the vector table that includes the two byte starting address of the service routine and a new two byte PSW value to be used at the beginning of service routine. The starting address of a service routine must be within the first 64K of code memory.

When the XA services an exception or interrupt, it first saves the Program Counter address on the stack, followed by the PSW contents. Next, the PC and the PSW are loaded with the starting address of the appropriate service routine and the new PSW contents, respectively, from the vector table.

When the routine completes, it returns to the interrupted code upon execution of the RETI (Return From Interrupt) instruction. This instruction loads first the PSW and then the Program Counter from the stack, resuming operation at the point of interruption. If more than the PC and PSW are used by the service routine, it is up to that routine to save and restore those registers or other portions of the matching state, normally by using the stack.

## PROGRAMMABLE REGISTERS

XA architecture and instruction encoding are optimized for register-based operations, although arithmetic and logical operations may be done directly on a data memory as well. Thus, the XA architecture avoids the bottleneck of having a single accumulator register.

### Register File

The register file allows access to 8 words of data at any one time, which are also addressable as 16 bytes. The bottom 4 word registers are "banked", that is, there are 4 groups of 4 registers, any one of which may occupy the bottom 4 words of the register file at any one time. This feature may be used to minimize the time required for context switching in interrupt services, or to provide more register space for complicated algorithms.

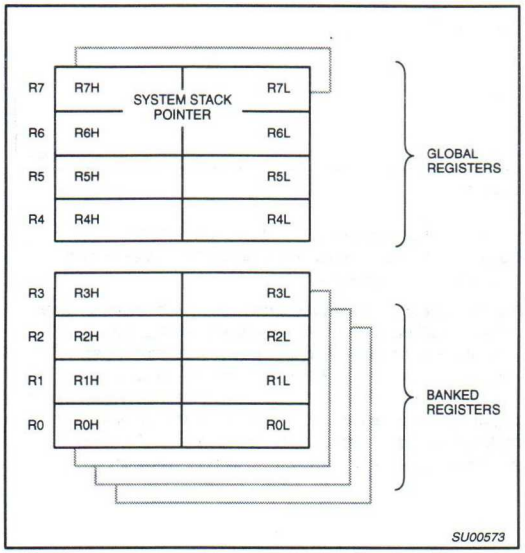



**Figure 2. Register File Structure**

In addition, some instructions (32-bit shifts, multiplies, and divides) allow addressing pairs of word registers as double words. These pairs are always formed by adjacent word registers.

The upper four words of the register file are not banked. The topmost word register acts as the stack pointer, while any other word register may be used as a general purpose pointer to data memory.

The entire register file is bit addressable, that is, any bit in the register file (except the 3 unselected banks of the bottom 4 words) may be operated on by bit manipulation instructions.

The XA instruction encoding allows for future expansion of the register file by the addition of 8 word registers. If implemented, these additional registers will be word data registers only and cannot be used as pointers or addressed as bytes.

The overall XA register file structure provides a superset of the 80C51 register structure. For details, refer to the section on 80C51 compatibility.

### Special Function Registers

Special Function Registers (SFRs) provide a means for the XA to access special purpose CPU registers, peripheral devices, and I/O ports. Any SFR may be accessed by a program at any time without regard to any pointer or segment. An SFR address is always contained entirely within an instruction.

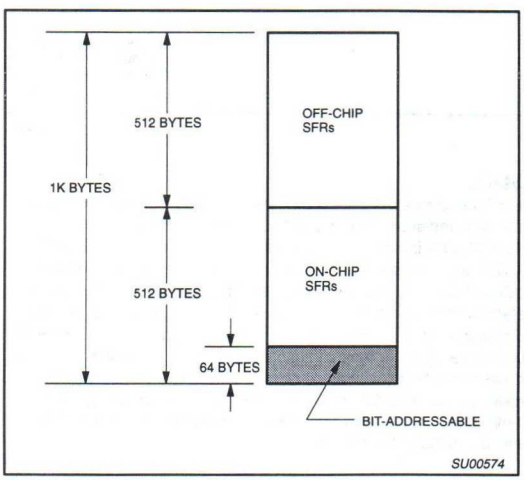



**Figure 3. SFR Addressing**

The total SFR space is 1K bytes in size. This is further divided into two 512 byte regions. The first is for accessing on-chip SFRs. The second allows for future expansion by providing a means to add off-chip I/O devices mapped into the XA as SFRs (Off-chip SFR access is not implemented on all XA derivatives).

The first 64 bytes of on-chip SFR space are bit-addressable. So, any CPU or peripheral register that allows bit addressing will be allocated and addressed within that range.

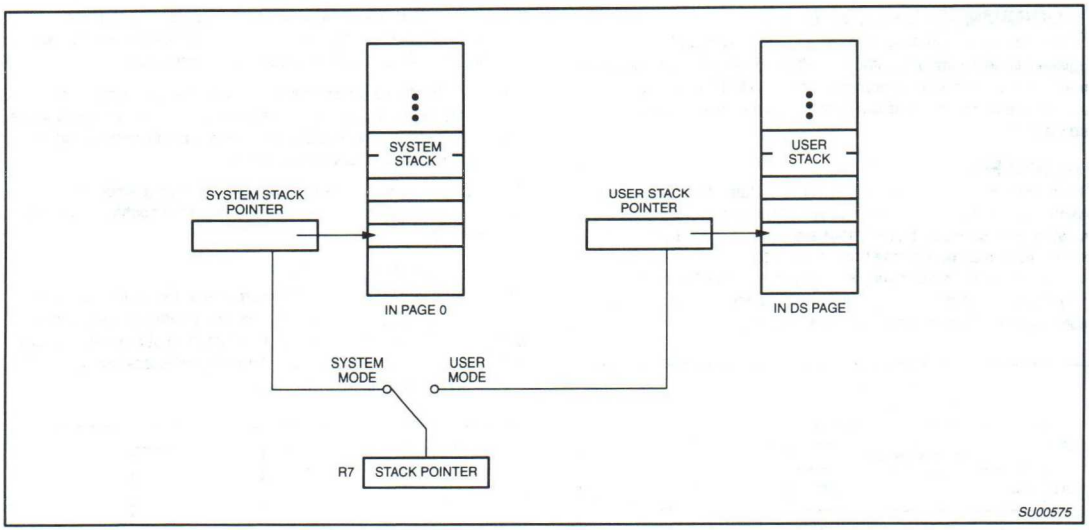



**Figure 4. XA Stacks**

## Stack

The processor stack on any processor provides a means to store interrupt and subroutine return addresses, as well as temporary data. The XA includes 2 stack pointers, the System Stack Pointer (SSP) and the User Stack Pointer (USP), which correspond to 2 different stacks: the system stack and the user stack. The system stack always resides in the first data memory segment, segment 0. The user stack resides in the data memory segment identified by the current value of the data segment (DS) register. Executing code has access to only one of these stacks at a time. Since each stack resides in a single data memory segment, its maximum size is 64K bytes. The purpose of the two stack pointers is discussed in the section on Task Management.

As shown in Figure 4, the System Mode (SM) bit, R7, in the Program Status Word determines whether the processor is in system mode or user mode.

The XA stack grows downwards, from higher addresses to lower addresses within data memory. The current stack pointer always points to the last item pushed on the stack, unless the stack is empty. Prior to a push operation, the stack pointer is decremented by 2, then data is written to memory. When the stack is popped, the reverse procedure is used. First, data is read from memory, then the stack pointer is incremented by 2. Data on the stack always occupies an even number of bytes and is word aligned.

## MEMORY SPACES

The XA register has several discrete memory spaces. Aside from programmable registers, they consist of separate code and data spaces. In both spaces, addressing will automatically roll-over from internal RAM, ROM, or EPROM to external memory when the limit of the memory on the particular derivative is reached.

### Data Memory

The XA architecture can support up to 16 megabytes of data memory space, although some derivative parts may not implement the bus for that entire range. The data space beginning at address 0 is normally on-chip and extends to the limit of the RAM size of a particular XA derivative. Above that, the XA will automatically roll over to external data memory.

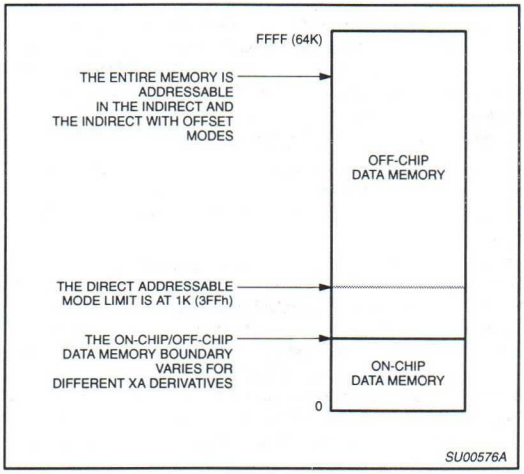



**Figure 5. One 64K Segment of Data Memory Space**

Data memory in the XA is divided into 64K byte segments, in order to provide an intrinsic protection mechanism in multi-tasking systems and to improve performance. Segment registers provide the upper 8 address bits needed to obtain a complete 24-bit address in applications that require large data memories.

The XA provides two segment registers that are used to access data memory, the Data Segment register (DS) and the Extra Segment register (ES). Each pointer register is associated with one of the segment registers via the Segment Select (SSEL) register. A pointer register retains this association until it is changed under program control.

The XA provides direct and indirect addressing modes. See the following section on addressing modes.

The XA data memory addressing scheme provides upward compatibility with the 80C51. For details, refer to the section on 80C51 compatibility.

### Code Memory

The XA is a Harvard architecture controller. This means that the code and data spaces are separate. The XA provides a continuous, unsegmented code space that may be as large as 16 megabytes in size. In parts with on-chip ROM or EPROM code memory, that space begins at code address 0 and extends to the limits of the on-chip code memory. Above that, code will be fetched from off-chip. Most XA derivatives will have an external bus for off-chip data and code access and may also be used in a ROM-less mode, with no code memory used on-chip.

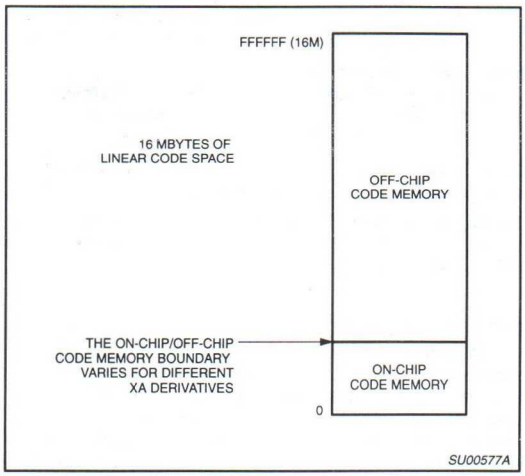



**Figure 6. Code Memory Space**

In some cases, code memory may be addressed as data. Special instructions provide access to the entire code space via pointers. Either a special segment register (CS or Code Segment) or the upper 8 bits of the Program Counter (PC) may be used to identify the portion of code memory referenced by the pointer. Also, immediate data values contained with an instruction are supported for most data manipulation instructions. These immediate values may be 4, 8, or 16-bits in size.

## ADDRESSING MODES

The XA provides flexible data addressing modes. Most arithmetic, logic, and data movement instructions support the following modes of addressing data memory:

**Direct**
The first 1K bytes of data memory in each segment may be accessed directly by the address contained within the instruction.

**Indirect**
A complete 24-bit data memory address is formed by an 8-bit segment register concatenated with 16-bits from a pointer register.

**Indirect with offset**
An 8-bit or 16-bit signed offset contained within the instruction is added to the contents of a pointer register, then concatenated with an 8-bit segment register to produce a complete address. This mode allows access into a data structure when a pointer register contains the starting address of the structure. It also allows subroutines to access parameters passed on the stack.

**Indirect with auto-increment**
The address is formed in the same manner as the indirect addressing mode, but the pointer register contents are automatically incremented following the operation.

Data movement instructions and some special purpose instructions also have additional data addressing modes.

The XA data memory addressing scheme provides upward compatibility with the 80C51. For details, refer to the section on 80C51 compatibility.

## INSTRUCTION SET

The XA instruction set is a superset of the 80C51 instruction set, and is versatile enough to support most common control applications. The instruction encoding is optimized for the most commonly used instructions: register-to-register or register with indirect arithmetic and logic operations; and short conditional and unconditional branches. These instructions are all encoded with 2 bytes. The bulk of the XA instructions are encoded as either 2 or 3 bytes, although there are two 1 byte instructions as well as 4-, 5-, and 6-byte instructions.

The execution of instructions normally overlaps instruction fetch, and sometimes write-back operations, to further speed processing. The number of cycles used by each instruction varies with the type of operation, the location of the operands (on- versus off-chip), and the addressing mode specified.

### Instruction Syntax

The instruction syntax used for the XA is similar in many ways to that used of the 80C51. A typical XA instruction has basic mnemonics, such as "ADD", followed by the operands on which the operation is to be performed. The direction of operation flow is determined by the order in which operands occur in the source line. For instance, the instruction: "ADD R1, R2" would cause the contents of R1 and R2 to be added together and the result stored in R1. Since R1 and R2 are word registers in the XA, this is a 16-bit operation.

An indirect reference (a reference to data memory using the contents of a register as an address) is specified by enclosing the operand in square brackets, as in: "ADD R1, [R2]." This instruction causes the contents of R1 and the data memory location pointed to by R2 (appended to its associated segment register) to be added together and the result stored in R1. Reversing the operand order ("ADD [R2], R1") causes the result to be stored in the data memory location pointed to by R2.

Most instructions support an additional feature called auto-increment that causes the register, used to supply the indirect memory address, to be automatically incremented after the memory access takes place. The source line for such an operation is written as follows: "ADD R1, [R2+]". The auto-increment amount always matches the data size used in the instruction. In the previous example, R2 will have 2 added to it because this was a word operation.

Another version of indirect addressing is called indirect with offset mode. In this version, an immediate value from the instruction word is added to the contents of the indirect register in order to form the actual address. This result of the add is 16 bits in size, which is then concatenated to the segment register for that pointer register. The immediate data from the instruction is a signed 8-bit or 16-bit offset. Thus, the range is +127 bytes to –128 bytes for an 8-bit offset, and +32767 to –32768 bytes for a 16-bit offset. Note that since the address calculation is limited to 16-bits, the 16-bit offset mode allows access to an entire data segment.

When an instruction requires an immediate data value (a value stored within the instruction itself), it is written using the "#" symbol. For example: "ADD R1, #12" says to add the value 12 to register R1.

Since indirect memory references and immediate data values do not implicitly identify the size of the operation to be performed, a few XA instructions must have an operation size explicitly called out. An example would be the instruction: "MOV[R1],#1". The immediate data value does not give a clue to the operation size, and the value stored in memory at the location pointed to by R1 could be either a byte or a word. To clarify the intent of such an instruction, a size identifier is added to the mnemonics as follows: "MOVb [R1],#1". This tells us that the operation should be performed on a byte. If the line read "MOV.w [R1]", it would be a word operation.

If a direct data address is used in an instruction, the address is simply written into the instruction: "ADD 123, R1", meaning to add the contents of register R1 to the data memory value stored at direct address 123. In an actual program, the direct data address could be given a name to make to program more readable such as "ADD Count, R1".

Operations using Special Function Registers (SFRs) are written in a way similar to direct addresses, except that they are normally called out by their names only: "MOV PSW.#12". Using actual SFR addresses rather than their names in instructions makes the code both harder to read and less transportable between XA derivatives.

Bit addresses within instructions may be specified in one of several ways. A bit may be given a unique name, or it may be referred to by its position within some large register or entity. An example of a bit name would be one of the status flags in the PSW, for instance, the carry ("C") flag. To clear the carry flag, the following instruction could be used: "CLR C". The same bit could be addressed by its position within the PSW as follows: "CLR PSWL.7", where the period (".") character indicates that this is a bit reference. A program may use its own name to identify bits that are defined as part of the application program.

Finally, code addresses are written with instructions either by name or by value. Again, a program is more readable and easier to modify if addresses are called out by name. Examples are "JMP Loop"; "JMP 123".

## Instruction Set Summary

**Basic Arithmetic, Logic, and Data Movement Instructions**
For microcontrollers, the most used operations in most programs are likely to be the basic arithmetic and logic instructions, plus the MOV (move data) instruction. The XA supports the following basic operations:

| | |
|---|---|
| ADD | Simple addition |
| ADDC | Add with carry |
| SUB | Subtract |
| SUBB | Subtract with borrow |
| CMP | Compare |
| AND | Logical AND |
| OR | Logical OR |
| XOR | Exclusive OR |

These instructions support all of the following standard XA addressing mode combinations:

| OPERANDS | DESCRIPTION |
|---|---|
| R, R | The source and destination operands are both registers. |
| R, [R] | The source operand is indirect, the destination operand is a register. |
| [R], R | The source operand is a register, the destination operand is indirect. |
| R, [R+] | The source operand is indirect with auto-increment, the destination operand is a register. |
| [R+], R | The source operand is a register, the destination operand is indirect with auto-increment. |
| R, [R+offset] | The source operand is indirect with an 8- or 16-bit offset, the destination operand is a register. |
| [R+offset], R | The source operand is a register, the destination operand is indirect with an 8- or 16-bit offset. |
| direct, R | The source operand is a register, the destination operand is a direct address. |
| R, direct | The source operand is a direct address, the destination operand is a register. |
| R, data | The source operand is an 8- or 16-bit immediate value, the destination operand is a register. |
| [R], data | The source operand is an 8- or 16-bit immediate value, the destination operand is indirect. |
| [R+], data | The source operand is an 8- or 16-bit immediate value, the destination operand is indirect with auto-increment. |
| [R+offset], #data | The source operand is an 8- or 16-bit immediate value, the destination operand is indirect with an 8- or 16-bit offset. |
| direct, #data | The source operand is an 8- or 16-bit immediate value, the destination operand is a direct address. |

Other instructions on the XA use different operand combinations. Following is a summary of other instruction types:

**Additional Arithmetic Instructions**

| | |
|---|---|
| ADDS | Add short immediate (4-bit signed value) |
| NEG | Negate (2's complement) |
| SEXT | Sign extend |
| MUL | Multiply |
| DIV | Divide |
| DA | Decimal adjust |
| ASL | Arithmetic shift left |
| ASR | Arithmetic shift right |
| LEA | Load effective address |

**Additional Logic Instructions**

| | |
|---|---|
| CPL | Complement (1's complement or logical inverse) |
| LSR | Logical shift right |
| NORM | Normalize |
| RL | Rotate left |
| RLC | Rotate left through carry |
| RR | Rotate right |
| RRC | Rotate right through carry |

**Other Data Movement Instructions**

| | |
|---|---|
| MOVS | Move short immediate (4-it signed value) |
| MOVC | Move to or from code memory |
| MOVX | Move to or from external data memory |
| PUSH | Push data onto the stack |
| POP | Pop data from the stack |
| XCH | Exchange data in two locations |

**Bit Manipulation Instructions**

| | |
|---|---|
| SET | Set (write a 1 to) a bit |
| CLR | Clear (write a 0 to) a bit |
| MOV | Move a bit to or from the carry flag |
| ANL | Logical AND a bit (or its inverse) to the carry flag |
| ORL | Logical OR a bit (or its inverse) to the carry flag |

**Jump, Branch, and Call Instructions**

| | |
|---|---|
| BR | Branch to code address (plus or minus 256 byte range) |
| JMP | Jump to code address (range depends on specific JMP variation) |
| CALL | Call subroutine (range depends on specific CALL variation) |
| RET | Return from subroutine or interrupt |
| Bcc | Conditional branches with 15 possible condition variations |
| JB, JNB | Jump if a bit set or not set |
| CJNE | Compare two operands and jump if they are not equal |
| DJNZ | Decrement and jump if the result is not zero |
| JZ, JNZ | Jump on zero or not zero (included for 80C51 compatibility |

**Other Instructions**

| | |
|---|---|
| NOP | No operation (used mainly to align branch targets) |
| BKPT | Breakpoint (used for debugging) |
| TRAP | Software trap (used to call system services in a multi-tasking system) |
| RESET | Reset the entire chip |

## RESET OPERATIONS

Power up reset and any other external reset of the XA is accomplished via an active low reset pin. A simple resistor and capacitor reset circuit is typically used to provide the power-on reset pulse. The reset pin is a Schmitt trigger, in order to prevent noise on the reset pin from causing spurious or incomplete resets.

The XA may be reset under program control by executing the RESET instruction. This instruction has the effect of resetting the processor as if an external reset occurred, except that some hardware features that are latched following a hardware reset (such as the state to the EA pin and bus size programming) are not re-latched by a software reset.

Some XA derivatives also have a hardware watchdog timer peripheral that will cause a chip reset if the program gets out of control or locks up for any reason.

## OSCILLATOR SUPPORT

XA derivatives have an on-chip oscillator that may be used with crystals or ceramic resonators, or an external clock to provide a clock source for the processor.

## POWER-DOWN AND STANDBY MODES

The XA supports two power saving modes: Idle and Power-Down. Either mode is activated by setting a bit in the Power Control (PCON) register. The Idle mode shuts down all processor functions, but leaves most of the on-chip peripherals and the external interrupts functioning. An interrupt from any operating source will cause the XA to resume operation where it left off.

The Power-Down mode goes one step further and shuts down everything including the on-chip oscillator. This reduces power consumption to a tiny amount of CMOS leakage plus whatever loads are placed on chip pins. Resuming operation from the Power-Down mode requires the oscillator to be restarted, which takes about 10 milliseconds. Power-Down mode can be terminated either by resetting the XA or by asserting one of the external interrupts, if one was left enabled when Power-Down mode was entered.

## EXTERNAL BUS

Most XA derivatives have the capability of accessing external code and/or data memory through the use of an external bus. The external bus provides address information to external devices, and initializes code read, data read, or data write strobes. The standard XA external bus is designed to provide flexibility, simplicity of connection, and optimization for external code fetches.

### External Bus Signals

The standard XA external bus supports 8- or 16-bit transfers and up to 24 address lines. The precise number of address lines varies by derivative. The standard control signals and their functions for the external bus are as follows:

| SIGNAL NAME | FUNCTION |
|---|---|
| ALE | Address Latch Enable. This signal directs an external address latch to store a portion of the address for the next bus operation. This may be a data address or a code address. |
| $\overline{\text{PSEN}}$ | Program Store Enable. Indicates that the XA is reading code information over the bus. Typically connected to the Output. This connects to the enable pin of external EPROMs. |
| $\overline{\text{RD}}$ | Read. The external data read strobe. Typically connected to the $\overline{\text{RD}}$ pin of external peripheral devices. |
| $\overline{\text{WR}}$ | Write. The external data write strobe. Typically connected to the $\overline{\text{WR}}$ pin of external peripheral devices. For a 16-bit data bus, this strobe applies only to the lower data byte. |
| $\overline{\text{WR1}}$ | Write 1. Upper byte write strobe for external data when using a 16-bit data bus. |
| WAIT | Wait. Allows slowing down any type of external bus cycle. When asserted during a bus operation, that operation waits for this signal to be de-asserted before it is completed. |

### Bus Configurations

The standard XA bus is user-configurable in several ways. First the bus size may be configured to either 8 bits or 16 bits. This may be configured by the logic level on a pin at rest, or under firmware control (if code is initially executed from on-chip code memory) prior to any actual external bus operations. As on the 80C51, the $\overline{\text{EA}}$ pin determines whether or not on-chip code memory is used for initial code fetches.

Second, the number of address lines may be configured in order to make optimal use of I/O ports. Since external bus functions are typically shared with I/O ports and/or peripheral I/O functions, it is advantageous to set the number of address lines to only what is needed for a particular application, freeing I/O pins for other uses.

### Bus Timing

The standard XA bus also provides a high degree of bus timing configurability. There are separate controls for ALE width, $\overline{\text{PSEN}}$ width, $\overline{\text{RD}}$ and $\overline{\text{WR}}$ width, and data hold time from $\overline{\text{WR}}$. These times are programmable in a range that will support most RAMs, ROMs, EPROMs and peripheral devices over a wide range of oscillator frequencies without the need for additional external latches, buffers, or WAIT state generators.

## I/O PORTS

Standard I/O ports on the XA have been enhanced to provide better versatility and programmability than was previously available in the 80C51 and most of its derivatives. Access to the I/O ports from a program is through SFR addresses assigned to those ports. Ports may be read and written in the same manner as any other SFR.

The XA provides more flexibility in the use of I/O ports by allowing different output configurations. Port outputs may be programmed to be quasi-bidirectional (80C51 style ports), open drain, push-pull, and high impedance (input only).

## PERIPHERALS

The XA CPU core is designed to make derivative design fast and easy. Peripheral devices are not part of the core, but are attached by means of a special peripheral bus, called the P-bus, which is separate from the CPU internal buses. So, a new XA derivative may be made by designing a new P-bus-compatible peripheral function block, if one does not already exist, then attaching it to the XA core.

## MULTI-TASKING

Several features of the XA have been designed to support multi-tasking. Multi-tasking can be thought of as running several programs at once on the same processor, with a supervisory program determining when each program, or task, runs, and for how long. Since each task shares the same CPU, the system resources required by each must be kept separate and the CPU state restored when switching execution from one task to another. The problem is much simpler for a microcontroller than it is for a microprocessor, because the code executed by a microcontroller always comes from the same source: the designers of the system on which it runs. Thus, this code can be considered to be basically trustworthy and extreme measures to prevent misbehavior are not necessary.

The first step in supporting multi-tasking is to provide two basic modes of operation, one for the basic tasks and one for the supervisory program. On the XA these are called user mode and system mode, respectively. A supervisory program running in system mode has access to all of the processor's resources and can set up and launch tasks.

Code running in system and user mode uses different stack pointers, the System Stack Pointer (SSP) and the User Stack Pointer (USP), respectively. The system stack is always located in the first 64K data memory segment, where it can take advantage of the fast on-chip RAM. The user stack is located within each task's local data segment, identified by the DS register. The fact that user mode code uses a different stack than system mode code prevents tasks from accidentally destroying data on the system stack.

Additional protection mechanisms are provided in the form of control bits and registers that are only writeable by system mode code. For instance, the DS register that identifies the local data segment for user mode code is only writeable in the system mode. While tasks can still write to the other segment register, the ES register, they cannot write to memory via the ES register unless specifically allowed to do so by the system. The data memory segmentation scheme thus prevents tasks from accessing data memory in unpredictable ways.

Other protected features include enabling of the Trace Mode, alteration of the Interrupt Mask, or changing the selected register bank (unless granted permission by the system).

The 4 register banks are a feature that can be useful in a small multi-tasking system by using each bank for a different task, including one for system code. This means fewer CPU states that must be saved during task switching.

## DEBUGGING FEATURES

The XA incorporates some special features designed to aid in program and system debugging. There is a software breakpoint instruction that may be inserted in a user's program by a debugger program, causing the user program to break at that point and go to the breakpoint service routine, which can transmit the CPU state so that it can be viewed by the user.

The trace mode is similar to a breakpoint, but is forced by hardware in the XA after the execution of every instruction. The trace service routine can then keep track of every instruction executed by a user program and transmit information about the CPU state to a serial port or other peripheral for display or storage. Trace mode is controlled by a bit in the PSW. The XA is able to alter the trace mode bit whenever an interrupt or exception vector is taken. This gives very flexible use of trace mode; for instance, by allowing all interrupts to run at full speed to comply with system hardware requirements, while single stepping through mainline code.

With these two features, a simple debugger routine can allow a user to single step through a program, or to run a program at full speed, stopping only when execution reaches a breakpoint, in either case viewing the CPU state before continuing.

## 80C51 COMPATIBILITY

The 80C51 is the most designed-in 8-bit microcontroller architecture in the world, and a vast amount of public and private code exists for this processor. For customers who have been using the 80C51 or one of its derivatives, preservation of some of their investment in code development is an important consideration. By permitting simple translation of source code, the XA allows existing 80C51 code to be re-used with the high performance 16-bit XA microcontroller.

Many trade-offs and considerations were taken into account in the creation of the XA architecture. The most important goal was that it would be possible for a software translator to take 80C51 assembler source code and automatically produce XA source code. The issues in making that possible fall under the categories of software and hardware, which are covered briefly in the following discussion.

### Software Compatibility

Several basic goals were observed in order to insure 80C51 compatibility with the XA, yet preserve a straightforward design. First, each 80C51 instruction translates into one XA instruction. Multi-instruction combinations that could result in problems if split by an interrupt were avoided as much as possible. In fact, only one rarely used 80C51 instruction cannot be replaced by a single XA instruction. Second, most 80C51 instructions should be a subset of more powerful XA instructions. If that is not possible, the original 80C51 instruction would be included "as-is".

Instruction timing is one item that must change if any increased processing speed is added over the 80C51. The XA does not attempt to retain timing compatibility with the 80C51, but rather provides the fastest execution that is feasible at low cost. When 80C51 code that is timing critical is translated to the XA, the user must re-analyze the timing and make adjustments.

## Hardware Compatibility

A major consideration in the hardware compatibility of the XA with 80C51 is the memory map. The XA approaches this issue by having each memory area (registers, data memory, code memory, stack, SFRs) be a superset of the 80C51. One area where some difference could not be avoided is in the use of the processor stack. Due to the fact that the XA supports 16-bit operations in memory, it was necessary to change the direction of stack growth to downward (which is standard for 16-bit processors), in order to match stack usage with the storage of 16-bit variables in memory. This is an important consideration for the support of high-level language compilers such as C. For most 80C51 code the changes to the XA stack will be transparent (have no effect) for translated code.

Translation of SFR accesses is usually simple, since SFRs are normally referenced by name. Such references are simply retained in the translated XA code. If a translated 80C51 program references an SFR by its address, the translator can try to relate that to an SFR name if it is given an SFR table for the 80C51 derivative on which the code was intended to run.

XA derivatives are not necessarily intended to be pin compatible with other 80C51 derivatives that have similar features. For instance, on-chip peripherals may be altered on the XA to provide more capabilities. Initially, however, most peripherals have been made upward compatible with the original 80C51 peripheral, and most enhancements are added transparently in such a way that they will not interfere with 80C51 code that does not use them.

## XA USER'S GUIDE

This document is an overview of the XA architecture. For a detailed discussion of the XA architecture please request a copy of the *XA User's Guide* from your local Philips sales office or sales representative.

# CMOS single-chip 16-bit microcontroller XA-G1

*This excerpt represents a short-form datasheet only.*
*For full datasheet, refer to* ***Data Handbook IC25: 16-bit 80C51XA Microcontrollers (eXtended Architecture).***

## FAMILY DESCRIPTION

The Philips Semiconductors XA (eXtended Architecture) family of 16-bit single-chip microcontrollers is powerful enough to easily handle the requirements of high performance embedded applications, yet inexpensive enough to compete in the market for high-volume, low-cost applications.

The XA family provides an upward compatibility path for 80C51 users who need higher performance and 64k or more of program memory. Existing 80C51 code can also easily be translated to run on XA microcontrollers.

The performance of the XA architecture supports the comprehensive bit-oriented operations of the 80C51 while incorporating support for multi-tasking operating systems and high-level languages such as C. The speed of the XA architecture, at 10 to 100 times that of the 80C51, gives designers an easy path to truly high performance embedded control.

The XA architecture supports:

- Upward compatibility with the 80C51 architecture
- 16-bit fully static CPU with a 24-bit program and data address range
- Eight 16-bit CPU registers each capable of performing all arithmetic and logic operations as well as acting as memory pointers. Operations may also be performed directly to memory.
- Both 8-bit and 16-bit CPU registers, each capable of performing all arithmetic and logic operations.
- An enhanced instruction set that includes bit intensive logic operations and fast signed or unsigned 16 × 16 multiply and 32 / 16 divide
- Instruction set tailored for high level language support
- Multi-tasking and real-time executives that include up to 32 vectored interrupts, 16 software traps, segmented data memory, and banked registers to support context switching
- Low power operation, which is intrinsic to the XA architecture, includes power-down and idle modes.

More detailed information on the core is available in the XA User Guide.

## SPECIFIC FEATURES OF THE XA-G1

- 20-bit address range, 1 megabyte each program and data space. (Note that the XA architecture supports up to 24 bit addresses.)
- 3.0V to 5.5V operation
- 8K bytes on-chip EPROM/ROM program memory
- 512 bytes of on-chip data RAM
- Three counter/timers with enhanced features (equivalent to 80C51 T0, T1, and T2)
- Watchdog timer
- Two enhanced UARTs
- Four 8-bit I/O ports with 4 programmable output configurations
- 44-pin PLCC and 44-pin LQFP packages

## ORDERING INFORMATION

| ROM | EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQ (MHz) | DRAWING NUMBER |
|---|---|---|---|---|---|
| P51XAG13GB BD | P51XAG17GB BD | OTP | 0 to +70, Plastic Low Profile Quad Flat Pkg. | 20 | SOT389–1 |
| P51XAG13GB A | P51XAG17GB A | OTP | 0 to +70, Plastic Leaded Chip Carrier | 20 | SOT187–2 |
| | P51XAG17GB KA | UV | 0 to +70, Ceramic Leaded Chip Carrier | 20 | 1472A |
| P51XAG13GF BD | P51XAG17GF BD | OTP | –40 to +85, Plastic Low Profile Quad Flat Pkg. | 20 | SOT389–1 |
| P51XAG13GF A | P51XAG17GF A | OTP | –40 to +85, Plastic Leaded Chip Carrier | 20 | SOT187–2 |
| | P51XAG17GF KA | UV | –40 to +85, Ceramic Leaded Chip Carrier | 20 | 1472A |
| P51XAG13JB BD | P51XAG17JB BD | OTP | 0 to +70, Plastic Low Profile Quad Flat Pkg. | 25 | SOT389–1 |
| P51XAG13JB A | P51XAG17JB A | OTP | 0 to +70, Plastic Leaded Chip Carrier | 25 | SOT187–2 |
| | P51XAG17JB KA | UV | 0 to +70, Ceramic Leaded Chip Carrier | 25 | 1472A |
| P51XAG13JF BD | P51XAG17JF BD | OTP | –40 to +85, Plastic Low Profile Quad Flat Pkg. | 25 | SOT389–1 |
| P51XAG13JF A | P51XAG17JF A | OTP | –40 to +85, Plastic Leaded Chip Carrier | 25 | SOT187–2 |
| | P51XAG17JF KA | UV | –40 to +85, Ceramic Leaded Chip Carrier | 25 | 1472A |
| P51XAG13KB BD | P51XAG17KB BD | OTP | 0 to +70, Plastic Low Profile Quad Flat Pkg. | 30 | SOT389–1 |
| P51XAG13KB A | P51XAG17KB A | OTP | 0 to +70, Plastic Leaded Chip Carrier | 30 | SOT187–2 |
| | P51XAG17KB KA | UV | 0 to +70, Ceramic Leaded Chip Carrier | 30 | 1472A |

**NOTE:**
1. OTP = One Time Programmable EPROM. UV = Erasable EPROM.

# CMOS single-chip 16-bit microcontroller XA-G2

*This excerpt represents a short-form datasheet only.*
*For full datasheet, refer to* ***Data Handbook IC25: 16-bit 80C51XA Microcontrollers (eXtended Architecture).***

## FAMILY DESCRIPTION

The Philips Semiconductors XA (eXtended Architecture) family of 16-bit single-chip microcontrollers is powerful enough to easily handle the requirements of high performance embedded applications, yet inexpensive enough to compete in the market for high-volume, low-cost applications.

The XA family provides an upward compatibility path for 80C51 users who need higher performance and 64k or more of program memory. Existing 80C51 code can also easily be translated to run on XA microcontrollers.

The performance of the XA architecture supports the comprehensive bit-oriented operations of the 80C51 while incorporating support for multi-tasking operating systems and high-level languages such as C. The speed of the XA architecture, at 10 to 100 times that of the 80C51, gives designers an easy path to truly high performance embedded control.

The XA architecture supports:

- Upward compatibility with the 80C51 architecture
- 16-bit fully static CPU with a 24-bit program and data address range
- Eight 16-bit CPU registers each capable of performing all arithmetic and logic operations as well as acting as memory pointers. Operations may also be performed directly to memory.
- Both 8-bit and 16-bit CPU registers, each capable of performing all arithmetic and logic operations.
- An enhanced instruction set that includes bit intensive logic operations and fast signed or unsigned 16 × 16 multiply and 32 / 16 divide
- Instruction set tailored for high level language support
- Multi-tasking and real-time executives that include up to 32 vectored interrupts, up to 16 software traps, segmented data memory, and banked registers to support context switching
- Low power operation, which is intrinsic to the XA architecture includes power-down and idle modes.

More detailed information on the core is available in the XA User Guide.

## SPECIFIC FEATURES OF THE XA-G2

- 20-bit address range, 1 megabyte each program and data space. (Note that the XA architecture supports up to 24 bit addresses.)
- 2.7V to 5.5V operation
- 16K bytes on-chip EPROM/ROM program memory
- 512 bytes of on-chip data RAM
- Three counter/timers with enhanced features (equivalent to 80C51 T0, T1, and T2)
- Watchdog timer
- Two enhanced UARTs
- Four 8-bit I/O ports with 4 programmable output configurations
- 44-pin PLCC and 44-pin LQFP packages

## ORDERING INFORMATION

| ROM | EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQ (MHz) | DRAWING NUMBER |
|---|---|---|---|---|---|
| P51XAG23GB BD | P51XAG27GB BD | OTP | 0 to +70, Plastic Low Profile Quad Flat Pkg. | 20 | SOT389–1 |
| P51XAG23GB A | P51XAG27GB A | OTP | 0 to +70, Plastic Leaded Chip Carrier | 20 | SOT187–2 |
| | P51XAG27GB KA | UV | 0 to +70, Ceramic Leaded Chip Carrier | 20 | 1472A |
| P51XAG23GF BD | P51XAG27GF BD | OTP | –40 to +85, Plastic Low Profile Quad Flat Pkg. | 20 | SOT389–1 |
| P51XAG23GF A | P51XAG27GF A | OTP | –40 to +85, Plastic Leaded Chip Carrier | 20 | SOT187–2 |
| | P51XAG27GF KA | UV | –40 to +85, Ceramic Leaded Chip Carrier | 20 | 1472A |
| P51XAG23JB BD | P51XAG27JB BD | OTP | 0 to +70, Plastic Low Profile Quad Flat Pkg. | 25 | SOT389–1 |
| P51XAG23JB A | P51XAG27JB A | OTP | 0 to +70, Plastic Leaded Chip Carrier | 25 | SOT187–2 |
| | P51XAG27JB KA | UV | 0 to +70, Ceramic Leaded Chip Carrier | 25 | 1472A |
| P51XAG23JF BD | P51XAG27JF BD | OTP | –40 to +85, Plastic Low Profile Quad Flat Pkg. | 25 | SOT389–1 |
| P51XAG23JF A | P51XAG27JF A | OTP | –40 to +85, Plastic Leaded Chip Carrier | 25 | SOT187–2 |
| | P51XAG27JF KA | UV | –40 to +85, Ceramic Leaded Chip Carrier | 25 | 1472A |
| P51XAG23KB BD | P51XAG27KB BD | OTP | 0 to +70, Plastic Low Profile Quad Flat Pkg. | 30 | SOT389–1 |
| P51XAG23KB A | P51XAG27KB A | OTP | 0 to +70, Plastic Leaded Chip Carrier | 30 | SOT187–2 |
| | P51XAG27KB KA | UV | 0 to +70, Ceramic Leaded Chip Carrier | 30 | 1472A |

**NOTE:**
1. OTP = One Time Programmable EPROM. UV = Erasable EPROM.

# CMOS single-chip 16-bit microcontroller XA-G3

*This excerpt represents a short-form datasheet only.*
*For full datasheet, refer to* ***Data Handbook IC25: 16-bit 80C51XA Microcontrollers (eXtended Architecture).***

## FAMILY DESCRIPTION

The Philips Semiconductors XA (eXtended Architecture) family of 16-bit single-chip microcontrollers is powerful enough to easily handle the requirements of high performance embedded applications, yet inexpensive enough to compete in the market for high-volume, low-cost applications.

The XA family provides an upward compatibility path for 80C51 users who need higher performance and 64k or more of program memory. Existing 80C51 code can also easily be translated to run on XA microcontrollers.

The performance of the XA architecture supports the comprehensive bit-oriented operations of the 80C51 while incorporating support for multi-tasking operating systems and high-level languages such as C. The speed of the XA architecture, at 10 to 100 times that of the 80C51, gives designers an easy path to truly high performance embedded control.

The XA architecture supports:

- Upward compatibility with the 80C51 architecture
- 16-bit fully static CPU with a 24-bit program and data address range
- Eight 16-bit CPU registers each capable of performing all arithmetic and logic operations as well as acting as memory pointers. Operations may also be performed directly to memory.
- Both 8-bit and 16-bit CPU registers, each capable of performing all arithmetic and logic operations.
- An enhanced instruction set that includes bit intensive logic operations and fast signed or unsigned 16 × 16 multiply and 32 / 16 divide
- Instruction set tailored for high level language support
- Multi-tasking and real-time executives that include up to 32 vectored interrupts, 16 software traps, segmented data memory, and banked registers to support context switching
- Low power operation, which is intrinsic to the XA architecture, includes power-down and idle modes.

More detailed information on the core is available in the XA User Guide.

## SPECIFIC FEATURES OF THE XA-G3

- 20-bit address range, 1 megabyte each program and data space. (Note that the XA architecture supports up to 24 bit addresses.)
- 3.0V to 5.5V operation
- 32K bytes on-chip EPROM/ROM program memory
- 512 bytes of on-chip data RAM
- Three counter/timers with enhanced features (equivalent to 80C51 T0, T1, and T2)
- Watchdog timer
- Two enhanced UARTs
- Four 8-bit I/O ports with 4 programmable output configurations
- 44-pin PLCC and 44-pin LQFP packages

## ORDERING INFORMATION

| ROMless | ROM | EPROM[1] | | TEMPERATURE RANGE °C AND PACKAGE | FREQ (MHz) | DRAWING NUMBER |
|---|---|---|---|---|---|---|
| P51XAG30GB BD | P51XAG33GB BD | P51XAG37GB BD | OTP | 0 to +70, Plastic Low Profile Quad Flat Pkg. | 20 | SOT389–1 |
| P51XAG30GB A | P51XAG33GB A | P51XAG37GB A | OTP | 0 to +70, Plastic Leaded Chip Carrier | 20 | SOT187–2 |
| | | P51XAG37GB KA | UV | 0 to +70, Ceramic Leaded Chip Carrier | 20 | 1472A |
| P51XAG30GF BD | P51XAG33GF BD | P51XAG37GF BD | OTP | –40 to +85, Plastic Low Profile Quad Flat Pkg. | 20 | SOT389–1 |
| P51XAG30GF A | P51XAG33GF A | P51XAG37GF A | OTP | –40 to +85, Plastic Leaded Chip Carrier | 20 | SOT187–2 |
| | | P51XAG37GF KA | UV | –40 to +85, Ceramic Leaded Chip Carrier | 20 | 1472A |
| P51XAG30JB BD | P51XAG33JB BD | P51XAG37JB BD | OTP | 0 to +70, Plastic Low Profile Quad Flat Pkg. | 25 | SOT389–1 |
| P51XAG30JB A | P51XAG33JB A | P51XAG37JB A | OTP | 0 to +70, Plastic Leaded Chip Carrier | 25 | SOT187–2 |
| | | P51XAG37JB KA | UV | 0 to +70, Ceramic Leaded Chip Carrier | 25 | 1472A |
| P51XAG30JF BD | P51XAG33JF BD | P51XAG37JF BD | OTP | –40 to +85, Plastic Low Profile Quad Flat Pkg. | 25 | SOT389–1 |
| P51XAG30JF A | P51XAG33JF A | P51XAG37JF A | OTP | –40 to +85, Plastic Leaded Chip Carrier | 25 | SOT187–2 |
| | | P51XAG37JF KA | UV | –40 to +85, Ceramic Leaded Chip Carrier | 25 | 1472A |
| P51XAG30KB BD | P51XAG33KB BD | P51XAG37KB BD | OTP | 0 to +70, Plastic Low Profile Quad Flat Pkg. | 30 | SOT389–1 |
| P51XAG30KB A | P51XAG33KB A | P51XAG37KB A | OTP | 0 to +70, Plastic Leaded Chip Carrier | 30 | SOT187–2 |
| | | P51XAG37KB KA | UV | 0 to +70, Ceramic Leaded Chip Carrier | 30 | 1472A |

**NOTE:**
1. OTP = One Time Programmable EPROM. UV = Erasable EPROM.

# CMOS single-chip 16-bit microcontroller with CAN/DeviceNet controller

# XA-C3

## DESCRIPTION

The XA-CAN device is a member of Philips' 80C51 XA (eXtended Architecture) family of high performance 16-bit single-chip microcontrollers, and is intended for industrial control applications.

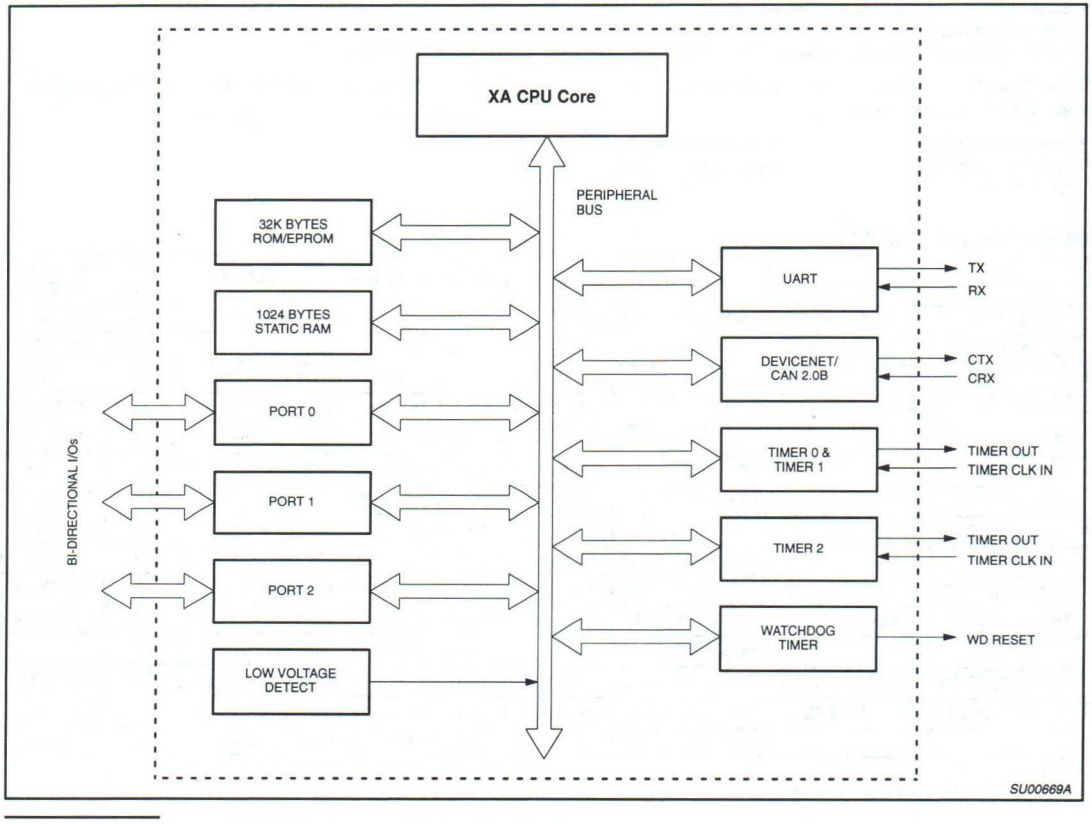
The XA-CAN device supports the DeviceNet™ /CAN Controller Area Network (CAN) 2.0B.. It supports both 11-bit and 29-bit identifiers (ID) at up to 1Mbit/s data rate.

The performance of the XA architecture supports the comprehensive bit-oriented operations of the 80C51 while incorporating support for multi-tasking operating systems and high-level languages such as C. The speed of the XA architecture, at 10 to 100 times that of the 80C51, gives designers an easy path to truly high performance embedded control, while maintaining great flexibility to adapt software to specific requirements.

## Specific Features of the XA-C3

- 2.7V to 5.5V operation
- 32K bytes of on-chip EPROM/ROM program memory
- 1024 bytes of on-chip data RAM
- CAN block supporting full CAN2.0B, with 11-/29-bit ID and up to 1Mbit/s
- Three standard counter/timers with enhanced features (equivalent to 80C51 T0, T1, and T2) with outputs
- Watchdog timer with output
- 1 UART
- Low voltage detect
- Three 8-bit I/O ports with 4 programmable output configurations
- EPROM/OTP versions can be programmed in circuit
- 25MHz operating frequency at 4.5 – 5.5V $V_{CC}$ over commercial operating conditions; 16MHz at 2.7V – 3.6V $V_{CC}$
- 40-pin DIP, 44-pin PLCC, and 44-pin QFP packages

## BLOCK DIAGRAM




DeviceNet™ is a trademark of Open DeviceNet Vendor Association (OVDA).

# Single-chip 16-bit microcontroller XA-S3

## DESCRIPTION

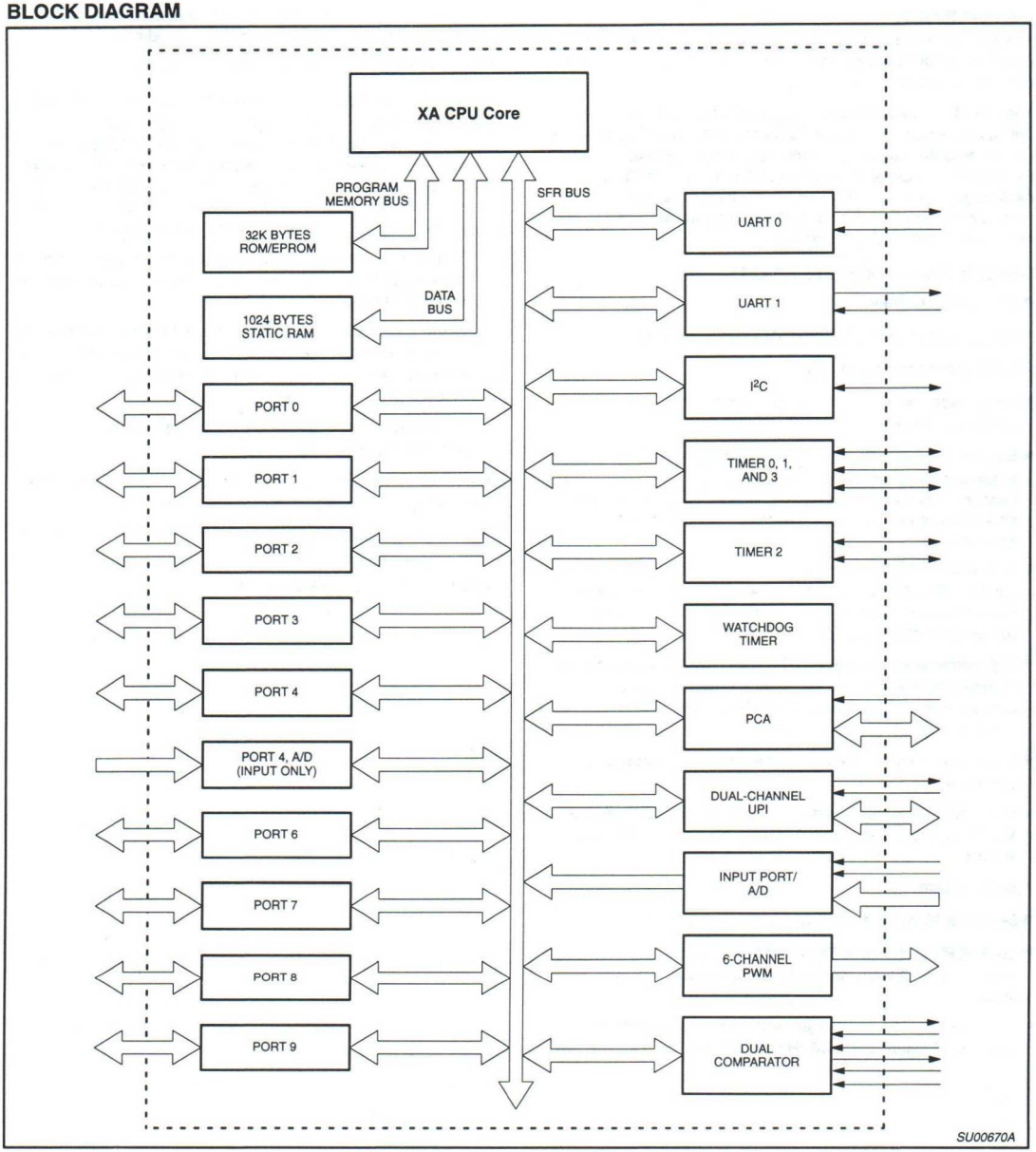
The XA-S3 device is a member of Philips' 80C51 XA (eXtended Architecture) family of high performance 16-bit single-chip general purpose microcontrollers.

The XA-S3 device combines many powerful peripherals on chip. With its dual-channel Universal Peripheral Interface (UPI), high performance A/D converters, timers/counters, watchdog, Programmable Counter Array (PCA), $I^2C$ interface, UARTs and multiple general purpose I/O ports, it is suited for general multipurpose high performance embedded control functions, PC peripheral control, and motor control.

## Specific Features of the XA-S3

- 2.7V to 5.5V operation
- 32K bytes of on-chip EPROM/ROM program memory
- 1024 bytes of on-chip data RAM
- Supports off-chip addressing up to 16 megabytes (24 address lines)
- Support for off-chip Special Function Registers. The off-chip SFR space is mapped into the normal external data memory space at locations 0 through 1FF hex. An optional SFR bus strobe allows distinguishing an external SFR access from other types of bus functions.
- Dual-channel 8-bit Universal Peripheral Interface (UPI). Similar to 8XC576 UPI, with separate IBF and OBF interrupts. Incorporates two data registers. A single byte control register contains the status flags for both UPI ports
- High performance 8-channel 10-bit A/D converter with automatic channel scan and repeated read functions. Completes a conversion in 5 microseconds at 20 MHz (100 clocks per conversion). Operates down to 3V.
- Two analog comparators with separate reference inputs and output pins.
- Four standard counter/timers with enhanced features (same as XA-G3 T0, T1, and T2, plus a T3 which is identical to T1). All timers have a toggle output capability.
- Watchdog timer.
- 5-channel 16-bit Programmable Counter Array (PCA).
- Six 8-bit PWM channels. The PWM frequency range is from 153.8 Hz to 39.2 kHz when an oscillator frequency of 20 MHz is used.
- $I^2C$-bus serial I/O port with byte-oriented master and slave functions. Supports both 100 kHz and 400 kHz $I^2C$ operating modes. Also has a "bus monitor" mode that turns off automatic address recognition and interrupts for any address.
- Two enhanced UARTs with independent baud rates
- Nine 8-bit I/O ports (100-pin package) or seven 8-bit I/O ports (80-pin package), each with 4 programmable output configurations. One additional 8-bit input-only port shared with analog inputs. One port will generate an interrupt for a low level on any of 8 pins. This may be used as a keyboard/keypad sense port that can wake-up the S3 from power-down when a key is pressed. One vector is used for all 8 interrupt sources.
- Interrupt system supports 26 distinct event interrupts associated with various peripheral functions. In addition, seven software interrupts are supported.
- Active low reset pin also acts as an open drain output to indicate internal reset occurrences (watchdog resets and the RESET instruction). A reset source register allows program determination of the cause of the most recent reset.
- EPROM/OTP versions can be programmed in circuit (On-Board Programming).
- 25MHz operating frequency at 4.5 – 5.5V $V_{DD}$ over commercial operating conditions; 16MHz at 2.7V – 3.6V $V_{DD}$
- Power saving operating modes: Idle and Power-Down. Wake-Up from power-down is supported.
- 80-pin and 100-pin QFP packages.

## BLOCK DIAGRAM



# Section 5
## Package Outlines

## 80C51-Based
## 8-Bit Microcontrollers

CONTENTS


Soldering — Package information ..... 5-2

**Plastic Dual In-Line Package**

DIP8: plastic dual in-line package; 8 leads (300 mil) ..... SOT97-1 .... 5-4
DIP24: plastic dual in-line package; 24 leads (300 mil) ..... SOT222-1 ... 5-5
DIP28: plastic dual in-line package; 28 leads (600 mil); long body ..... SOT117-2 .... 5-6
DIP40: plastic dual in-line package; 40 leads (600 mil) ..... SOT129-1 ... 5-7

**Plastic Shrink Dual In-Line Package**

SDIP42: plastic shrink dual in-line package; 42 leads (600 mil) ..... SOT270-1 ... 5-8
SDIP64: plastic shrink dual in-line package; 64 leads (750 mil) ..... SOT274-1 ... 5-9

**Ceramic Dual In-Line Package**

24-Pin (300 mils wide) Ceramic Dual In-line (F) Package (with Window (FA) Package) ..... 0586B ..... 5-10
28-Pin (600 mils wide) Ceramic Dual In-line (F) Package (with Window (FA) Package) ..... 0589B ..... 5-11
40-Pin (600 mils wide) Ceramic Dual In-line (F) Package (with Window (FA) Package) ..... 0590B ..... 5-12

**Plastic Leaded Chip Carrier**

PLCC28: plastic leaded chip carrer; 28 leads; pedestal ..... SOT261-3 ... 5-13
PLCC44: plastic leaded chip carrier; 44 leads ..... SOT187-2 ... 5-14
PLCC68: plastic leaded chip carrier; 68 leads ..... SOT188-2 ... 5-15
PLCC68: plastic leaded chip carrier; 68 leads; pedestal ..... SOT188-3 ... 5-16

**Ceramic Leaded Chip Carrier**

68-Pin Chip Carrier, J-Bend (L) Package ..... 1240C ..... 5-17
Ceramic leaded chip carrier (window); 68 leads ..... NO330 ..... 5-18

**Plastic Quad Flat Package**

QFP44: plastic quad flat package; 44 leads (lead length 1.3 mm); body 10 x 10 x 1.75 mm ..... SOT307-2 ... 5-19
QFP44: plastic quad flat package; 44 leads (lead length 2.35 mm); body 14 x 14 x 2.2 mm ..... SOT205-1 ... 5-20
LQFP44: plastic low profile quad flat package; 44 leads; body 10 x 10 x 1.4 mm ..... SOT389-1 ... 5-21
QFP64: plastic quad flat package; 64 leads (lead length 2.35 mm); body 14 x 20 x 2.75 mm ..... SOT208-1 ... 5-22
QFP64: plastic quad flat package; 64 leads (lead length 1.95mm); body 14 x 20 x 2.8 mm ..... SOT319-2 ... 5-23
QFP80: plastic quad flat package; 80 leads (lead length 1.6 mm); body 14 x 20 x 3.0 mm ..... SOT310-1 ... 5-24
QFP80: plastic quad flat package; 80 leads (lead length 1.95 mm); body 14 x 20 x 2.7 mm; high stand-off height ..... SOT318-1 ... 5-25
QFP80: plastic quad flat package; 80 leads (lead length 1.95 mm); body 14 x 20 x 2.8 mm ..... SOT318-2 ... 5-26

**Ceramic Quad Flat Package**

44-pin CerQuad J-Bend (K) Package ..... 1472A ..... 5-27
68-pin CerQuad J-Bend (K) Package ..... 1473A ..... 5-28

**Plastic Small Outline Package**

SO8: plastic small outline package; 8 leads; body width 3.9mm ..... SOT96-1 .... 5-29
SO28: plastic small outline package; 28 leads; body width 7.5mm ..... SOT136-1 ... 5-30
VSO40: plastic very small outline package; 40 leads ..... SOT158-1 ... 5-31
VSO56: plastic very small outline package; 56 leads ..... SOT190-1 ... 5-32

**Plastic Shrink Small Outline Package**

SSOP24: plastic shrink small outline package; 24 leads; body width 5.3 mm ..... SOT340-1 ... 5-33
SSOP28: plastic shrink small outline package; 28 leads; body width 5.3mm ..... SOT341-1 ... 5-34

## INTRODUCTION

There is no soldering method that is ideal for all IC packages. Wave soldering is often preferred when though-hole and surface mounted components are mixed on one printed-circuit board. However, wave soldering is not always suitable for surface mounted ICs, or for printed-circuits with high population densities. In these situations reflow soldering is often used.

This text gives a very brief insight to a complex technology. A more in-depth account of soldering ICs can be found in our *"IC Package Databook"* (order code 9398 652 90011).

## THROUGH-HOLE MOUNTED PACKAGES

### Table 1. Types of through-hole mounted packages

| TYPE | DESCRIPTION |
|---|---|
| DIP | plastic dual in-line package |
| SDIP | plastic shrink dual in-line package |
| HDIP | plastic heat-dissipating dual in-line package |
| DBS | plastic dual in-line bent from a single in-line package |
| SIL | plastic single in-line package |

### Soldering by dipping or wave

The maximum permissible temperature of the solder is 260°C; solder at this temperature must not be in contact with the joint for more than 5 seconds. The total contact time of successive solder waves must not exceed 5 seconds.

The device may be mounted to the seating plane, but the temperature of the plastic body must not exceed the specified maximum storage temperature ($T_{stg\ max}$). If the printed-circuit board has been pre-heated, forced cooling may be necessary immediately after soldering to keep the temperature within the permissible limit.

### Repairing soldered joints

Apply a low voltage soldering iron (less than 24V) to the lead(s) of the package, below the seating plane or not more than 2mm above it. If the temperature of the soldering iron bit is less than 300°C it may remain in contact for up to 10 seconds. If the bit temperature is between 300 and 400°C, contact may be up to 5 seconds.

## SURFACE MOUNTED PACKAGES

### Table 2. Types of surface mounted packages

| TYPE | DESCRIPTION |
|---|---|
| SO | plastic small outline package |
| SSOP | plastic shrink small outline package |
| TSSOP | plastic thin shrink small outline package |
| VSO | plastic very small outline package |
| QFP | plastic quad flat package |
| LQFP | plastic low profile quad flat package |
| SQFP | plastic shrink quad flat package |
| TQFP | plastic thin quad flat package |
| PLCC | plastic leaded chip carrier |

### Reflow soldering

Reflow soldering techniques are suitable for all SMD packages, ease of soldering varies with the type of package as indicated in Table 3.

The choice of heating method may be indluenced by larger plastic packages (QFP or PLCC with 44 leads, or more). If infrared or vapor phase heating is used and the large packages are not absolutely dry (less than 0.1% moisture content by weight), vaporization of the small amount of moisture in them can cause cracking of the plastic body. For more information on moisture prevention, refer to the Drypack chapter in our *"Quality Reference Manual"* (order code 9398 510 63011).

Reflow soldering requires solder paste (a suspension of fine solder particles, flux and binding agent) to be applied to the printed-circuit board by screen printing, stenciling or pressure-syringe dispensing before package placement.

Several techniques exist for reflowing; for example, thermal conduction by heated belt. Dwell times vary between 50 and 300 seconds depending on heating method. Typical reflow temperatures range from 215 to 250°C.

Preheating is necessary to dry the paste and evaporate the binding agent. Preheating duration: 45 minutes at 45°C.

**Table 3. Suitability of surface mounted packages for various soldering methods**

Rating from 'a' to 'd': **'a' indicates most suitable** (soldering is not difficult); **'d' indicates least suitable** (soldering is achievable with difficulty).

| TYPE | REFLOW METHOD | | | | | DOUBLE WAVE METHOD |
|---|---|---|---|---|---|---|
| | INFRARED | HOT BELT | HOT GAS | VAPOR PHASE | RESISTANCE | |
| SO | a | a | a | a | d | a |
| SSOP | a | a | a | c | d | c |
| TSSOP | b | b | b | c | d | d |
| VSO | b | b | a | b | a | b |
| QFP | b | b | a | c | a | c |
| LQFP | b | b | a | c | d | d |
| SQFP | b | b | a | c | d | d |
| TQFP | b | b | a | c | d | d |
| PLCC | c | b | b | d | d | b |

## Wave soldering

Wave soldering is **not** recommended for SSOP, TSSOP, QFP, LQFP, SQFP or TQFP packages. This is because of the likelihood of solder bridging due to closely-spaced leads and the possibility of incomplete solder penetration in multi-lead devices.

If wave soldering cannot be avoided, the following conditions must be observed:

- A double-wave (a turbulent wave with high upward pressure followed by a smooth laminar wave) soldering technique should be used.
- For SSOP, TSSOP and VSO packages, the longitudinal axis of the package footprint must be parallel to the solder flow **and** must incorporate solder theives at the downstream end.
- For QFP, LQFP and TQFP packages, the footprint must be at and angle of 45° to the board direction **and** must incorporate solder thieves downstream and at the side corners.

Even with these conditions, only consider wave soldering for the following package types:

- SO
- VSO
- PLCC
- SSOP **only with body width 4.4mm**, e.g., SSOP16 (SOT369-1) or SSOP20 (SOT266-1).
- QFP **except** QFP52 (SOT379-1), QFP100 (SOT317-1, SOT317-2 and SOT382-1) and QFP160 (SOT322-1); these are **not** suitable for wave soldering.
- LQFP **except** LQFP32 (SOT401-1), LQFP48 (SOT313-1, SOT313-2), LQFP64 (SOT314-2), LQFP80 (SOT315-1); these are **not** suitable for wave soldering.
- TQFP **except** TQFP64 (SOT357-1), TQFP80 (SOT375-1) and TQFP100 (SOT386-1); these are **not** suitable for wave soldering.

SQFP are **not** suitable for wave soldering.

During placement and before soldering, the package must be fixed with a droplet of adhesive. The adhesive can be applied by screen printing, pin transfer or syringe dispensing. The package can be soldered after the adhesive is cured.

Maximum permissible solder temperature is 260°C, and maximum duration of package immersion in solder is 10 seconds, if cooled to less than 150°C within 6 seconds. Typical dwell time is 4 seconds at 250°C.

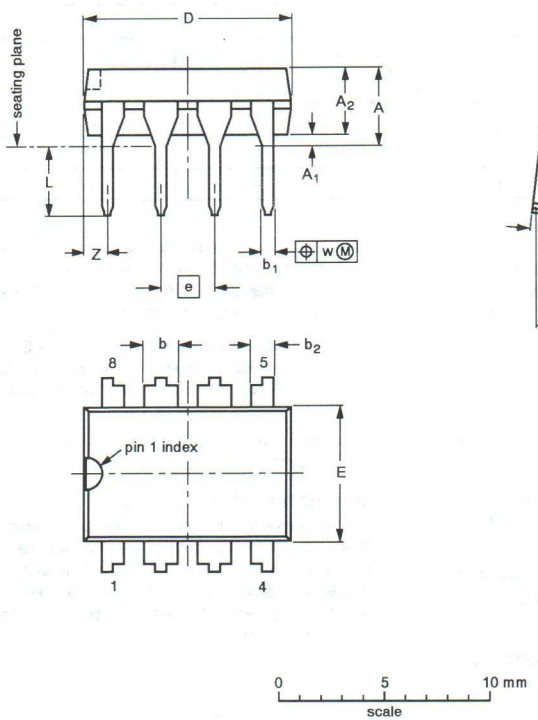
A mildly-activated flux will eliminate the need for removal of corrosive residues in most applications.

## Repairing soldered joints

Fix the component by first soldering two diagonally-opposite end leads. Use only a low voltage soldering iron (less than 24V) applied to the flat part of the lead. Contact time must be limited to 10 seconds at up to 300°C. When using a dedicated tool, all other leads can be soldered in one operation within 2 to 5 seconds at between 270 and 320°C.

# Package outlines

## DIP8: plastic dual in-line package; 8 leads (300 mil) SOT97-1




**DIMENSIONS (inch dimensions are derived from the original mm dimensions)**

| UNIT | A max. | $A_1$ min. | $A_2$ max. | b | $b_1$ | $b_2$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $e_1$ | L | $M_E$ | $M_H$ | w | $Z^{(1)}$ max. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 4.2 | 0.51 | 3.2 | 1.73<br>1.14 | 0.53<br>0.38 | 1.07<br>0.89 | 0.36<br>0.23 | 9.8<br>9.2 | 6.48<br>6.20 | 2.54 | 7.62 | 3.60<br>3.05 | 8.25<br>7.80 | 10.0<br>8.3 | 0.254 | 1.15 |
| inches | 0.17 | 0.020 | 0.13 | 0.068<br>0.045 | 0.021<br>0.015 | 0.042<br>0.035 | 0.014<br>0.009 | 0.39<br>0.36 | 0.26<br>0.24 | 0.10 | 0.30 | 0.14<br>0.12 | 0.32<br>0.31 | 0.39<br>0.33 | 0.01 | 0.045 |

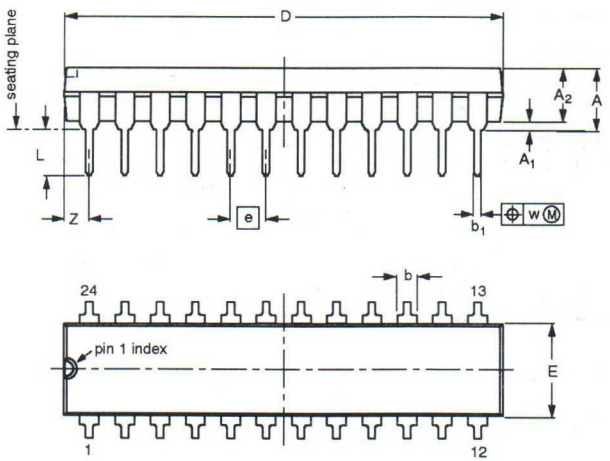
**Note**

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT97-1 | 050G01 | MO-001AN | | | | ~~92-11-17~~<br>95-02-04 |

# Package outlines

## DIP24: plastic dual in-line package; 24 leads (300 mil) SOT222-1







**DIMENSIONS (millimetre dimensions are derived from the original inch dimensions)**

| UNIT | A max. | $A_1$ min. | $A_2$ max. | b | $b_1$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $e_1$ | L | $M_E$ | $M_H$ | w | $Z^{(1)}$ max. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 4.70 | 0.38 | 3.94 | 1.63<br>1.14 | 0.56<br>0.43 | 0.36<br>0.25 | 31.9<br>31.5 | 6.73<br>6.48 | 2.54 | 7.62 | 3.51<br>3.05 | 8.13<br>7.62 | 10.03<br>7.62 | 0.25 | 2.05 |
| inches | 0.185 | 0.015 | 0.155 | 0.064<br>0.045 | 0.022<br>0.017 | 0.014<br>0.010 | 1.256<br>1.240 | 0.265<br>0.255 | 0.100 | 0.300 | 0.138<br>0.120 | 0.32<br>0.30 | 0.395<br>0.300 | 0.01 | 0.081 |

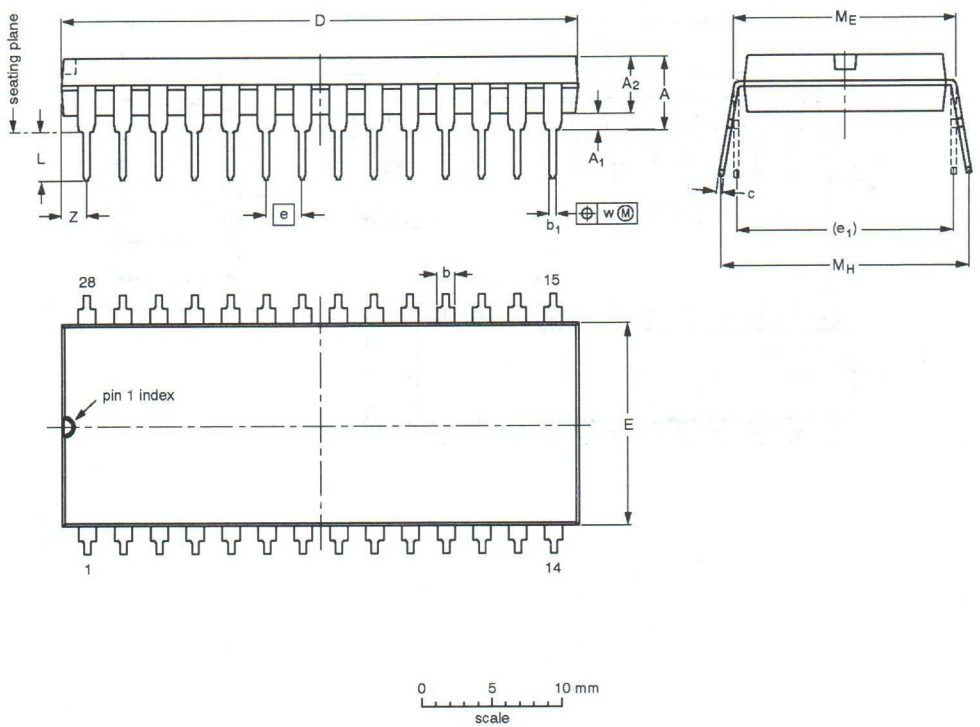
**Note**

1. Plastic or metal protrusions of 0.01 inches maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT222-1 | | MS-001AF | | | | 95-03-11 |

# Package outlines

## DIP28: plastic dual in-line package; 28 leads (600 mil); long body — SOT117-2




**DIMENSIONS (millimetre dimensions are derived from the original inch dimensions)**

| UNIT | A max. | $A_1$ min. | $A_2$ max. | b | $b_1$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $e_1$ | L | $M_E$ | $M_H$ | w | $Z^{(1)}$ max. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 5.08 | 0.51 | 3.94 | 1.63<br>1.14 | 0.56<br>0.43 | 0.38<br>0.25 | 37.08<br>35.94 | 14.22<br>13.84 | 2.54 | 15.24 | 3.51<br>3.05 | 15.75<br>15.24 | 17.65<br>15.24 | 0.25 | 2.10 |
| inches | 0.200 | 0.020 | 0.155 | 0.064<br>0.045 | 0.022<br>0.017 | 0.015<br>0.010 | 1.460<br>1.415 | 0.560<br>0.545 | 0.100 | 0.600 | 0.138<br>0.120 | 0.62<br>0.60 | 0.695<br>0.600 | 0.01 | 0.083 |

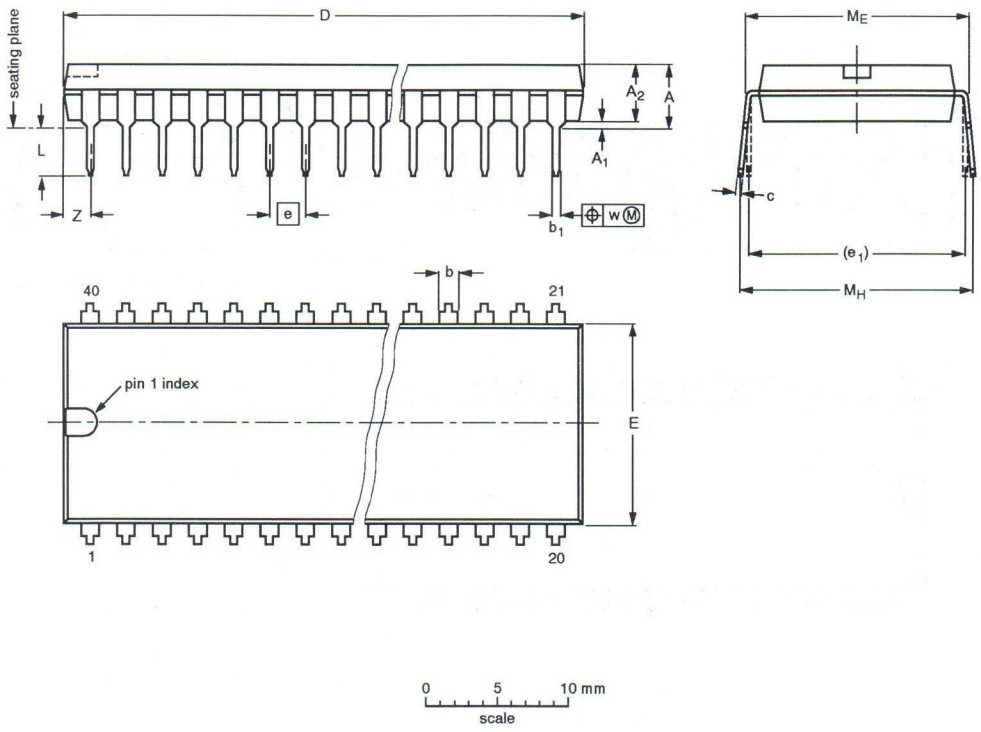
**Note**

1. Plastic or metal protrusions of 0.01 inches maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT117-2 | | MS-011AB | | | | 95-03-11 |

# Package outlines

## DIP40: plastic dual in-line package; 40 leads (600 mil) SOT129-1







**DIMENSIONS (inch dimensions are derived from the original mm dimensions)**

| UNIT | A max. | $A_1$ min. | $A_2$ max. | b | $b_1$ | c | D (1) | E (1) | e | $e_1$ | L | $M_E$ | $M_H$ | w | Z (1) max. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 4.7 | 0.51 | 4.0 | 1.70<br>1.14 | 0.53<br>0.38 | 0.36<br>0.23 | 52.50<br>51.50 | 14.1<br>13.7 | 2.54 | 15.24 | 3.60<br>3.05 | 15.80<br>15.24 | 17.42<br>15.90 | 0.254 | 2.25 |
| inches | 0.19 | 0.020 | 0.16 | 0.067<br>0.045 | 0.021<br>0.015 | 0.014<br>0.009 | 2.067<br>2.028 | 0.56<br>0.54 | 0.10 | 0.60 | 0.14<br>0.12 | 0.62<br>0.60 | 0.69<br>0.63 | 0.01 | 0.089 |

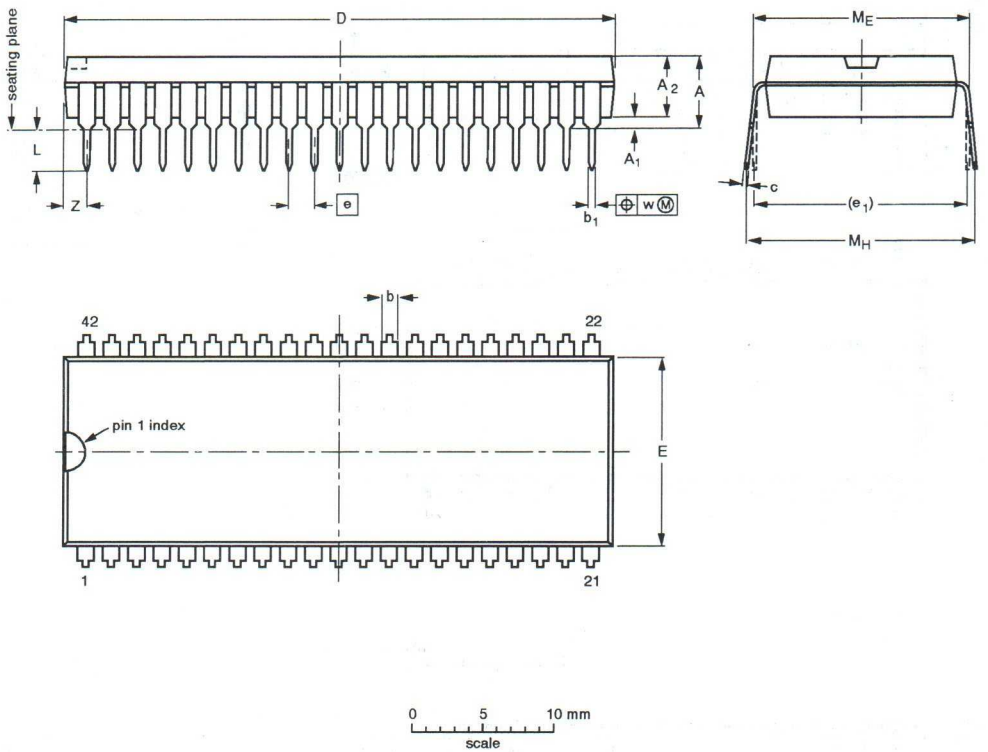
**Note**

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT129-1 | 051G08 | MO-015AJ | | | | ~~92-11-17~~<br>95-01-14 |

# Package outlines

## SDIP42: plastic shrink dual in-line package; 42 leads (600 mil) SOT270-1




**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ min. | $A_2$ max. | b | $b_1$ | c | D (1) | E (1) | e | $e_1$ | L | $M_E$ | $M_H$ | w | Z (1) max. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 5.08 | 0.51 | 4.0 | 1.3<br>0.8 | 0.53<br>0.40 | 0.32<br>0.23 | 38.9<br>38.4 | 14.0<br>13.7 | 1.778 | 15.24 | 3.2<br>2.9 | 15.80<br>15.24 | 17.15<br>15.90 | 0.18 | 1.73 |

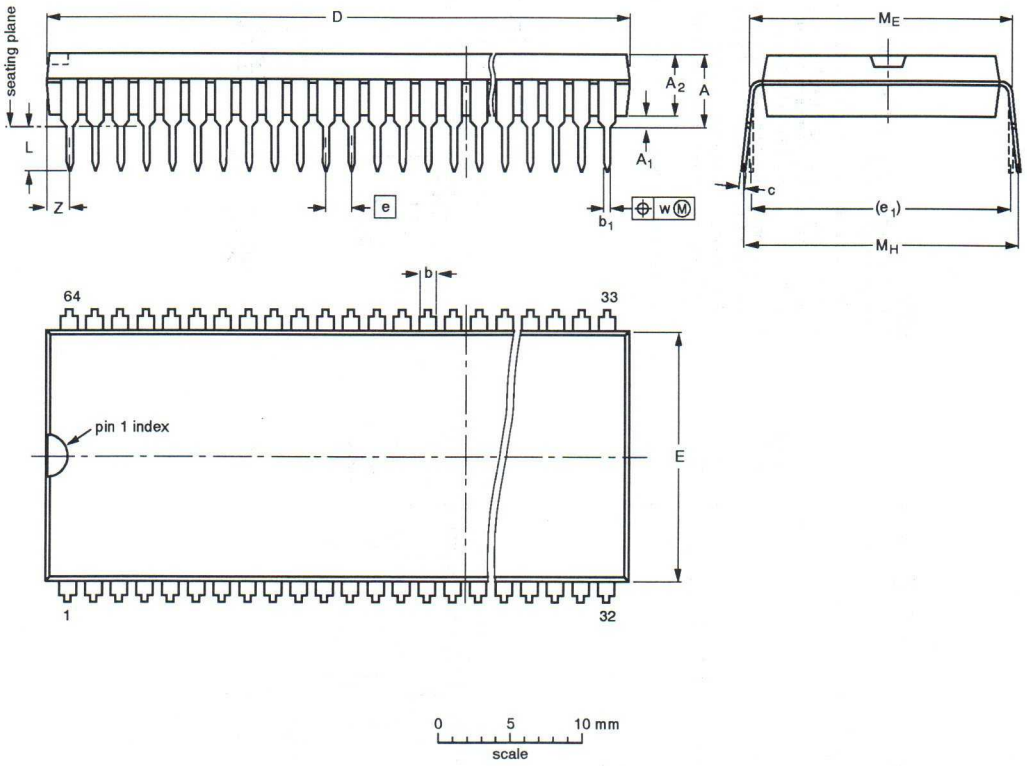
**Note**

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT270-1 | | | | | | ~~90-02-13~~<br>95-02-04 |

# Package outlines

## SDIP64: plastic shrink dual in-line package; 64 leads (750 mil) SOT274-1




**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ min. | $A_2$ max. | b | $b_1$ | c | D (1) | E (1) | e | $e_1$ | L | $M_E$ | $M_H$ | w | Z (1) max. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 5.84 | 0.51 | 4.57 | 1.3<br>0.8 | 0.53<br>0.40 | 0.32<br>0.23 | 58.67<br>57.70 | 17.2<br>16.9 | 1.778 | 19.05 | 3.2<br>2.8 | 19.61<br>19.05 | 20.96<br>19.71 | 0.18 | 1.73 |

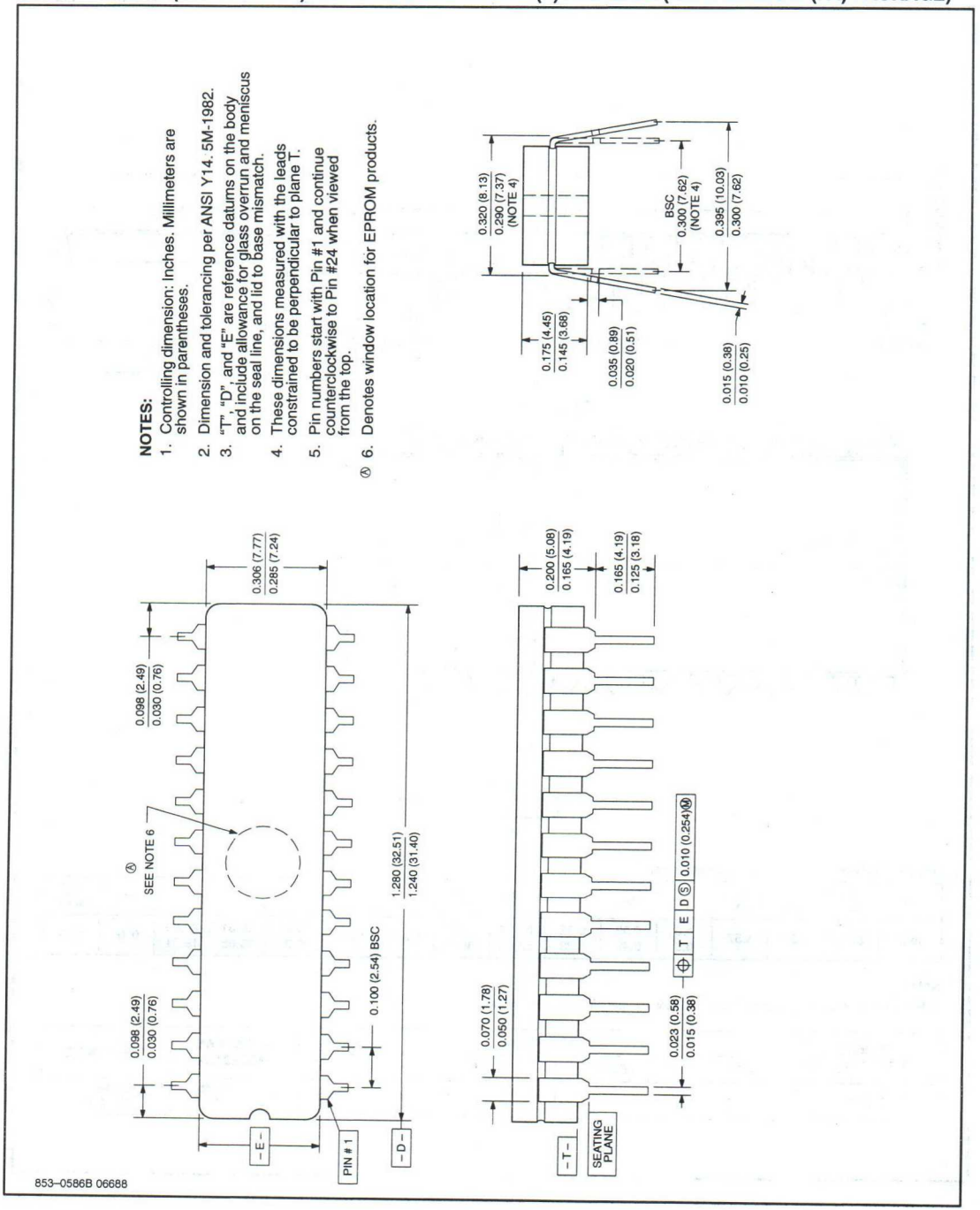
**Note**

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT274-1 | | | | | | ~~92-10-13~~<br>95-02-04 |

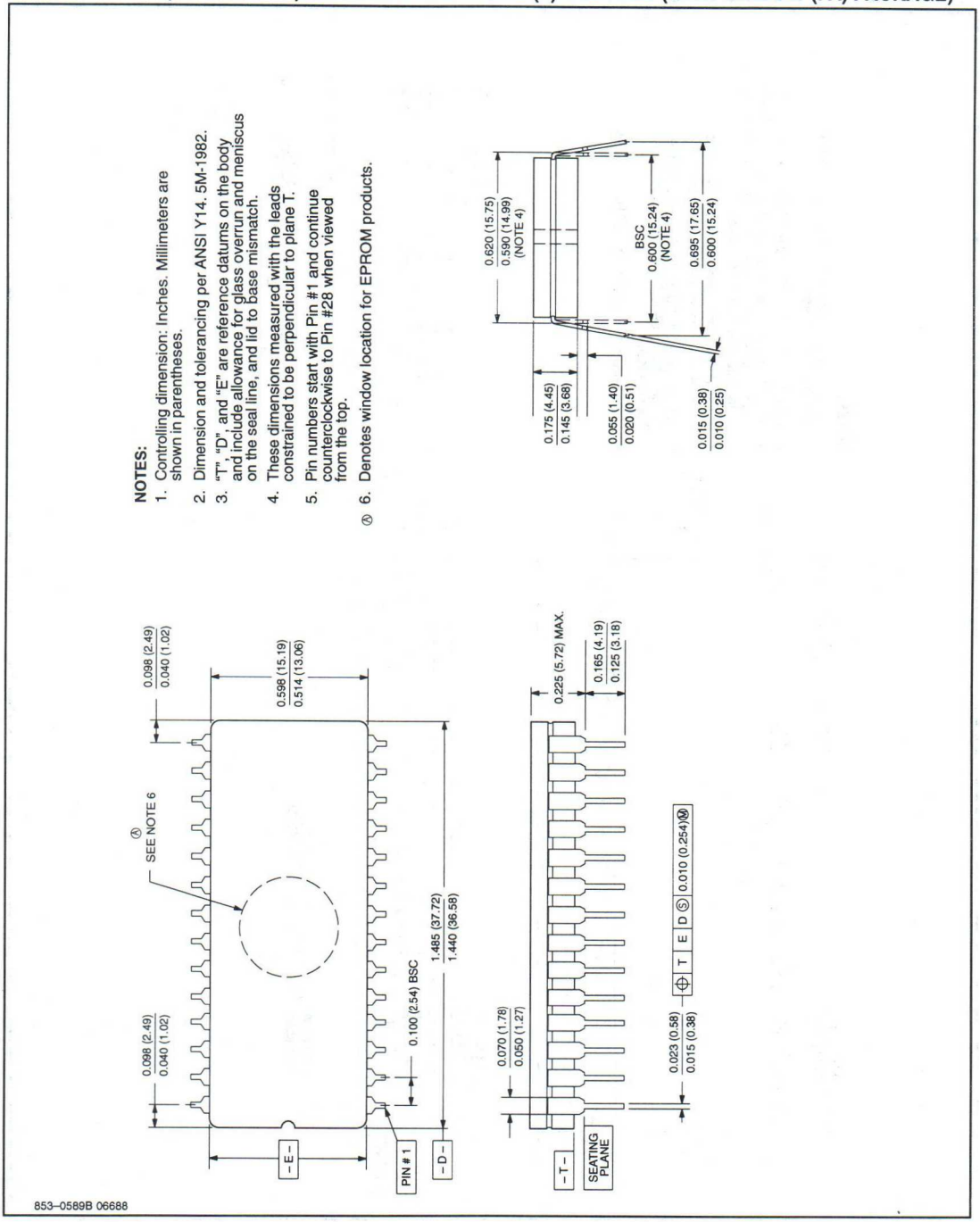
# Package outlines

## 0586B 24-PIN (300 mils wide) CERAMIC DUAL IN-LINE (F) PACKAGE (WITH WINDOW (FA) PACKAGE)



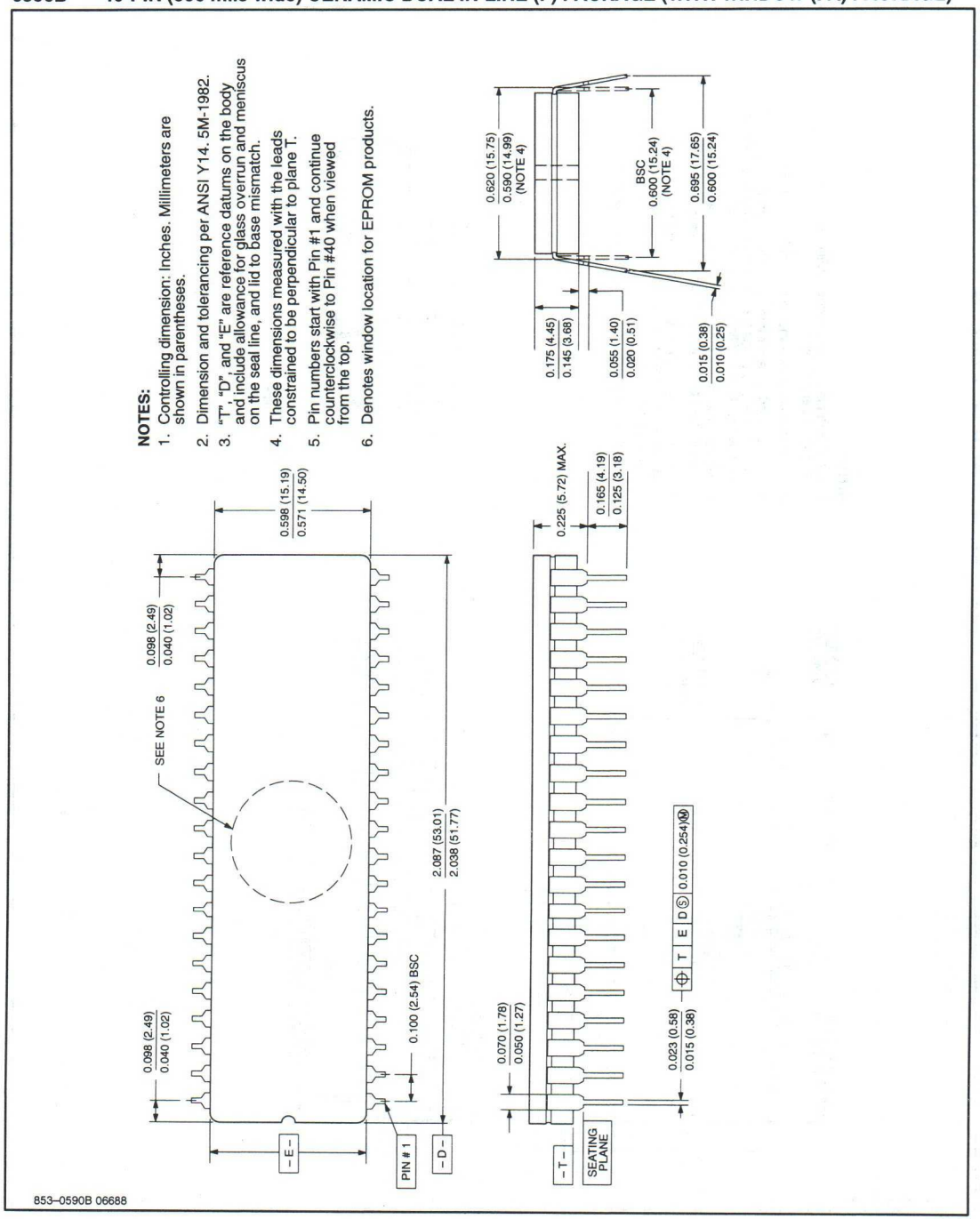
# Package outlines

## 0589B 28-PIN (600 mils wide) CERAMIC DUAL IN-LINE (F) PACKAGE (WITH WINDOW (FA) PACKAGE)



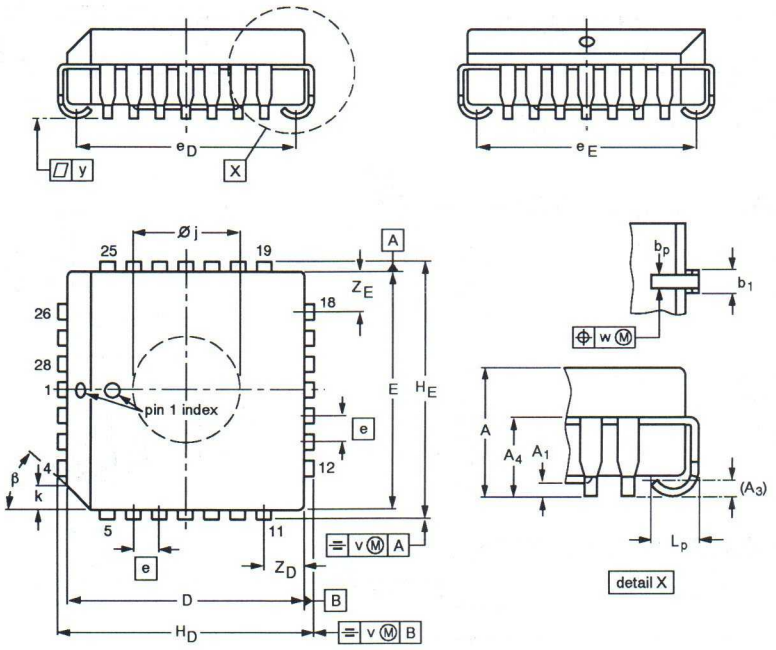
# Package outlines

## 0590B 40-PIN (600 mils wide) CERAMIC DUAL IN-LINE (F) PACKAGE (WITH WINDOW (FA) PACKAGE)



# Package outlines

## PLCC28: plastic leaded chip carrer; 28 leads; pedestal — SOT261-3




**DIMENSIONS (millimetre dimensions are derived from the original inch dimensions)**

| UNIT | A | $A_1$ min. | $A_3$ | $A_4$ max. | $b_p$ | $b_1$ | $D^{(1)}$ | $E^{(1)}$ | e | $e_D$ | $e_E$ | $H_D$ | $H_E$ | k | $Ø_j$ | $L_p$ | v | w | y | $Z_D^{(1)}$ max. | $Z_E^{(1)}$ max. | β |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 4.57<br>4.19 | 0.13 | 0.25 | 3.05 | 0.53<br>0.33 | 0.81<br>0.66 | 11.58<br>11.43 | 11.58<br>11.43 | 1.27 | 10.92<br>9.91 | 10.92<br>9.91 | 12.57<br>12.32 | 12.57<br>12.32 | 1.22<br>1.07 | 5.69<br>5.54 | 1.44<br>1.02 | 0.18 | 0.18 | 0.10 | 2.06 | 2.06 | 45° |
| inches | 0.180<br>0.165 | 0.005 | 0.01 | 0.12 | 0.021<br>0.013 | 0.032<br>0.026 | 0.456<br>0.450 | 0.456<br>0.450 | 0.05 | 0.430<br>0.390 | 0.430<br>0.390 | 0.495<br>0.485 | 0.495<br>0.485 | 0.048<br>0.042 | 0.224<br>0.218 | 0.057<br>0.040 | 0.007 | 0.007 | 0.004 | 0.081 | 0.081 | |

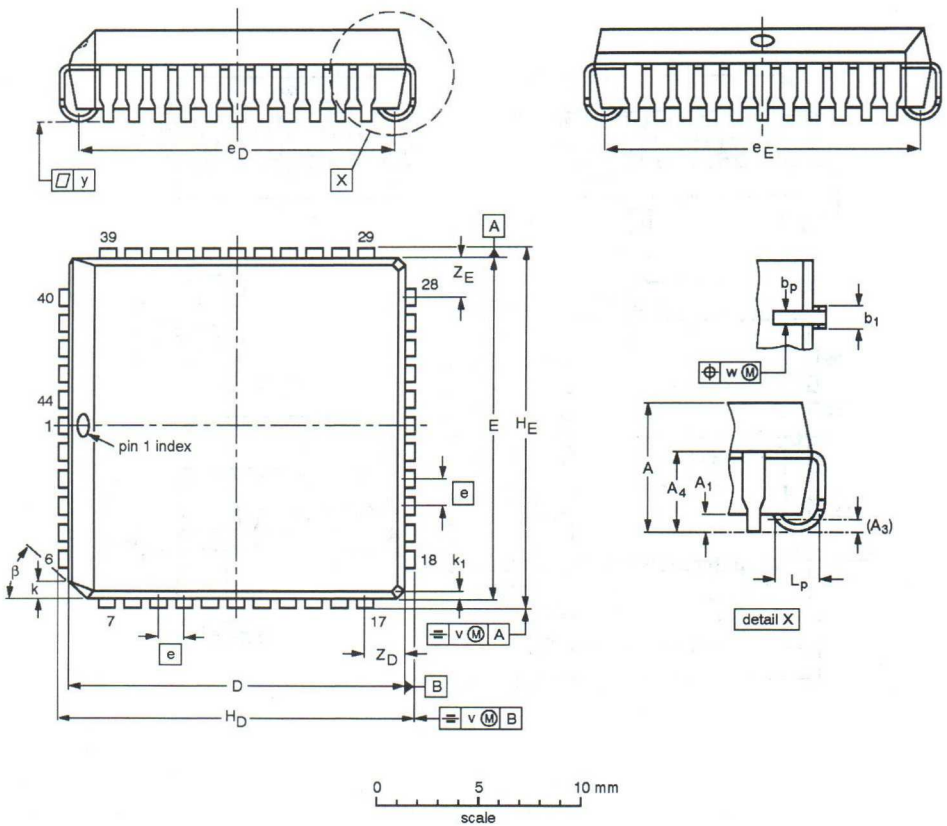
**Note**

1. Plastic or metal protrusions of 0.01 inches maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT261-3 | | MO-047AB | | | | ~~92-11-17~~<br>95-02-25 |

# Package outlines

## PLCC44: plastic leaded chip carrier; 44 leads — SOT187-2




**DIMENSIONS (millimetre dimensions are derived from the original inch dimensions)**

| UNIT | A | $A_1$ min. | $A_3$ | $A_4$ max. | $b_p$ | $b_1$ | $D^{(1)}$ | $E^{(1)}$ | e | $e_D$ | $e_E$ | $H_D$ | $H_E$ | k | $k_1$ max. | $L_p$ | v | w | y | $Z_D^{(1)}$ max. | $Z_E^{(1)}$ max. | β |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 4.57<br>4.19 | 0.51 | 0.25 | 3.05 | 0.53<br>0.33 | 0.81<br>0.66 | 16.66<br>16.51 | 16.66<br>16.51 | 1.27 | 16.00<br>14.99 | 16.00<br>14.99 | 17.65<br>17.40 | 17.65<br>17.40 | 1.22<br>1.07 | 0.51 | 1.44<br>1.02 | 0.18 | 0.18 | 0.10 | 2.16 | 2.16 | 45° |
| inches | 0.180<br>0.165 | 0.020 | 0.01 | 0.12 | 0.021<br>0.013 | 0.032<br>0.026 | 0.656<br>0.650 | 0.656<br>0.650 | 0.05 | 0.630<br>0.590 | 0.630<br>0.590 | 0.695<br>0.685 | 0.695<br>0.685 | 0.048<br>0.042 | 0.020 | 0.057<br>0.040 | 0.007 | 0.007 | 0.004 | 0.085 | 0.085 | |

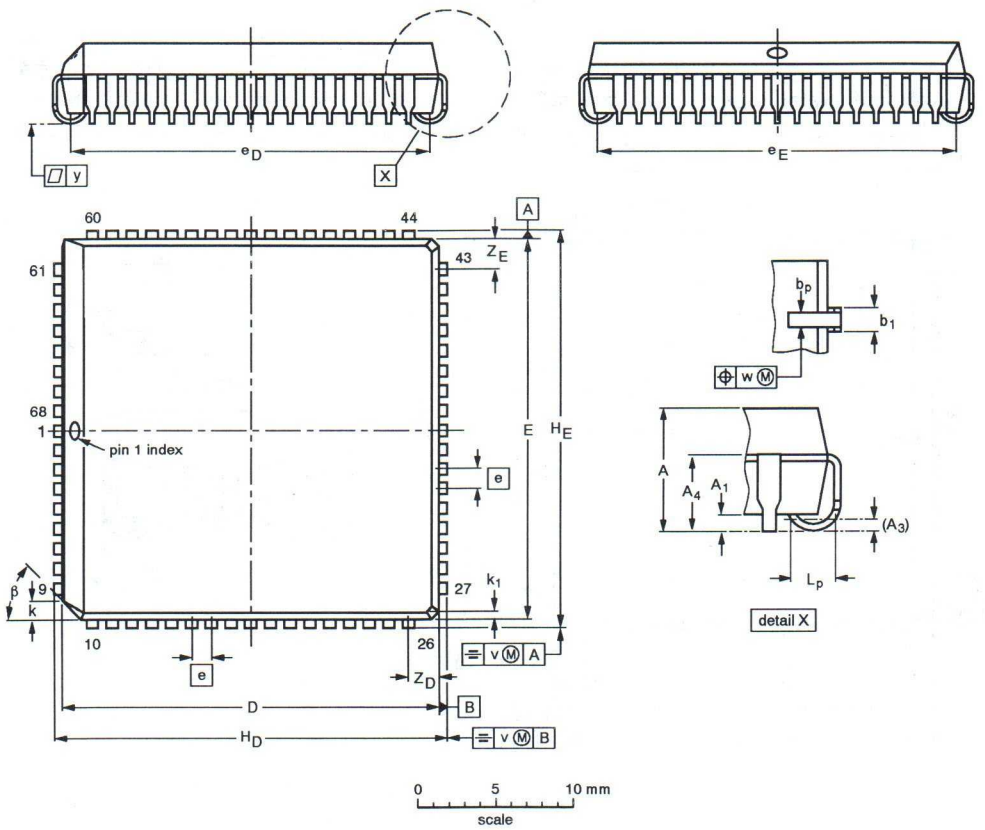
**Note**

1. Plastic or metal protrusions of 0.01 inches maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT187-2 | 112E10 | MO-047AC | | | | ~~92-11-17~~<br>95-02-25 |

# Package outlines

## PLCC68: plastic leaded chip carrier; 68 leads — SOT188-2




**DIMENSIONS (millimetre dimensions are derived from the original inch dimensions)**

| UNIT | A | $A_1$ min. | $A_3$ | $A_4$ max. | $b_p$ | $b_1$ | $D^{(1)}$ | $E^{(1)}$ | e | $e_D$ | $e_E$ | $H_D$ | $H_E$ | k | $k_1$ max. | $L_p$ | v | w | y | $Z_D^{(1)}$ max. | $Z_E^{(1)}$ max. | β |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 4.57<br>4.19 | 0.51 | 0.25 | 3.30 | 0.53<br>0.33 | 0.81<br>0.66 | 24.33<br>24.13 | 24.33<br>24.13 | 1.27 | 23.62<br>22.61 | 23.62<br>22.61 | 25.27<br>25.02 | 25.27<br>25.02 | 1.22<br>1.07 | 0.51 | 1.44<br>1.02 | 0.18 | 0.18 | 0.10 | 2.16 | 2.16 | 45° |
| inches | 0.180<br>0.165 | 0.020 | 0.01 | 0.13 | 0.021<br>0.013 | 0.032<br>0.026 | 0.958<br>0.950 | 0.958<br>0.950 | 0.05 | 0.930<br>0.890 | 0.930<br>0.890 | 0.995<br>0.985 | 0.995<br>0.985 | 0.048<br>0.042 | 0.020 | 0.057<br>0.040 | 0.007 | 0.007 | 0.004 | 0.085 | 0.085 | |

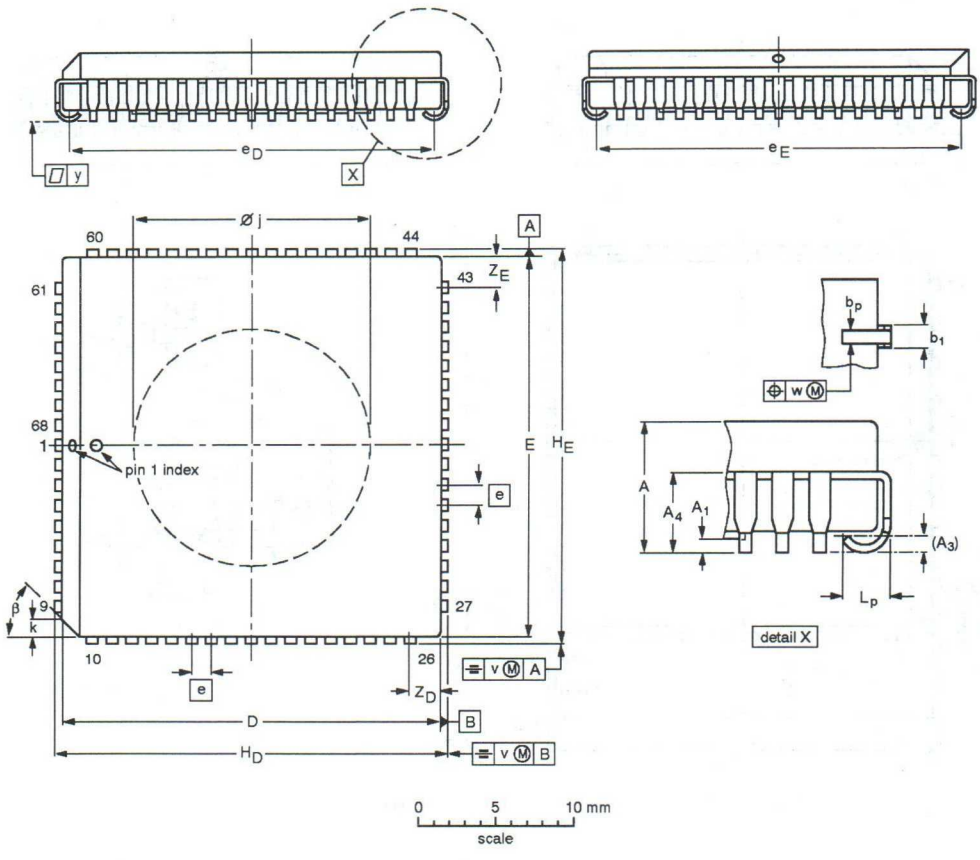
**Note**

1. Plastic or metal protrusions of 0.01 inches maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT188-2 | 112E10 | MO-047AC | | | | ~~92-11-17~~<br>95-03-11 |

# Package outlines

## PLCC68: plastic leaded chip carrier; 68 leads; pedestal — SOT188-3




**DIMENSIONS (millimetre dimensions are derived from the original inch dimensions)**

| UNIT | A | $A_1$ min. | $A_3$ | $A_4$ max. | $b_p$ | $b_1$ | $D^{(1)}$ | $E^{(1)}$ | e | $e_D$ | $e_E$ | $H_D$ | $H_E$ | k | $Ø_j$ | $L_p$ | v | w | y | $Z_D^{(1)}$ max. | $Z_E^{(1)}$ max. | β |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 4.57<br>4.19 | 0.13 | 0.25 | 3.05 | 0.53<br>0.33 | 0.81<br>0.66 | 24.33<br>24.13 | 24.33<br>24.13 | 1.27 | 23.62<br>22.61 | 23.62<br>22.61 | 25.27<br>25.02 | 25.27<br>25.02 | 1.22<br>1.07 | 15.34<br>15.19 | 1.44<br>1.02 | 0.18 | 0.18 | 0.10 | 2.06 | 2.06 | 45° |
| inches | 0.180<br>0.165 | 0.005 | 0.01 | 0.12 | 0.021<br>0.013 | 0.032<br>0.026 | 0.958<br>0.950 | 0.958<br>0.950 | 0.05 | 0.930<br>0.890 | 0.930<br>0.890 | 0.995<br>0.985 | 0.995<br>0.985 | 0.048<br>0.042 | 0.604<br>0.598 | 0.057<br>0.040 | 0.007 | 0.007 | 0.004 | 0.081 | 0.081 | |

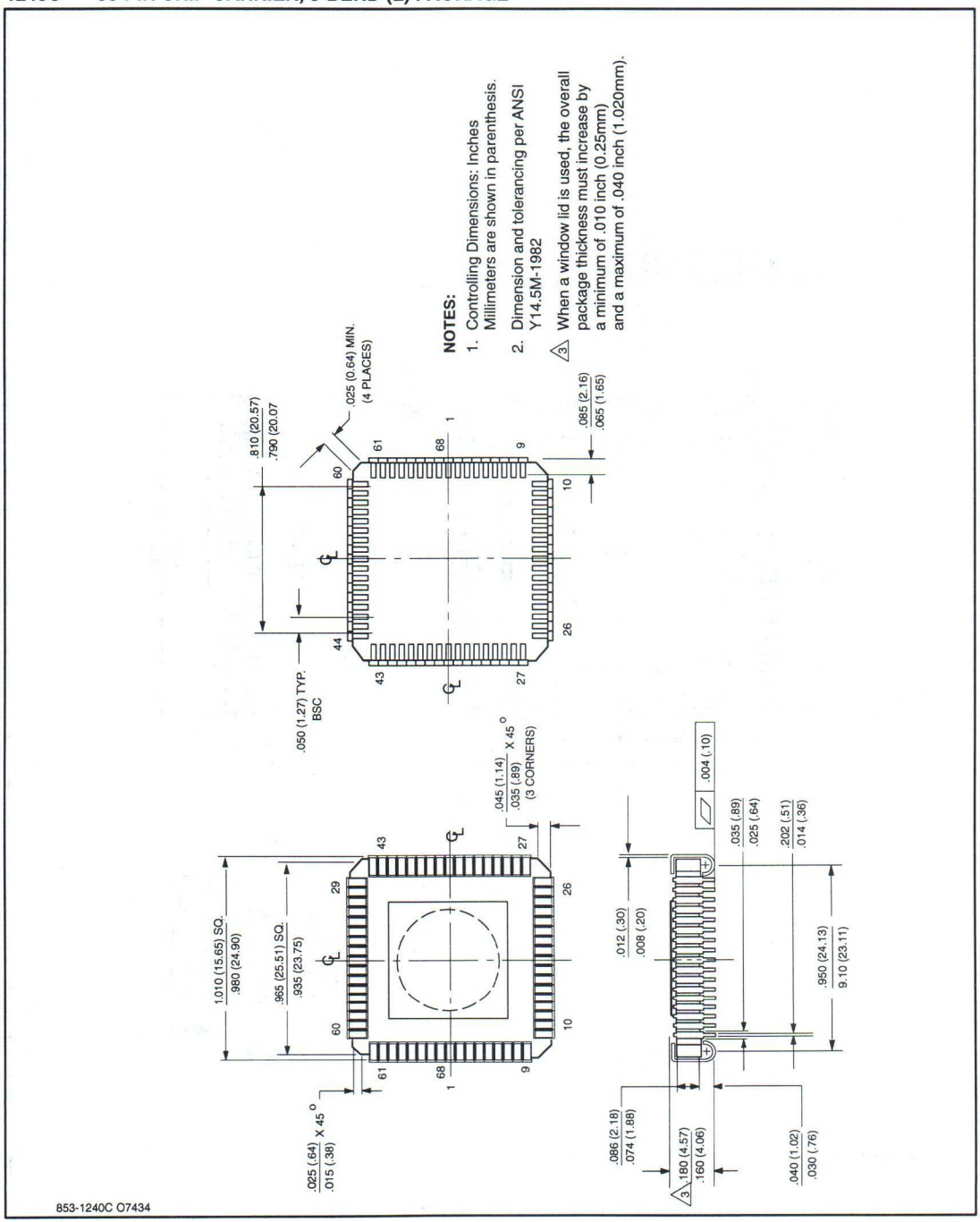
**Note**

1. Plastic or metal protrusions of 0.01 inches maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT188-3 | 112E10 | MO-047AE | | | | ~~92-11-17~~<br>95-02-25 |

# Package outlines

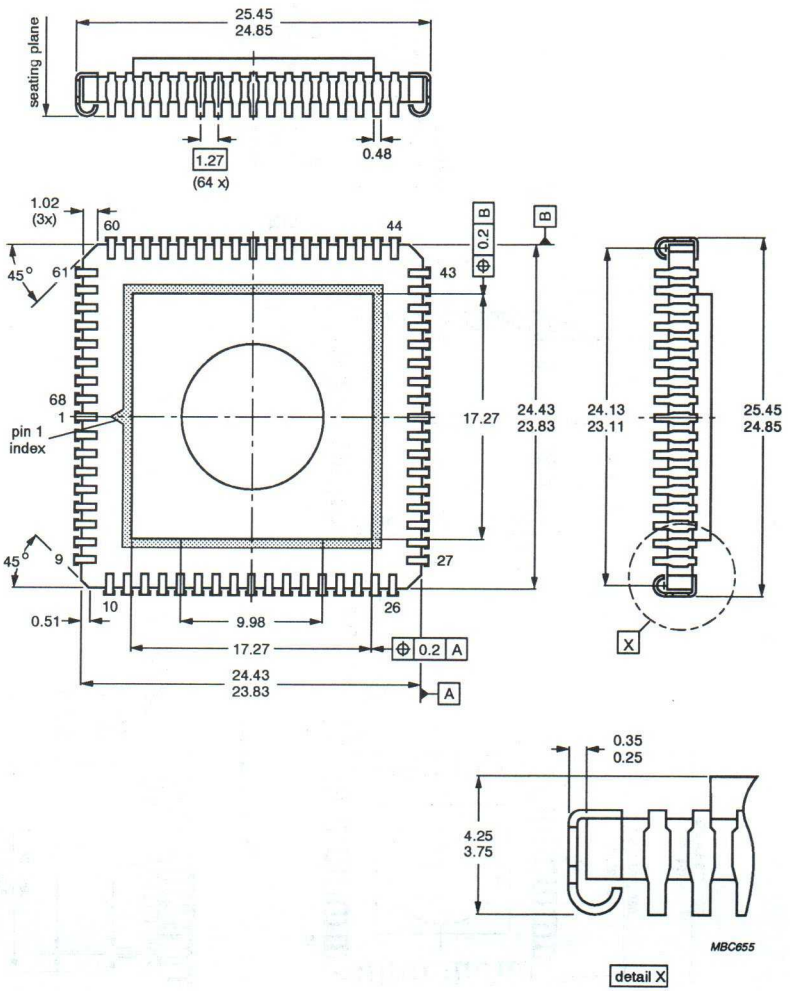
## 1240C 68-PIN CHIP CARRIER, J-BEND (L) PACKAGE



# Package outlines

## Ceramic leaded chip carrier (window); 68 leads

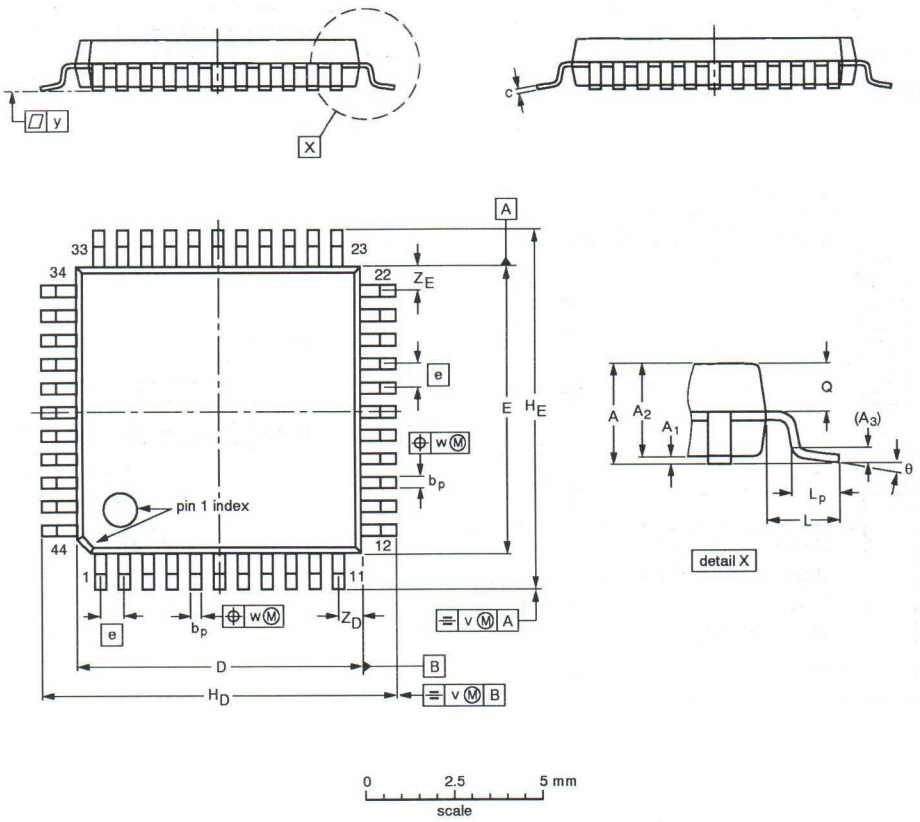
**NO330**




Dimensions in mm.

# Package outlines

## QFP44: plastic quad flat package; 44 leads (lead length 1.3 mm); body 10 x 10 x 1.75 mm — SOT307-2




**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $H_D$ | $H_E$ | L | $L_p$ | Q | v | w | y | $Z_D^{(1)}$ | $Z_E^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 2.10 | 0.25<br>0.05 | 1.85<br>1.65 | 0.25 | 0.40<br>0.20 | 0.25<br>0.14 | 10.1<br>9.9 | 10.1<br>9.9 | 0.8 | 12.9<br>12.3 | 12.9<br>12.3 | 1.3 | 0.95<br>0.55 | 0.85<br>0.75 | 0.15 | 0.15 | 0.1 | 1.2<br>0.8 | 1.2<br>0.8 | 10°<br>0° |

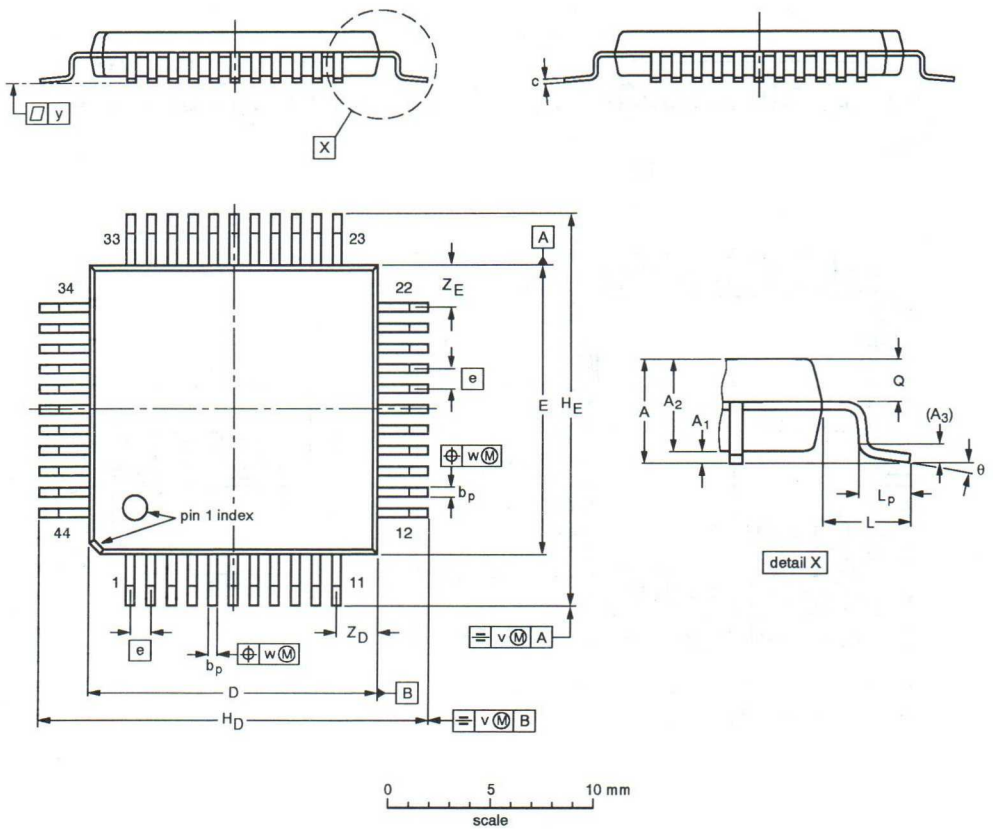
**Note**

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT307-2 | | | | | | ~~92-11-17~~<br>95-02-04 |

# Package outlines

## QFP44: plastic quad flat package; 44 leads (lead length 2.35 mm); body 14 x 14 x 2.2 mm SOT205-1




**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $H_D$ | $H_E$ | L | $L_p$ | Q | v | w | y | $Z_D^{(1)}$ | $Z_E^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 2.60 | 0.25<br>0.05 | 2.3<br>2.1 | 0.25 | 0.50<br>0.35 | 0.25<br>0.14 | 14.1<br>13.9 | 14.1<br>13.9 | 1 | 19.2<br>18.2 | 19.2<br>18.2 | 2.35 | 2.0<br>1.2 | 1.2<br>0.9 | 0.3 | 0.15 | 0.1 | 2.4<br>1.8 | 2.4<br>1.8 | 7°<br>0° |

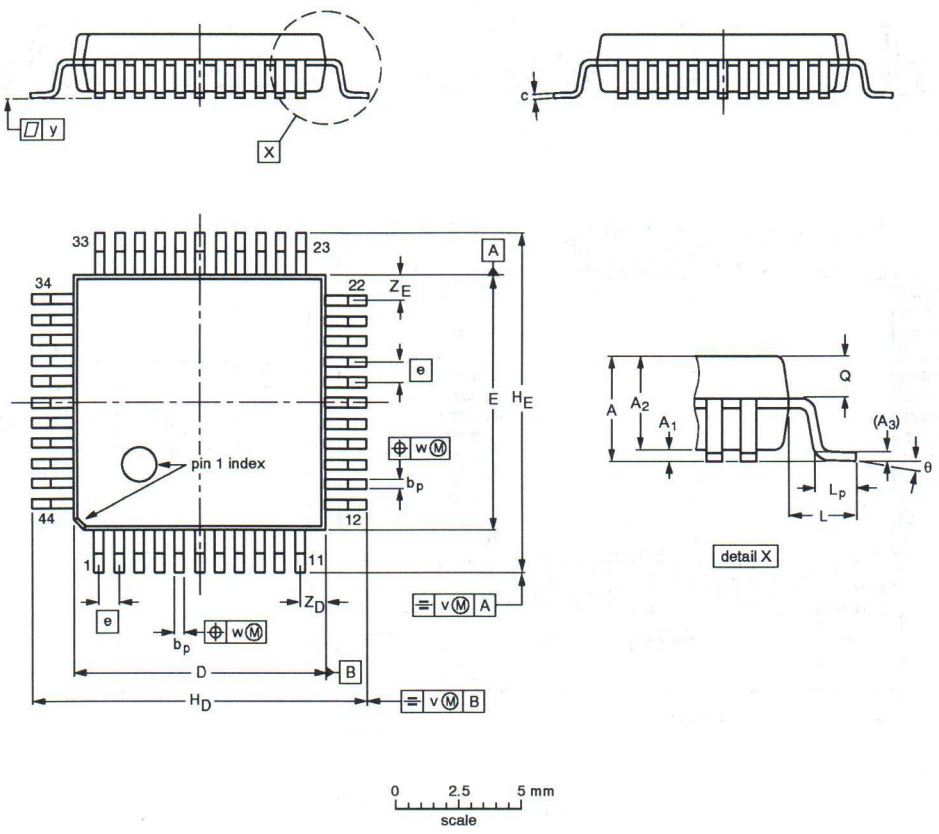
**Note**

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT205-1 | 133E01A | | | | | ~~92-11-17~~<br>95-02-04 |

# Package outlines

## LQFP44: plastic low profile quad flat package; 44 leads; body 10 x 10 x 1.4 mm — SOT389-1




**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $H_D$ | $H_E$ | L | $L_p$ | Q | v | w | y | $Z_D^{(1)}$ | $Z_E^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 1.60 | 0.15<br>0.05 | 1.45<br>1.35 | 0.25 | 0.45<br>0.30 | 0.20<br>0.12 | 10.10<br>9.90 | 10.10<br>9.90 | 0.80 | 12.15<br>11.85 | 12.15<br>11.85 | 1.0 | 0.75<br>0.45 | 0.70<br>0.57 | 0.20 | 0.20 | 0.10 | 1.14<br>0.85 | 1.14<br>0.85 | 7°<br>0° |

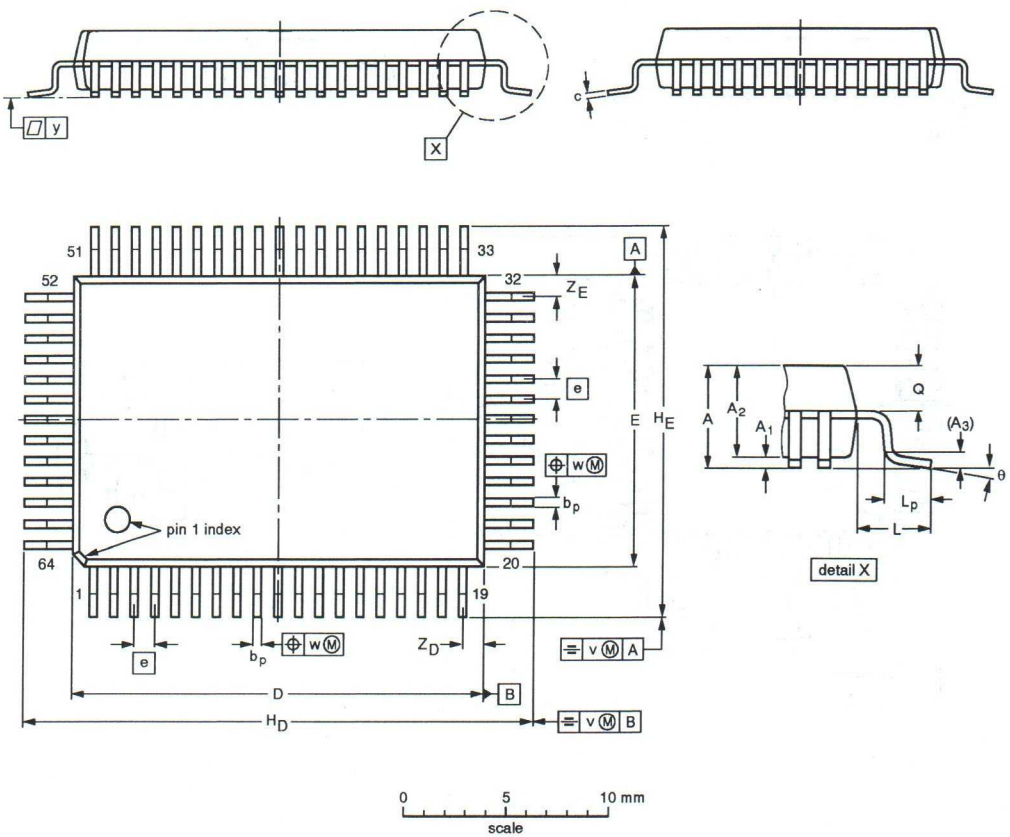
**Note**

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT389-1 | | | | | | 95-12-19 |

# Package outlines

## QFP64: plastic quad flat package; 64 leads (lead length 2.35 mm); body 14 x 20 x 2.75 mm SOT208-1




**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | D (1) | E (1) | e | $H_D$ | $H_E$ | L | $L_p$ | Q | v | w | y | $Z_D$ (1) | $Z_E$ (1) | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 3.2 | 0.30<br>0.05 | 2.85<br>2.65 | 0.25 | 0.50<br>0.35 | 0.25<br>0.14 | 20.1<br>19.9 | 14.1<br>13.9 | 1 | 25.2<br>24.2 | 19.2<br>18.2 | 2.35 | 1.55<br>0.85 | 1.45<br>1.15 | 0.3 | 0.15 | 0.1 | 1.2<br>0.8 | 1.2<br>0.8 | 7°<br>0° |

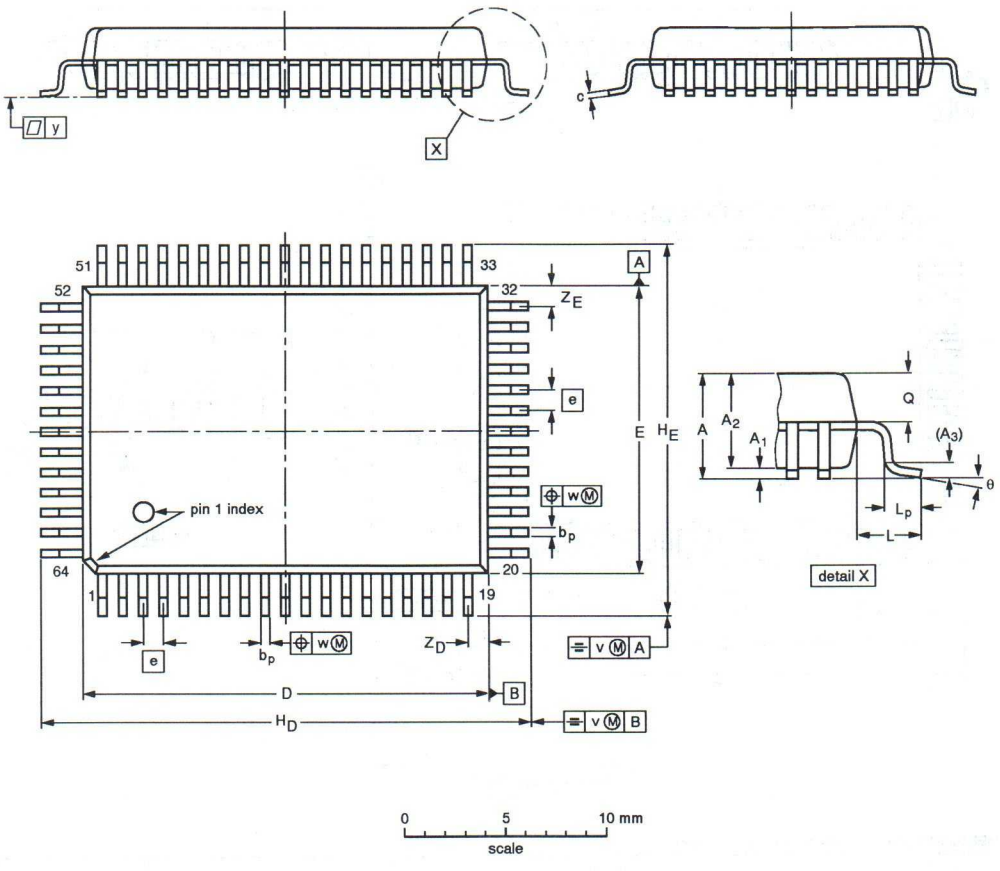
**Note**

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT208-1 | 133E02C | | | | | ~~92-11-17~~<br>95-02-04 |

# Package outlines

## QFP64: plastic quad flat package; 64 leads (lead length 1.95mm); body 14 x 20 x 2.8 mm — SOT319-2




**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $H_D$ | $H_E$ | L | $L_p$ | Q | v | w | y | $Z_D^{(1)}$ | $Z_E^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 3.20 | 0.25<br>0.05 | 2.90<br>2.65 | 0.25 | 0.50<br>0.35 | 0.25<br>0.14 | 20.1<br>19.9 | 14.1<br>13.9 | 1 | 24.2<br>23.6 | 18.2<br>17.6 | 1.95 | 1.0<br>0.6 | 1.4<br>1.2 | 0.2 | 0.2 | 0.1 | 1.2<br>0.8 | 1.2<br>0.8 | 7°<br>0° |

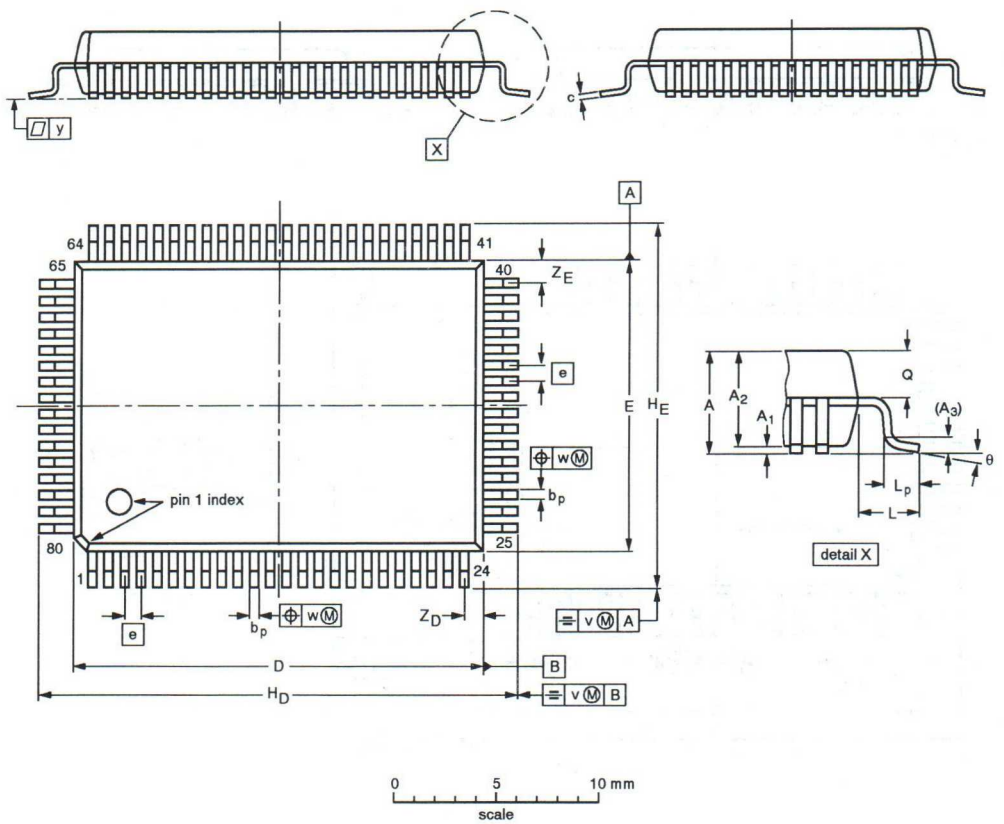
**Note**

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT319-2 | | | | | | ~~92-11-17~~<br>95-02-04 |

# Package outlines

## QFP80: plastic quad flat package; 80 leads (lead length 1.6 mm); body 14 x 20 x 3.0 mm — SOT310-1




**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $H_D$ | $H_E$ | L | $L_p$ | Q | v | w | y | $Z_D^{(1)}$ | $Z_E^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 3.4 | 0.2<br>0 | 3.2<br>2.8 | 0.25 | 0.45<br>0.30 | 0.25<br>0.14 | 20.1<br>19.9 | 14.1<br>13.9 | 0.8 | 23.6<br>22.8 | 17.6<br>16.8 | 1.6 | 1.1<br>0.5 | 1.55<br>1.25 | 0.3 | 0.15 | 0.1 | 1.0<br>0.6 | 1.2<br>0.8 | 7°<br>0° |

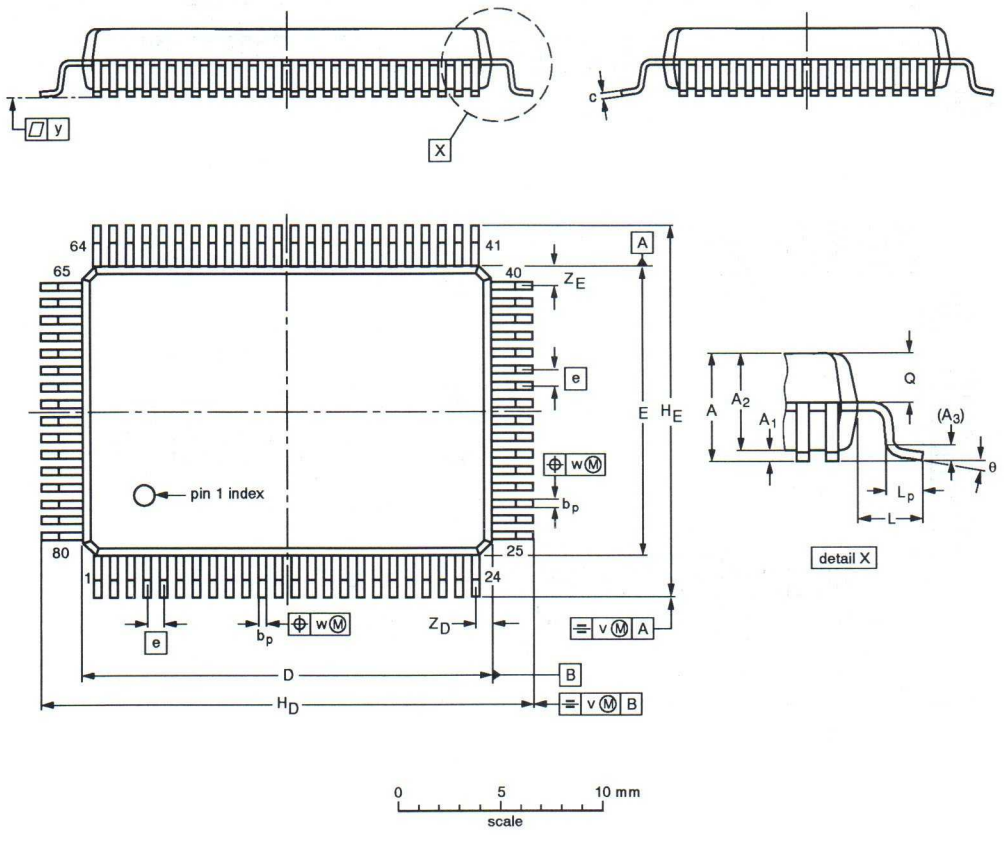
**Note**

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT310-1 | | | | | | ~~92-06-30~~<br>95-02-04 |

# Package outlines

**QFP80: plastic quad flat package;**
**80 leads (lead length 1.95 mm); body 14 x 20 x 2.7 mm; high stand-off height** **SOT318-1**




**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $H_D$ | $H_E$ | L | $L_p$ | Q | v | w | y | $Z_D^{(1)}$ | $Z_E^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 3.3 | 0.36<br>0.10 | 2.87<br>2.57 | 0.25 | 0.45<br>0.30 | 0.25<br>0.13 | 20.1<br>19.9 | 14.1<br>13.9 | 0.8 | 24.2<br>23.6 | 18.2<br>17.6 | 1.95 | 1.0<br>0.6 | 1.43<br>1.23 | 0.2 | 0.2 | 0.1 | 1.0<br>0.6 | 1.2<br>0.8 | 7°<br>0° |

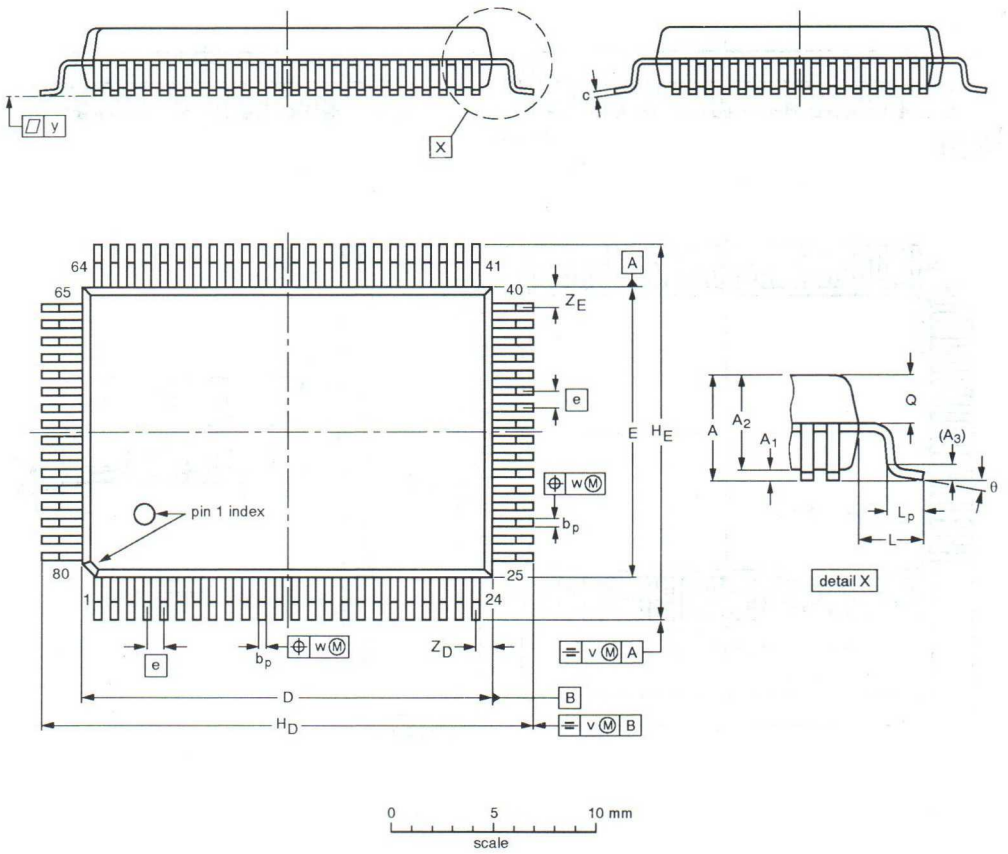
**Note**

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT318-1 | | | | | | ~~92-11-17~~<br>95-02-04 |

# Package outlines

**QFP80: plastic quad flat package; 80 leads (lead length 1.95 mm); body 14 x 20 x 2.8 mm** **SOT318-2**




**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $H_D$ | $H_E$ | L | $L_p$ | Q | v | w | y | $Z_D^{(1)}$ | $Z_E^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 3.2 | 0.25<br>0.05 | 2.90<br>2.65 | 0.25 | 0.45<br>0.30 | 0.25<br>0.14 | 20.1<br>19.9 | 14.1<br>13.9 | 0.8 | 24.2<br>23.6 | 18.2<br>17.6 | 1.95 | 1.0<br>0.6 | 1.4<br>1.2 | 0.2 | 0.2 | 0.1 | 1.0<br>0.6 | 1.2<br>0.8 | 7°<br>0° |

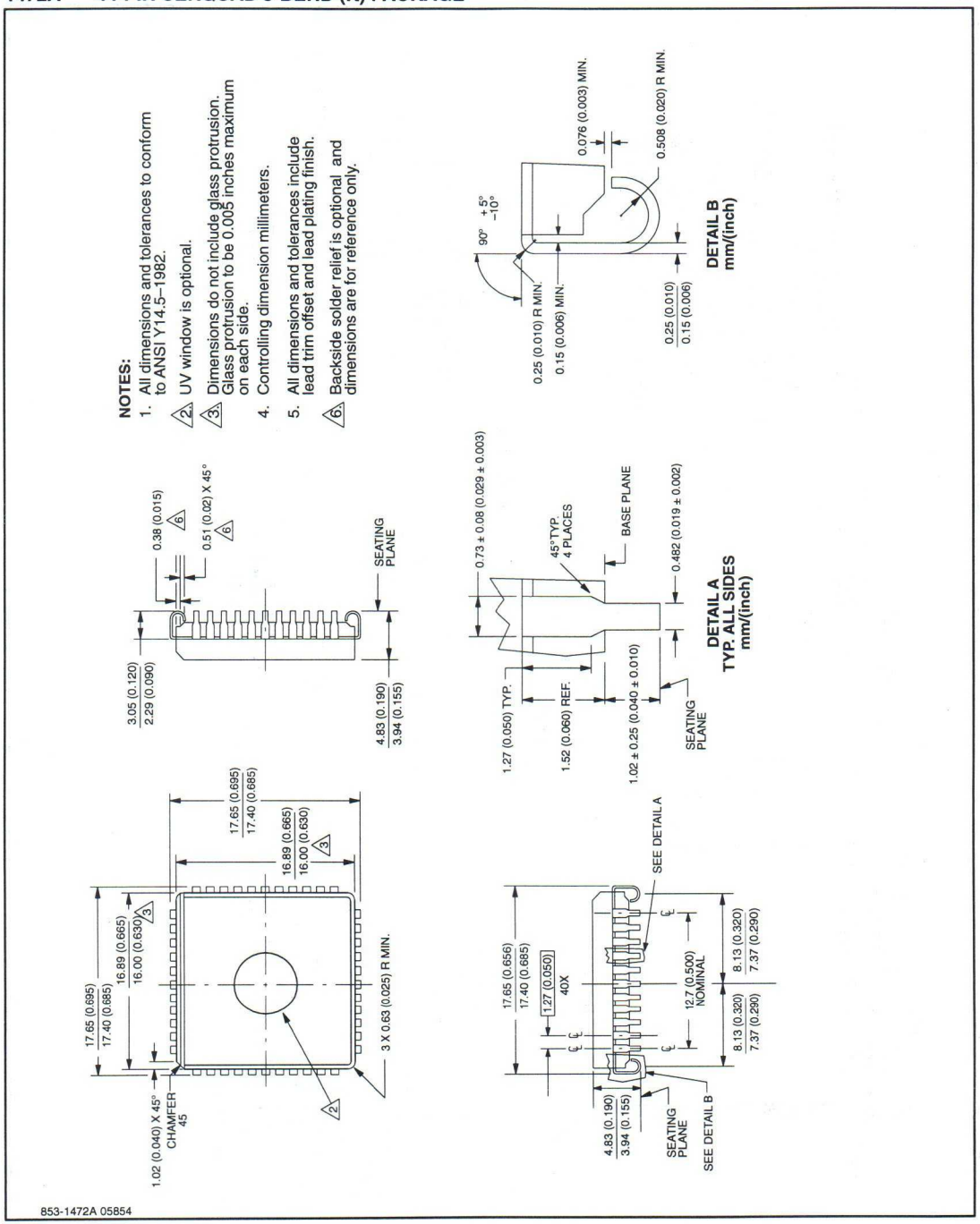
**Note**

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT318-2 | | | | | | ~~92-12-15~~<br>95-02-04 |

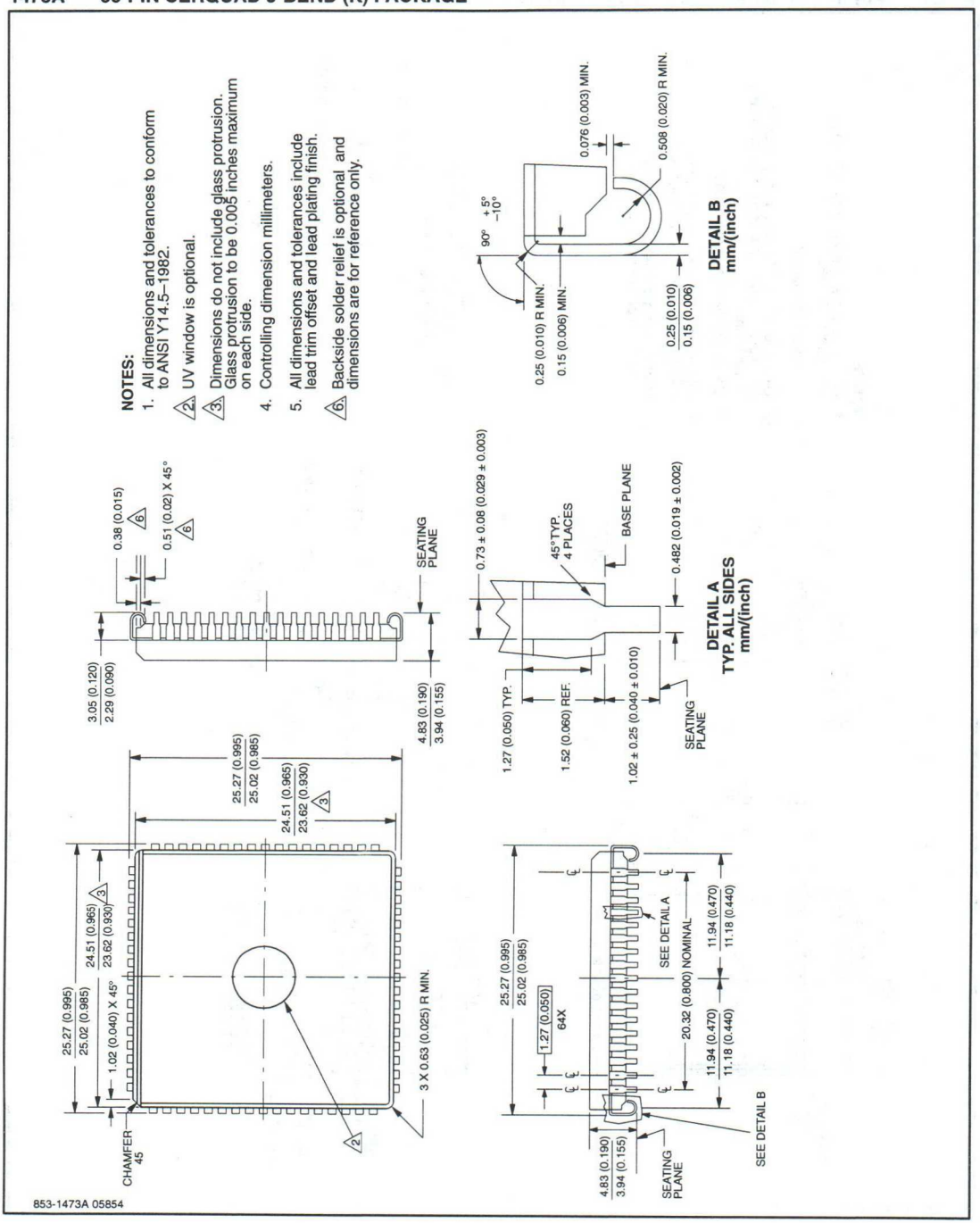
# Package outlines

## 1472A 44-PIN CERQUAD J-BEND (K) PACKAGE



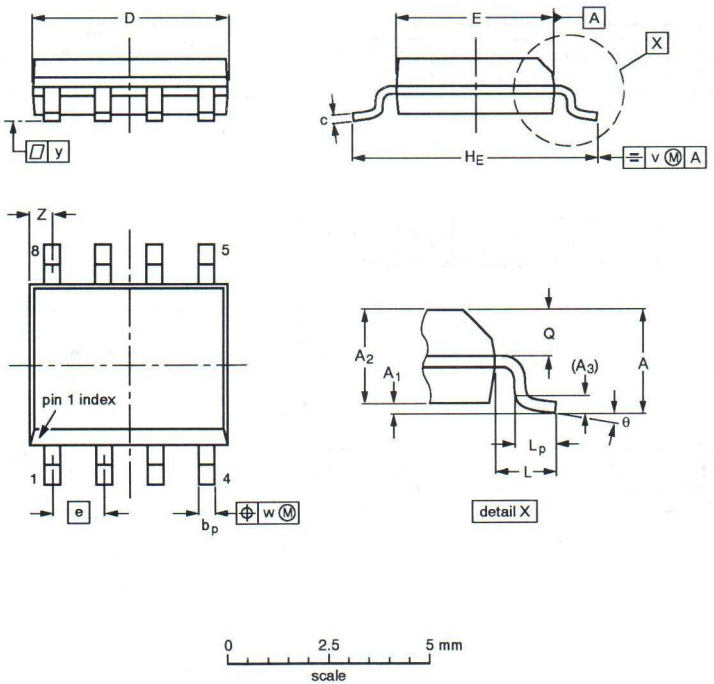
# Package outlines

## 1473A 68-PIN CERQUAD J-BEND (K) PACKAGE



# Package outlines

## SO8: plastic small outline package; 8 leads; body width 3.9mm — SOT96-1







**DIMENSIONS (inch dimensions are derived from the original mm dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(2)}$ | e | $H_E$ | L | $L_p$ | Q | v | w | y | $Z^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 1.75 | 0.25<br>0.10 | 1.45<br>1.25 | 0.25 | 0.49<br>0.36 | 0.25<br>0.19 | 5.0<br>4.8 | 4.0<br>3.8 | 1.27 | 6.2<br>5.8 | 1.05 | 1.0<br>0.4 | 0.7<br>0.6 | 0.25 | 0.25 | 0.1 | 0.7<br>0.3 | 8°<br>0° |
| inches | 0.069 | 0.0098<br>0.0039 | 0.057<br>0.049 | 0.01 | 0.019<br>0.014 | 0.0098<br>0.0075 | 0.20<br>0.19 | 0.16<br>0.15 | 0.050 | 0.24<br>0.23 | 0.041 | 0.039<br>0.016 | 0.028<br>0.024 | 0.01 | 0.01 | 0.004 | 0.028<br>0.012 | |

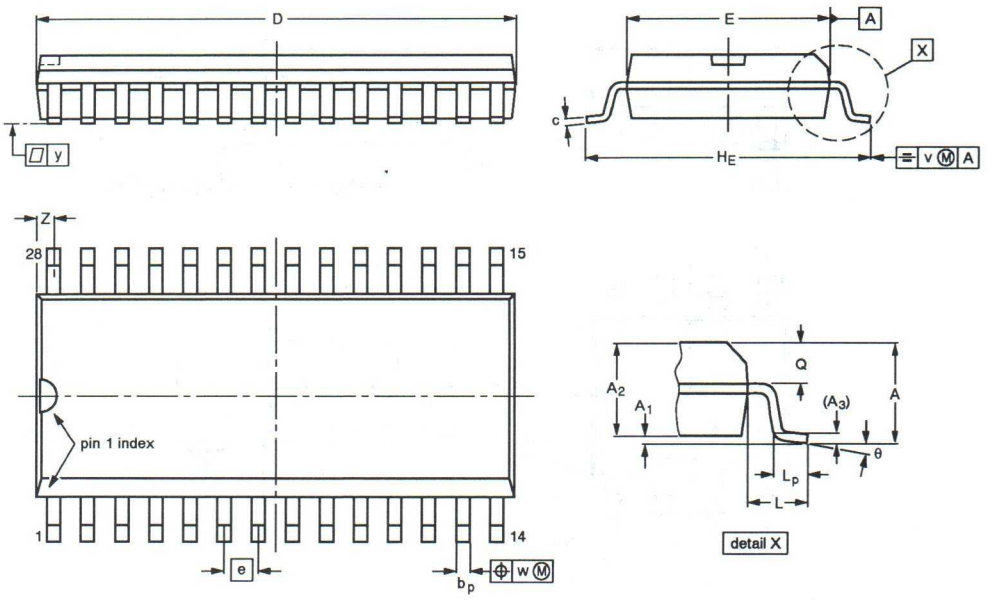
**Notes**

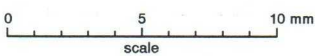
1. Plastic or metal protrusions of 0.15 mm maximum per side are not included.
2. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT96-1 | 076E03S | MS-012AA | | | | ~~92-11-17~~<br>95-02-04 |

# Package outlines

## SO28: plastic small outline package; 28 leads; body width 7.5mm — SOT136-1




**DIMENSIONS (inch dimensions are derived from the original mm dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $H_E$ | L | $L_p$ | Q | v | w | y | $Z^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 2.65 | 0.30<br>0.10 | 2.45<br>2.25 | 0.25 | 0.49<br>0.36 | 0.32<br>0.23 | 18.1<br>17.7 | 7.6<br>7.4 | 1.27 | 10.65<br>10.00 | 1.4 | 1.1<br>0.4 | 1.1<br>1.0 | 0.25 | 0.25 | 0.1 | 0.9<br>0.4 | 8°<br>0° |
| inches | 0.10 | 0.012<br>0.004 | 0.096<br>0.089 | 0.01 | 0.019<br>0.014 | 0.013<br>0.009 | 0.71<br>0.69 | 0.30<br>0.29 | 0.050 | 0.42<br>0.39 | 0.055 | 0.043<br>0.016 | 0.043<br>0.039 | 0.01 | 0.01 | 0.004 | 0.035<br>0.016 | |

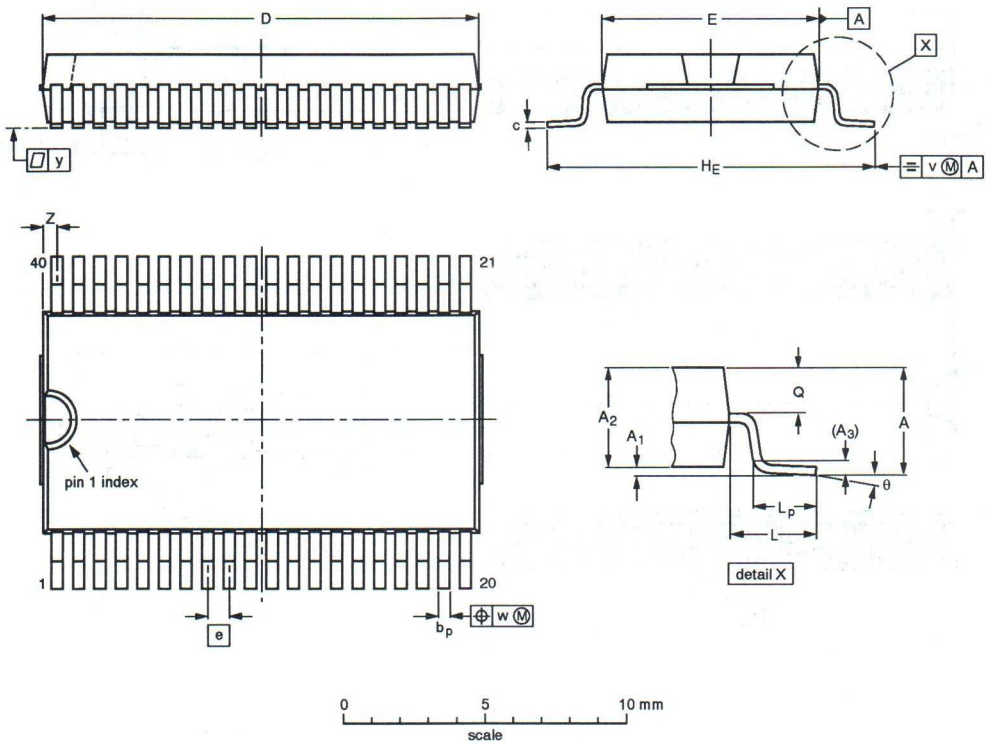
**Note**

1. Plastic or metal protrusions of 0.15 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT136-1 | 075E06 | MS-013AE | | | | ~~91-08-13~~<br>95-01-24 |

# Package outlines

## VSO40: plastic very small outline package; 40 leads — SOT158-1




**DIMENSIONS (inch dimensions are derived from the original mm dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(2)}$ | e | $H_E$ | L | $L_p$ | Q | v | w | y | $Z^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 2.70 | 0.3<br>0.1 | 2.45<br>2.25 | 0.25 | 0.42<br>0.30 | 0.22<br>0.14 | 15.6<br>15.2 | 7.6<br>7.5 | 0.762 | 12.3<br>11.8 | 2.25 | 1.7<br>1.5 | 1.15<br>1.05 | 0.2 | 0.1 | 0.1 | 0.6<br>0.3 | 7°<br>0° |
| inches | 0.11 | 0.012<br>0.004 | 0.096<br>0.089 | 0.010 | 0.017<br>0.012 | 0.0087<br>0.0055 | 0.61<br>0.60 | 0.30<br>0.29 | 0.03 | 0.48<br>0.46 | 0.089 | 0.067<br>0.059 | 0.045<br>0.041 | 0.008 | 0.004 | 0.004 | 0.024<br>0.012 | |

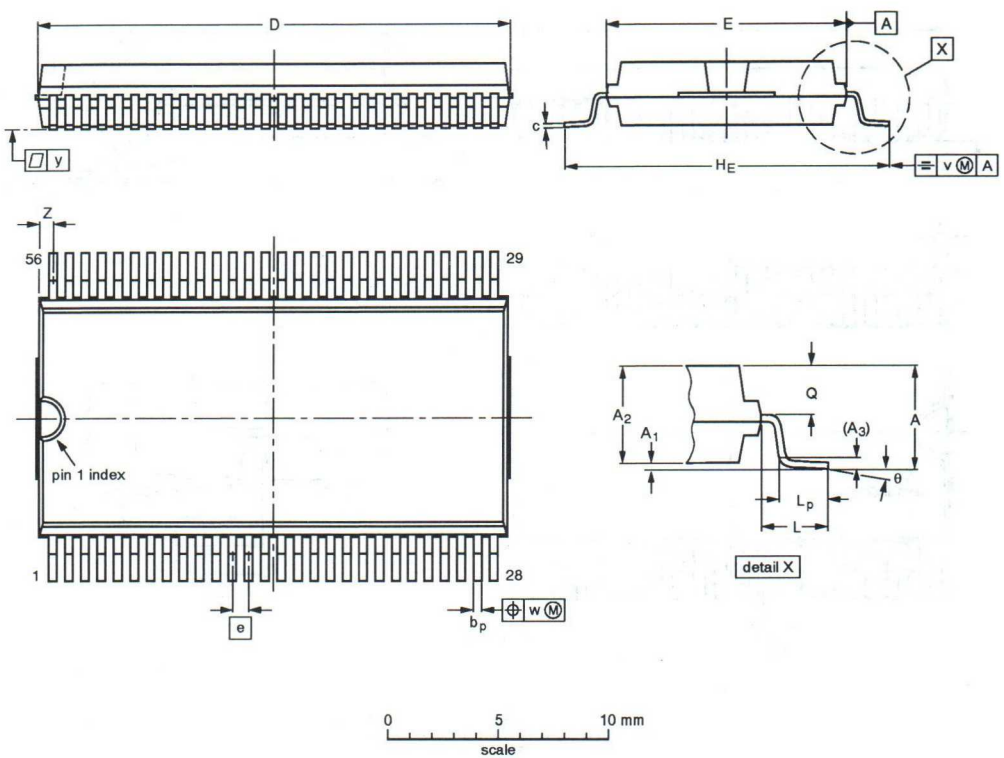
**Note**

1. Plastic or metal protrusions of 0.4 mm maximum per side are not included.
2. Plastic interlead protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT158-1 | | | | | | ~~92-11-17~~<br>95-01-24 |

# Package outlines

## VSO56: plastic very small outline package; 56 leads — SOT190-1




**DIMENSIONS (inch dimensions are derived from the original mm dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(2)}$ | e | $H_E$ | L | $L_p$ | Q | v | w | y | $Z^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 3.3 | 0.3<br>0.1 | 3.0<br>2.8 | 0.25 | 0.42<br>0.30 | 0.22<br>0.14 | 21.65<br>21.35 | 11.1<br>11.0 | 0.75 | 15.8<br>15.2 | 2.25 | 1.6<br>1.4 | 1.45<br>1.30 | 0.2 | 0.1 | 0.1 | 0.90<br>0.55 | 7°<br>0° |
| inches | 0.13 | 0.012<br>0.004 | 0.12<br>0.11 | 0.01 | 0.017<br>0.012 | 0.0087<br>0.0055 | 0.85<br>0.84 | 0.44<br>0.43 | 0.03 | 0.62<br>0.60 | 0.089 | 0.063<br>0.055 | 0.057<br>0.051 | 0.008 | 0.004 | 0.004 | 0.035<br>0.022 | |

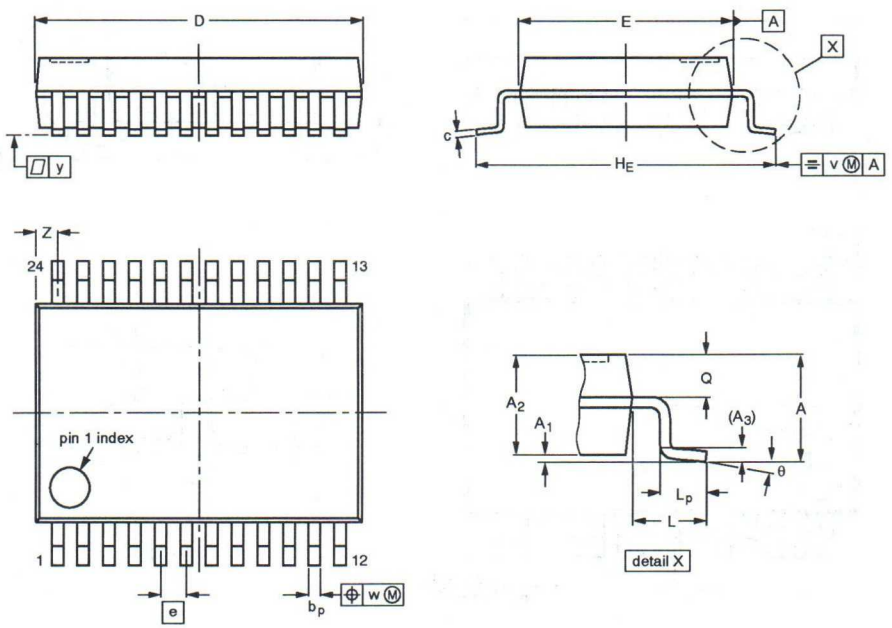
**Note**

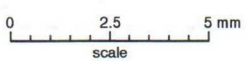
1. Plastic or metal protrusions of 0.3 mm maximum per side are not included.
2. Plastic interlead protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT190-1 | | | | | | ~~92-11-17~~<br>96-04-02 |

# Package outlines

## SSOP24: plastic shrink small outline package; 24 leads; body width 5.3 mm — SOT340-1




**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $H_E$ | L | $L_p$ | Q | v | w | y | $Z^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 2.0 | 0.21<br>0.05 | 1.80<br>1.65 | 0.25 | 0.38<br>0.25 | 0.20<br>0.09 | 8.4<br>8.0 | 5.4<br>5.2 | 0.65 | 7.9<br>7.6 | 1.25 | 1.03<br>0.63 | 0.9<br>0.7 | 0.2 | 0.13 | 0.1 | 0.8<br>0.4 | 8°<br>0° |

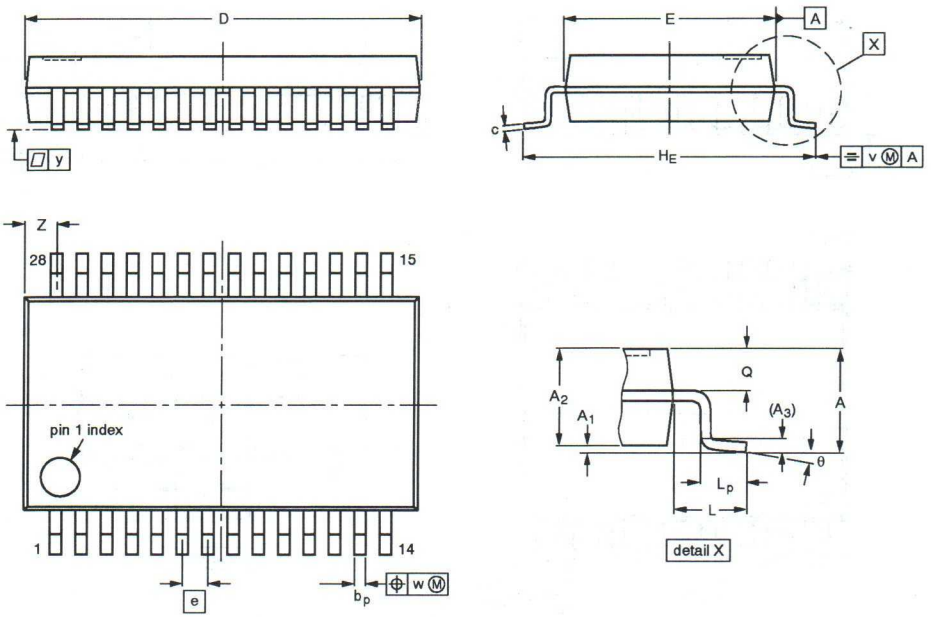
**Note**

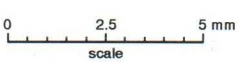
1. Plastic or metal protrusions of 0.20 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT340-1 | | MO-150AG | | | | ~~93-09-08~~<br>95-02-04 |

# Package outlines

## SSOP28: plastic shrink small outline package; 28 leads; body width 5.3mm — SOT341-1




**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $H_E$ | L | $L_p$ | Q | v | w | y | $Z^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 2.0 | 0.21<br>0.05 | 1.80<br>1.65 | 0.25 | 0.38<br>0.25 | 0.20<br>0.09 | 10.4<br>10.0 | 5.4<br>5.2 | 0.65 | 7.9<br>7.6 | 1.25 | 1.03<br>0.63 | 0.9<br>0.7 | 0.2 | 0.13 | 0.1 | 1.1<br>0.7 | 8°<br>0° |

**Note**

1. Plastic or metal protrusions of 0.20 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT341-1 | | MO-150AH | | | | ~~93-09-08~~<br>95-02-04 |

# Section 6
## Data Handbook System

**80C51-Based**
**8-Bit Microcontrollers**

CONTENTS

Data handbook system ..... 6-2

# Data handbook system

## DATA HANDBOOK SYSTEM

Philips Semiconductors data handbooks contain all pertinent data available at the time of publication and each is revised and reissued regularly.

Loose data sheets are sent to subscribers to keep them up-to-date on additions or alterations made during the lifetime of a data handbook.

Catalogs are available for selected product ranges (some catalogs are also on floppy discs).

Our data handbook titles are listed here.

### Integrated Circuits

| *Book* | *Title* |
|---|---|
| IC01 | Semiconductors for Radio and Audio Systems |
| IC02 | Semiconductors for Television and Video Systems |
| IC03 | Semiconductors for Wired Telecom Systems |
| IC04 | HE4000B Logic Family CMOS |
| IC05 | Advanced Low-power Schottky (ALS) Logic Products |
| IC06 | High-speed CMOS Logic Family |
| IC11 | General-purpose/Linear ICs |
| IC12 | $I^2C$ Peripherals |
| IC13 | Programmable Logic Devices (PLD) |
| IC14 | 8048-based 8-bit Microcontrollers |
| IC15 | FAST TTL Logic Series |
| IC16 | CMOS ICs for Clocks and Watches |
| IC17 | Semiconductors for Wireless Communications |
| IC18 | Semiconductors for In-Car Electronics |
| IC19 | ICs for Data Communications |
| IC20 | 80C51-based 8-bit Microcontrollers |
| IC22 | Desktop Video |
| IC23 | BiCMOS Bus Interface Logic |
| IC24 | Low Voltage CMOS & BiCMOS Logic |
| IC25 | 16-bit 80C51XA Microcontrollers (eXtended Architecture) |
| IC26 | IC Package Databook |

### Discrete Semiconductors

| *Book* | *Title* |
|---|---|
| SC01 | Diodes |
| SC02 | Power Diodes |
| SC03 | Thyristors and Triacs |
| SC04 | Small-signal Transistors |
| SC05 | Video Transistors and Modules for Monitors |
| SC06 | High-voltage and Switching NPN Power Transistors |
| SC07 | Small-signal Field-effect Transistors |
| SC08a | RF Power Transistors for HF and VHF |
| SC08b | RF Power Transistors for UHF |
| SC09 | RF Power Modules |
| SC13 | Power MOS Transistors including TOPFETs and IGBTs |
| SC14 | RF Wideband Transistors |
| SC15 | Microwave Transistors (new version planned) |
| SC16 | Wideband Hybrid IC Modules |
| SC17 | Semiconductor Sensors |

### Professional Components

| | |
|---|---|
| PC06 | Circulators and Isolators |

**MORE INFORMATION FROM PHILIPS SEMICONDUCTORS?**
For more information about Philips Semiconductors data handbooks, catalogs and subscriptions, contact your nearest Philips Semiconductors national organization, select from the **address list on the back cover of this handbook.** Product specialists are at your service and inquiries are answered promptly.

# Data handbook system

## OVERVIEW OF PHILIPS COMPONENTS DATA HANDBOOKS

Our sister product division, Philips Components, also has a comprehensive data handbook system to support their products. Their data handbook titles are listed here.

### Display Components

| *Book* | *Title* |
|---|---|
| DC01 | Colour TV Picture Tubes and Assemblies<br>Colour Monitor Tubes |
| DC02 | Monochrome Monitor Tubes and Deflection Units |
| DC03 | Television Tuners, Coaxial Aerial Input Assemblies |
| DC05 | Flyback Transformers, Mains Transformers and General-purpose FXC Assemblies |

### Magnetic Products

| | |
|---|---|
| MA01 | Soft Ferrites |
| MA03 | Piezoelectric Ceramics<br>Specialty Ferrites |
| MA04 | Dry-reed Switches |

### Passive Components

| | |
|---|---|
| PA01 | Electrolytic Capacitors |
| PA02 | Varistors, Thermistors and Sensors |
| PA03 | Potentiometers |
| PA04 | Variable Capacitors |
| PA05 | Film Capacitors |
| PA06 | Ceramic Capacitors |
| PA07 | Quartz Crystals for Special and Industrial Applications |
| PA08 | Fixed Resistors |
| PA10 | Quartz Crystals for Automotive and Standard Applications |
| PA11 | Quartz Oscillators |

### Professional Components

| | |
|---|---|
| PC04 | Photo Multipliers |
| PC05 | Plumbicon Camera Tubes and Accessories |
| PC07 | Vidicon and Newvicon Camera Tubes and Deflection Units |
| PC08 | Image Intensifiers |
| PC12 | Electron Multipliers |

## MORE INFORMATION FROM PHILIPS COMPONENTS?

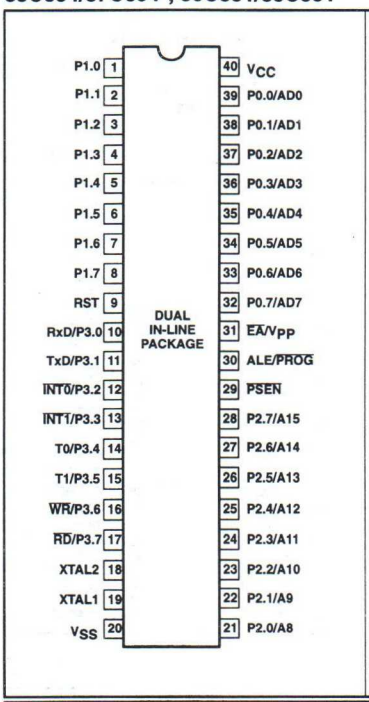
For more information contact your nearest Philips Components national organizaiton shown in the following list.

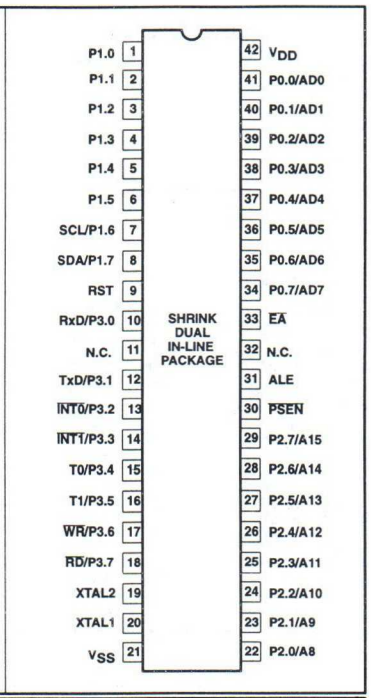
**Argentina:** BUENOS AIRES, Tel. (541)786 7635, Fax . (541)786 9367.
**Australia:** NORTH RYDE, Tel. (02)9805 4455, Fax. (02)9805 4466.
**Austria:** WIEN, Tel. (01)60101 1820, Fax. (01)601 01 12 12.
**Belgium:** NL EINDHOVEN, Tel. (31)40 2783 749, Fax. (31)40 2788 399.
**Brazil:** SÃO PAULO, Tel. (011)821 2333, Fax (011)829 1849.
**Canada:** SCARBOROUGH, Tel. (0416)292 5161, Fax. (0416)754 6248.
**Chile:** SANTIAGO, Tel. (02)77 38 16, Fax. (02)735 3594.
**China** (Peoples Republic of)**:** SHANGHAI, Tel. (21)326 4141, Fax. (21)320 2160.
**Columbia:** BOGOTA, Tel. (571)248 5571, Fax (571)217 4549.
**Denmark:** COPENHAGEN, Tel. (032)883 333, Fax. (031)571 949.
**Finland:** ESPOO, Tel. (9)0-5615 800, Fax. (9)0-615 80920.
**France:** SURESNES, Tel. (01)4099 6161, Fax, (01)4099 6431.
**Germany:** HAMBURG, Tel. (040)3296-0, Fax. (040)3296 213.
**Greece:** TAVROS, Tel. (01)489 4339/(01)489 4911, Fax. (01)481 5180.
**Hong Kong:** KWAI CHUNG, Tel. (852)2784 3000, Fax. (852)2784 3003.
**India:** BOMBAY, Tel. (022)4938 541, Fax. (022)4938 722.
**Indonesia:** JAKARTA, Tel. (021)5201122. Fax. (021)5205189.
**Ireland:** DUBLIN, Tel. (01)76 40 203, Fax. (01)76 40 210.
**Israel:** TEL AVIV, Tel (03)6450 444, Fax. (03)491 007.
**Italy:** MILANO, Tel. (02)6752 2531, Fax. (02)6752 2557.
**Japan:** TOKIO, Tel. (03)3740 5143, Fax. (03)3740 5035.
**Korea** (Republic of)**:** SEOUL, Tel. (02)709-1412, Fax. (02)709-1479.
**Malaysia:** KUALA LUMPUR, Tel. (03)757 5511, Fax. (03)757 4880.
**Mexico:** EL PASO, Tel. (915)772 4020, Fax. (915)772 4332.
**Netherlands:** EINDHOVEN, Tel. (040)2783 749, Fax. (040)2788 399.
**New Zealand:** AUKLAND, Tel. (09)849-4160, Fax. (09)849-7811.
**Norway:** OSLO, Tel. (22)74 8000, Fax (22)74 8341.
**Pakistan:** KARACHI, Tel. (021)587 4641-49, Fax. (021)577035/5874546.
**Philippines:** MANILA, Tel. (02)810-0161, Fax. (02)817-3474.
**Portugal:** LINDA-A-VELHA, Tel. (01)4163160/4163333, Fax. (01)4163174/4163366.
**Singapore:** SINGAPORE, Tel. (65)350 2000, Fax. (65)355 1758.
**South Africa:** JOHANNESBURG, Tel. (011)470-5911, Fax. (011)470-5494.
**Spain:** BARCELONA, Tel. (93)301 6312, Fax. (93)301 4243.
**Sweden:** STOCKHOLM, Tel. (08)632 2000, Fax. (08)632 2745.
**Switzerland:** ZÜRICH, Tel. (01)488 2211, Fax. (01) 481 7730.
**Taiwan:** TAIPEI, Tel. (02)388 7666, Fax. (02)382 4382.
**Thailand:** BANGKOK, Tel. (66)2 745 4090, Fax. (66)2 398 0793.
**Turkey:** ISTANBUL, Tel. (0212)279 2770, Fax. (0212)282 6707.
**United Kingdom:** DORKING, Surrey, Tel. (01306)512000, Fax. (01306)512345.
**United States:** JUPITER, Tel. (407)744 4200, Fax, (407)743 2113.
**Uruguay:** MONTEVIDEO, Tel. (02)704 044, Fax (02)920 601.
**Internet:** http://www.semiconductors.philips.com/ps/

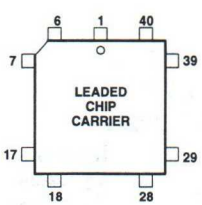
For all other countries apply to:
**Philips Components.**
Marketing Communications,
P.O. Box 218,
5600 MD, EINDHOVEN, The Netherlands
Telex 35000 phtcnl, Fax. +31–40–2724547.

# Appendix A

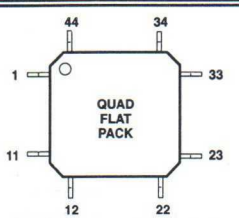
## 80C31/80C51/87C51, 83C508/87C508, 80C652/83C652/87C652*, 83C654/87C654*, 80C851/83C851




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC | 23 | NC |
| 2 | P1.0 | 24 | P2.0/A8 |
| 3 | P1.1 | 25 | P2.1/A9 |
| 4 | P1.2 | 26 | P2.2/A10 |
| 5 | P1.3 | 27 | P2.3/A11 |
| 6 | P1.4 | 28 | P2.4/A12 |
| 7 | P1.5 | 29 | P2.5/A13 |
| 8 | P1.6 | 30 | P2.6/A14 |
| 9 | P1.7 | 31 | P2.7/A15 |
| 10 | RST | 32 | PSEN |
| 11 | P3.0/RxD | 33 | ALE/PROG |
| 12 | NC | 34 | NC |
| 13 | P3.1/TxD | 35 | EA/VPP |
| 14 | P3.2/INT0 | 36 | P0.7/AD7 |
| 15 | P3.3/INT1 | 37 | P0.6/AD6 |
| 16 | P3.4/T0 | 38 | P0.5/AD5 |
| 17 | P3.5/T1 | 39 | P0.4/AD4 |
| 18 | P3.6/WR | 40 | P0.3/AD3 |
| 19 | P3.7/RD | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | VSS | 44 | VCC |



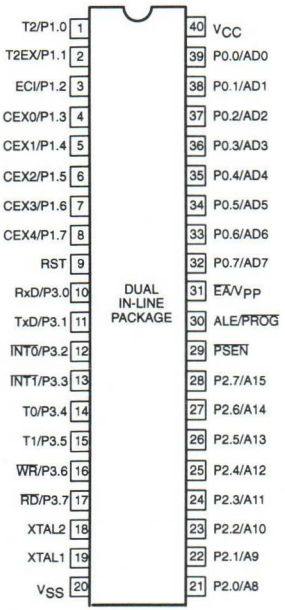

| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5 | 23 | P2.5/A13 |
| 2 | P1.6 | 24 | P2.6/A14 |
| 3 | P1.7 | 25 | P2.7/A15 |
| 4 | RST | 26 | PSEN |
| 5 | P3.0/RxD | 27 | ALE/PROG |
| 6 | NC | 28 | NC |
| 7 | P3.1/TxD | 29 | EA/VPP |
| 8 | P3.2/INT0 | 30 | P0.7/AD7 |
| 9 | P3.3/INT1 | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/WR | 34 | P0.3/AD3 |
| 13 | P3.7/RD | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | VSS | 38 | VCC |
| 17 | NC | 39 | NC |
| 18 | P2.0/A8 | 40 | P1.0 |
| 19 | P2.1/A9 | 41 | P1.1 |
| 20 | P2.2/A10 | 42 | P1.2 |
| 21 | P2.3/A11 | 43 | P1.3 |
| 22 | P2.4/A12 | 44 | P1.4 |

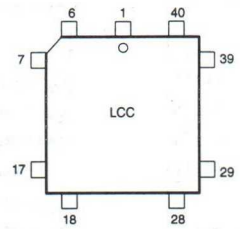
* P1.6 and P1.7 have the alternate functions SCL and SDA, respectively, on the 80C652/83C652/87C652 and 83C654/87C654.

**NOTES TO QFP ONLY:**

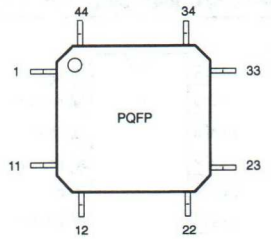
1. Due to EMC improvements, it is advised to connect pins 6, 16, 28, 39 to $V_{SS}$ on the 80C652/83C652, and 83C654.

## 83C51FA/87C51FA, 83C51FB/87C51FB, 83C51FC/87C51FC, 87L51FA/87L51FB, 80C58/87C58



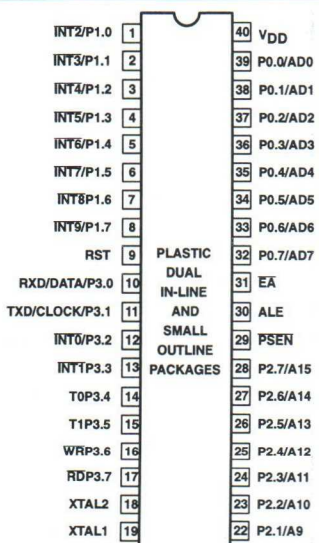



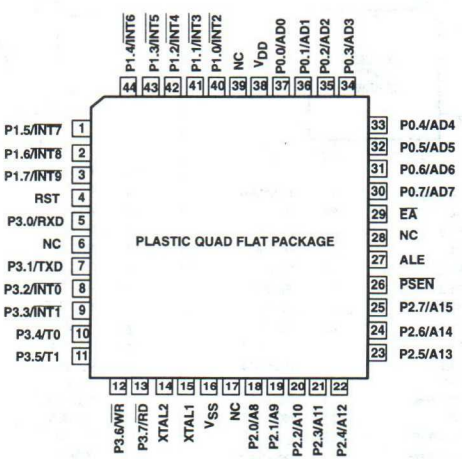

| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC | 23 | NC |
| 2 | P1.0/T2 | 24 | P2.0/A8 |
| 3 | P1.1/T2EX | 25 | P2.1/A9 |
| 4 | P1.2/ECI | 26 | P2.2/A10 |
| 5 | P1.3/CEX0 | 27 | P2.3/A11 |
| 6 | P1.4/CEX1 | 28 | P2.4/A12 |
| 7 | P1.5/CEX2 | 29 | P2.5/A13 |
| 8 | P1.6/CEX3 | 30 | P2.6/A14 |
| 9 | P1.7/CEX4 | 31 | P2.7/A15 |
| 10 | RST | 32 | PSEN |
| 11 | P3.0/RxD | 33 | ALE/PROG |
| 12 | NC | 34 | NC |
| 13 | P3.1/TxD | 35 | EA/VPP |
| 14 | P3.2/INT0 | 36 | P0.7/AD7 |
| 15 | P3.3/INT1 | 37 | P0.6/AD6 |
| 16 | P3.4/T0 | 38 | P0.5/AD5 |
| 17 | P3.5/T1 | 39 | P0.4/AD4 |
| 18 | P3.6/WR | 40 | P0.3/AD3 |
| 19 | P3.7/RD | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | VSS | 44 | VCC |



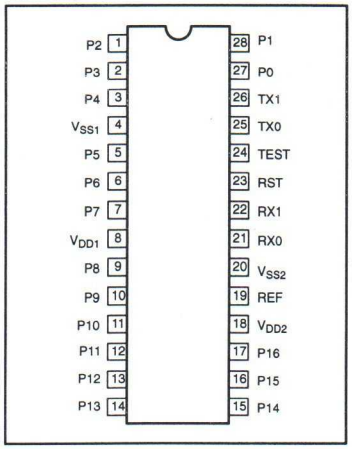

| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5/CEX2 | 23 | P2.5/A13 |
| 2 | P1.6/CEX3 | 24 | P2.6/A14 |
| 3 | P1.7/CEX4 | 25 | P2.7/A15 |
| 4 | RST | 26 | PSEN |
| 5 | P3.0/RxD | 27 | ALE/PROG |
| 6 | NC | 28 | NC |
| 7 | P3.1/TxD | 29 | EA/VPP |
| 8 | P3.2/INT0 | 30 | P0.7/AD7 |
| 9 | P3.3/INT1 | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/WR | 34 | P0.3/AD3 |
| 13 | P3.7/RD | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | VSS | 38 | VCC |
| 17 | NC | 39 | NC |
| 18 | P2.0/A8 | 40 | P1.0/T2 |
| 19 | P2.1/A9 | 41 | P1.1/T2EX |
| 20 | P2.2/A10 | 42 | P1.2/ECI |
| 21 | P2.3/A11 | 43 | P1.3/CEX0 |
| 22 | P2.4/A12 | 44 | P1.4/CEX1 |

## 80CL31/80CL51

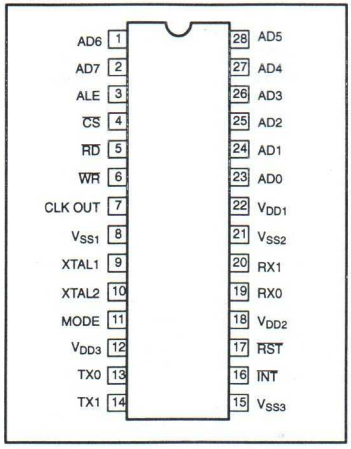


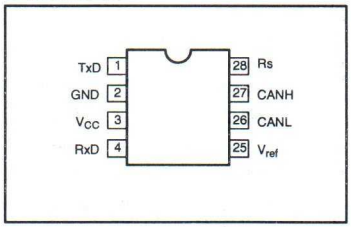
## 82C150



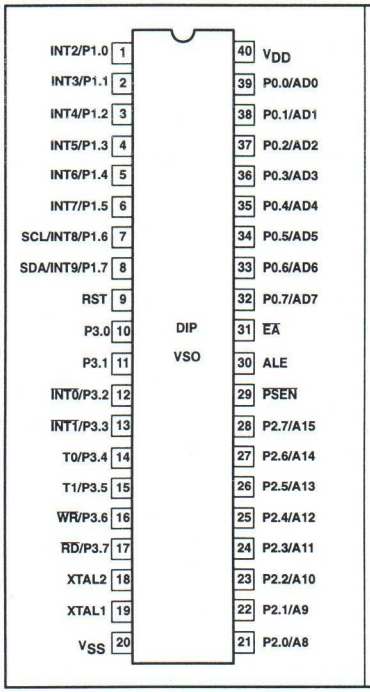

## 82C200

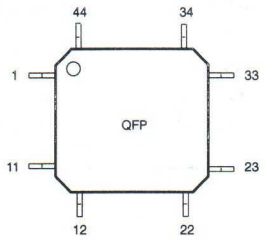



## PCA82C250




## 80CL410/83CL410*

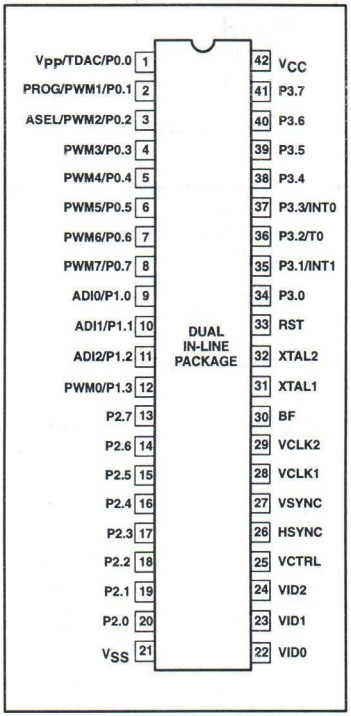


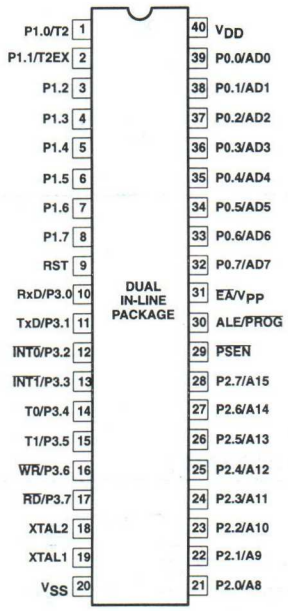

| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5/INT7 | 23 | P2.5/A13 |
| 2 | P1.6/INT8/SCL | 24 | P2.6/A14 |
| 3 | P1.7/INT9/SDA | 25 | P2.7/A15 |
| 4 | RST | 26 | PSEN |
| 5 | P3.0 | 27 | ALE |
| 6 | NC | 28 | NC |
| 7 | P3.1 | 29 | EA |
| 8 | P3.2/INT0 | 30 | P0.7/AD7 |
| 9 | P3.3/INT1 | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/WR | 34 | P0.3/AD3 |
| 13 | P3.7/RD | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | VSS | 38 | VDD |
| 17 | NC | 39 | NC |
| 18 | P2.0/A8 | 40 | P1.0/INT2 |
| 19 | P2.1/A9 | 41 | P1.1/INT3 |
| 20 | P2.2/A10 | 42 | P1.2/INT4 |
| 21 | P2.3/A11 | 43 | P1.3/INT5 |
| 22 | P2.4/A12 | 44 | P1.4/INT6 |

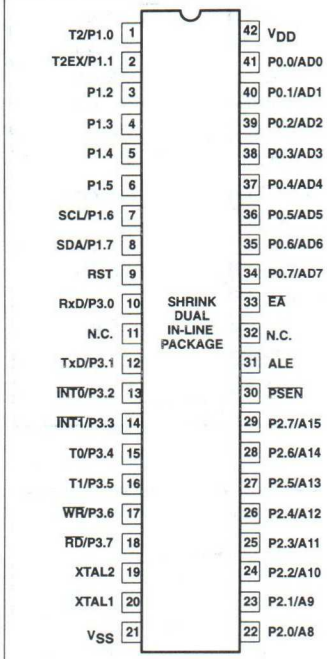
* P1.6 and P1.7 have the alternate functions SCL and SDA, respectively, on the 80CL410/83CL410.

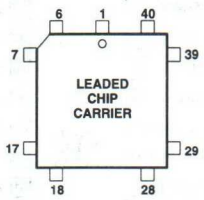
## 83C055/87C055



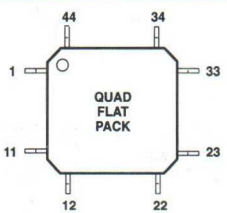
## 80C32/80C52/87C52, 80C54/87C54, 83C524/87C524*, 80C528/83C528/87C528*



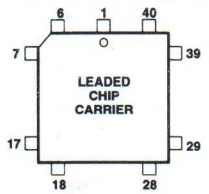

| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC | 23 | NC |
| 2 | T2/P1.0 | 24 | P2.0/A8 |
| 3 | T2EX/P1.1 | 25 | P2.1/A9 |
| 4 | P1.2 | 26 | P2.2/A10 |
| 5 | P1.3 | 27 | P2.3/A11 |
| 6 | P1.4 | 28 | P2.4/A12 |
| 7 | P1.5 | 29 | P2.5/A13 |
| 8 | P1.6 | 30 | P2.6/A14 |
| 9 | P1.7 | 31 | P2.7/A15 |
| 10 | RST | 32 | PSEN |
| 11 | RxD/P3.0 | 33 | ALE/PROG |
| 12 | NC | 34 | NC |
| 13 | TxD/P3.1 | 35 | EA/VPP |
| 14 | INT0/P3.2 | 36 | P0.7/AD7 |
| 15 | INT1/P3.3 | 37 | P0.6/AD6 |
| 16 | T0/P3.4 | 38 | P0.5/AD5 |
| 17 | T1/P3.5 | 39 | P0.4/AD4 |
| 18 | WR/P3.6 | 40 | P0.3/AD3 |
| 19 | RD/P3.7 | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | VSS | 44 | VCC |



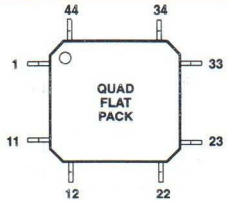

| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5 | 23 | P2.5/A13 |
| 2 | P1.6 | 24 | P2.6/A14 |
| 3 | P1.7 | 25 | P2.7/A15 |
| 4 | RST | 26 | PSEN |
| 5 | RxD/P3.0 | 27 | ALE/PROG |
| 6 | NC | 28 | NC |
| 7 | TxD/P3.1 | 29 | EA/VPP |
| 8 | INT0/P3.2 | 30 | P0.7/AD7 |
| 9 | INT1/P3.3 | 31 | P0.6/AD6 |
| 10 | T0/P3.4 | 32 | P0.5/AD5 |
| 11 | T1/P3.5 | 33 | P0.4/AD4 |
| 12 | WR/P3.6 | 34 | P0.3/AD3 |
| 13 | RD/P3.7 | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | VSS | 38 | VCC |
| 17 | NC | 39 | NC |
| 18 | P2.0/A8 | 40 | T2/P1.0 |
| 19 | P2.1/A9 | 41 | T2EX/P1.1 |
| 20 | P2.2/A10 | 42 | P1.2 |
| 21 | P2.3/A11 | 43 | P1.3 |
| 22 | P2.4/A12 | 44 | P1.4 |

* P1.6 and P1.7 have the alternate functions SCL and SDA, respectively, on the 83C524/87C524 and 80C528/83C528/87C528.

## P8XCE528

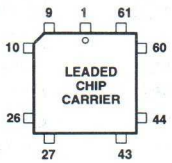



| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | $V_{SS3}$ | 23 | $V_{DD1}$ |
| 2 | P1.0/T2 | 24 | P2.0/A8 |
| 3 | P1.1/T2EX | 25 | P2.1/A9 |
| 4 | P1.2 | 26 | P2.2/A10 |
| 5 | P1.3 | 27 | P2.3/A11 |
| 6 | P1.4 | 28 | P2.4/A12 |
| 7 | P1.5 | 29 | P2.5/A13 |
| 8 | P1.6/SCL | 30 | P2.6/A14 |
| 9 | P1.7/SDA | 31 | P2.7/A15 |
| 10 | RST | 32 | $\overline{PSEN}$ |
| 11 | P3.0/data/RxD | 33 | ALE/$\overline{WE}$ |
| 12 | $V_{SS4}$ | 34 | $V_{SS2}$ |
| 13 | P3.1/clock/TxD | 35 | $\overline{EA}$ |
| 14 | P3.2/$\overline{INT0}$ | 36 | P0.7/AD7 |
| 15 | P3.3/$\overline{INT1}$ | 37 | P0.6/AD6 |
| 16 | P3.4/T0 | 38 | P0.5/AD5 |
| 17 | P3.5/T1 | 39 | P0.4/AD4 |
| 18 | P3.6/$\overline{WR}$ | 40 | P0.3/AD3 |
| 19 | P3.7/$\overline{RD}$ | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | $V_{SS1}$ | 44 | $V_{DD2}$ |



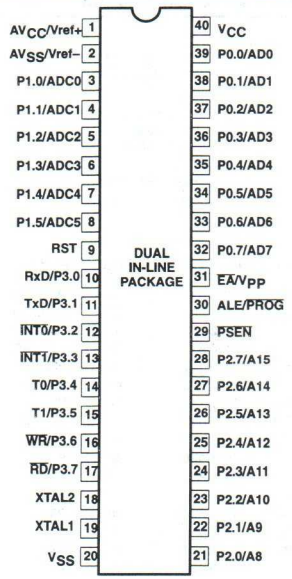

| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5 | 23 | P2.5/A13 |
| 2 | P1.6/SCL | 24 | P2.6/A14 |
| 3 | P1.7/SDA | 25 | P2.7/A15 |
| 4 | RST | 26 | $\overline{PSEN}$ |
| 5 | P3.0/data/RxD | 27 | ALE |
| 6 | $V_{SS4}$ | 28 | $V_{SS2}$ |
| 7 | P3.1/clock/TxD | 29 | $\overline{EA}$ |
| 8 | P3.2/$\overline{INT0}$ | 30 | P0.7/AD7 |
| 9 | P3.3/$\overline{INT1}$ | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/$\overline{WR}$ | 34 | P0.3/AD3 |
| 13 | P3.7/$\overline{RD}$ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | $V_{SS1}$ | 38 | $V_{DD2}$ |
| 17 | $V_{DD1}$ | 39 | $V_{SS3}$ |
| 18 | P2.0/A8 | 40 | P1.0/T2 |
| 19 | P2.1/A9 | 41 | P1.1/T2EX |
| 20 | P2.2/A10 | 42 | P1.2 |
| 21 | P2.3/A11 | 43 | P1.3 |
| 22 | P2.4/A12 | 44 | P1.4 |

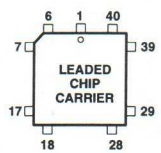
## 80C451/83C451/87C451, 80C453/83C453/87C453




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | $\overline{EA}$/$V_{PP}$ | 35 | RST |
| 2 | P2.0/A8 | 36 | P3.0/RxD |
| 3 | P2.1/A9 | 37 | P3.1/TxD |
| 4 | P2.2/A10 | 38 | P3.2/$\overline{INT0}$ |
| 5 | P2.3/A11 | 39 | P3.3/$\overline{INT1}$ |
| 6 | P2.4/A12 | 40 | P3.4/T0 |
| 7 | P2.5/A13 | 41 | P3.5/T1 |
| 8 | P2.6/A14 | 42 | P3.6/$\overline{WR}$ |
| 9 | P2.7/A15 | 43 | P3.7/$\overline{RD}$ |
| 10 | P0.7/AD7 | 44 | P5.0 |
| 11 | P0.6/AD6 | 45 | P5.1 |
| 12 | P0.5/AD5 | 46 | P5.2 |
| 13 | P0.4/AD4 | 47 | P5.3 |
| 14 | P0.3/AD3 | 48 | P5.4 |
| 15 | P0.2/AD2 | 49 | P5.5 |
| 16 | P0.1/AD1 | 50 | P5.6 |
| 17 | P0.0/AD0 | 51 | P5.7 |
| 18 | $V_{CC}$ | 52 | XTAL2 |
| 19 | P4.7 | 53 | XTAL1 |
| 20 | P4.6 | 54 | $V_{SS}$ |
| 21 | P4.5 | 55 | $\overline{ODS}$ |
| 22 | P4.4 | 56 | $\overline{IDS}$ |
| 23 | P4.3 | 57 | BFLAG |
| 24 | P4.2 | 58 | AFLAG |
| 25 | P4.1 | 59 | P6.0 |
| 26 | P4.0 | 60 | P6.1 |
| 27 | P1.0 | 61 | P6.2 |
| 28 | P1.1 | 62 | P6.3 |
| 29 | P1.2 | 63 | P6.4 |
| 30 | P1.3 | 64 | P6.5 |
| 31 | P1.4 | 65 | P6.6 |
| 32 | P1.5 | 66 | P6.7 |
| 33 | P1.6 | 67 | $\overline{PSEN}$ |
| 34 | P1.7 | 68 | ALE/$\overline{PROG}$ |

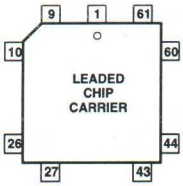
## 80C550/83C550/87C550



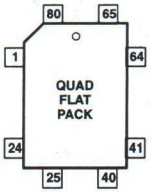




| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | $AV_{CC}$ | 16 | P3.2/$\overline{INT0}$ | 31 | P2.6/A14 |
| 2 | Vref+ | 17 | P3.3/$\overline{INT1}$ | 32 | P2.7/A15 |
| 3 | Vref– | 18 | P3.4/T0 | 33 | $\overline{PSEN}$ |
| 4 | $AV_{SS}$ | 19 | P3.5/T1 | 34 | ALE/$\overline{PROG}$ |
| 5 | P1.0/ADC0 | 20 | P3.6/$\overline{WR}$ | 35 | $\overline{EA}/V_{PP}$ |
| 6 | P1.1/ADC1 | 21 | P3.7/$\overline{RD}$ | 36 | P0.7/AD7 |
| 7 | P1.2/ADC2 | 22 | XTAL2 | 37 | P0.6/AD6 |
| 8 | P1.3/ADC3 | 23 | XTAL1 | 38 | P0.5/AD5 |
| 9 | P1.4/ADC4 | 24 | $V_{SS}$ | 39 | P0.4/AD4 |
| 10 | P1.5/ADC5 | 25 | P2.0/A8 | 40 | P0.3/AD3 |
| 11 | P1.6/ADC6 | 26 | P2.1/A9 | 41 | P0.2/AD2 |
| 12 | P1.7/ADC7 | 27 | P2.2/A10 | 42 | P0.1/AD1 |
| 13 | RST | 28 | P2.3/A11 | 43 | P0.0/AD0 |
| 14 | P3.0/RxD | 29 | P2.4/A12 | 44 | $V_{CC}$ |
| 15 | P3.1/TxD | 30 | P2.5/A13 | | |

## 80C552/83C552/87C552*, 80C562/83C562




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P5.0/ADC0 | 35 | XTAL1 |
| 2 | $V_{DD}$ | 36 | $V_{SS}$ |
| 3 | STADC | 37 | $V_{SS}$ |
| 4 | $\overline{PWM0}$ | 38 | NC |
| 5 | $\overline{PWM1}$ | 39 | P2.0/A08 |
| 6 | $\overline{EW}$ | 40 | P2.1/A09 |
| 7 | P4.0/CMSR0 | 41 | P2.2/A10 |
| 8 | P4.1/CMSR1 | 42 | P2.3/A11 |
| 9 | P4.2/CMSR2 | 43 | P2.4/A12 |
| 10 | P4.3/CMSR3 | 44 | P2.5/A13 |
| 11 | P4.4/CMSR4 | 45 | P2.6/A14 |
| 12 | P4.5/CMSR5 | 46 | P2.7/A15 |
| 13 | P4.6/CMT0 | 47 | $\overline{PSEN}$ |
| 14 | P4.7/CMT1 | 48 | ALE/$\overline{PROG}$ |
| 15 | RST | 49 | $\overline{EA}/V_{PP}$ |
| 16 | P1.0/CT0I | 50 | P0.7/AD7 |
| 17 | P1.1/CT1I | 51 | P0.6/AD6 |
| 18 | P1.2/CT2I | 52 | P0.5/AD5 |
| 19 | P1.3/CT3I | 53 | P0.4/AD4 |
| 20 | P1.4/T2 | 54 | P0.3/AD3 |
| 21 | P1.5/RT2 | 55 | P0.2/AD2 |
| 22 | P1.6/SCL | 56 | P0.1/AD1 |
| 23 | P1.7/SDA | 57 | P0.0/AD0 |
| 24 | P3.0/RxD | 58 | AVref– |
| 25 | P3.1/TxD | 59 | AVref+ |
| 26 | P3.2/$\overline{INT0}$ | 60 | $AV_{SS}$ |
| 27 | P3.3/$\overline{INT1}$ | 61 | $AV_{DD}$ |
| 28 | P3.4/T0 | 62 | P5.7/ADC7 |
| 29 | P3.5/T1 | 63 | P5.6/ADC6 |
| 30 | P3.6/$\overline{WR}$ | 64 | P5.5/ADC5 |
| 31 | P3.7/$\overline{RD}$ | 65 | P5.4/ADC4 |
| 32 | NC | 66 | P5.3/ADC3 |
| 33 | NC | 67 | P5.2/ADC2 |
| 34 | XTAL2 | 68 | P5.1/ADC1 |




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P4.1/CMSR1 | 41 | P2.3/A11 |
| 2 | P4.2/CMSR2 | 42 | P2.4/A12 |
| 3 | NC | 43 | NC |
| 4 | P4.3/CMSR3 | 44 | NC |
| 5 | P4.4/CMSR4 | 45 | P2.5/A13 |
| 6 | P4.5/CMSR5 | 46 | P2.6/A14 |
| 7 | P4.6/CMT0 | 47 | P2.7/A15 |
| 8 | P4.7/CMT1 | 48 | $\overline{PSEN}$ |
| 9 | RST | 49 | ALE/$\overline{PROG}$ |
| 10 | P1.0/CT0I | 50 | $\overline{EA}/V_{PP}$ |
| 11 | P1.1/CT1I | 51 | P0.7/AD7 |
| 12 | P1.2/CT2I | 52 | P0.6/AD6 |
| 13 | P1.3/CT3I | 53 | P0.5/AD5 |
| 14 | P1.4/T2 | 54 | P0.4/AD4 |
| 15 | P1.5/RT2 | 55 | P0.3/AD3 |
| 16 | P1.6/SCL | 56 | P0.2/AD2 |
| 17 | P1.7/SDA | 57 | P0.1/AD1 |
| 18 | P3.0/RxD | 58 | P0.0/AD0 |
| 19 | P3.1/TxD | 59 | AVref– |
| 20 | P3.2/$\overline{INT0}$ | 60 | AVref+ |
| 21 | NC | 61 | $AV_{SS}$ |
| 22 | NC | 62 | NC |
| 23 | P3.3/$\overline{INT1}$ | 63 | $AV_{DD}$ |
| 24 | P3.4/T0 | 64 | P5.7/ADC7 |
| 25 | P3.5/T1 | 65 | P5.6/ADC6 |
| 26 | P3.6/$\overline{WR}$ | 66 | P5.5/ADC5 |
| 27 | P3.7/$\overline{RD}$ | 67 | P5.4/ADC4 |
| 28 | NC | 68 | P5.3/ADC3 |
| 29 | NC | 69 | P5.2/ADC2 |
| 30 | NC | 70 | P5.1/ADC1 |
| 31 | XTAL2 | 71 | P5.0/ADC0 |
| 32 | XTAL1 | 72 | $V_{DD}$ |
| 33 | IC | 73 | IC |
| 34 | $V_{SS}$ | 74 | STADC |
| 35 | $V_{SS}$ | 75 | $\overline{PWM0}$ |
| 36 | $V_{SS}$ | 76 | $\overline{PWM1}$ |
| 37 | NC | 77 | $\overline{EW}$ |
| 38 | P2.0/A08 | 78 | NC |
| 39 | P2.1/A09 | 79 | NC |
| 40 | P2.2/A10 | 80 | P4.0/CMSR0 |

NC = not connected
IC = internally connected (do not use)

* P1.6 and P1.7 have the alternate functions SCL and SDA, respectively, on the 80C552/83C552/87C552.

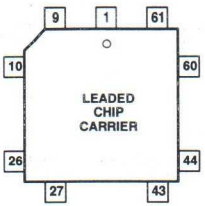
## 80CE558/83CE558/89CE558, P83CE559/P80CE559/P89CE559




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | AVref– | 41 | P3.0/RXD |
| 2 | AVref+ | 42 | P3.1/TXD |
| 3 | $AV_{SS1}$ | 43 | P3.2/$\overline{INT0}$ |
| 4 | $AV_{DD1}$ | 44 | P3.3/$\overline{INT1}$ |
| 5 | P5.7/ADC7 | 45 | P3.4/T0 |
| 6 | P5.6/ADC6 | 46 | P3.5/T1 |
| 7 | P5.5/ADC5 | 47 | P3.6/$\overline{WR}$ |
| 8 | P5.4/ADC4 | 48 | P3.7/$\overline{RD}$ |
| 9 | P5.3/ADC3 | 49 | NC |
| 10 | P5.2/ADC2 | 50 | NC |
| 11 | P5.1/ADC1 | 51 | XTAL2 |
| 12 | P5.0/ADC0 | 52 | XTAL1 |
| 13 | $V_{SS1}$ | 53 | $V_{DD3}$ |
| 14 | $V_{DD1}$ | 54 | $V_{SS3}$ |
| 15 | ADEXS | 55 | P2.0/A8 |
| 16 | $\overline{PWM0}$ | 56 | P2.1/A9 |
| 17 | $\overline{PWM1}$ | 57 | P2.2/A10 |
| 18 | $\overline{EW}$ | 58 | P2.3/A11 |
| 19 | P4.0/CMSR0 | 59 | P2.4/A12 |
| 20 | P4.1/CMSR1 | 60 | P2.5/A13 |
| 21 | P4.2/CMSR2 | 61 | P2.6/A14 |
| 22 | P4.3/CMSR3 | 62 | P2.7/A15 |
| 23 | RSTOUT | 63 | $\overline{PSEN}$ |
| 24 | P4.4/CMSR4 | 64 | ALE/$\overline{WE}$ * |
| 25 | P4.5/CMSR5 | 65 | $\overline{EA}$ |
| 26 | P4.6/CMT0 | 66 | $V_{DD4}$ |
| 27 | P4.7/CMT1 | 67 | $V_{SS4}$ |
| 28 | $V_{DD2}$ | 68 | P0.7/AD7 |
| 29 | $V_{SS2}$ | 69 | P0.6/AD6 |
| 30 | RSTIN | 70 | P0.5/AD5 |
| 31 | P1.0/CT0I/INT2 | 71 | P0.4/AD4 |
| 32 | P1.1/CT1I/INT3 | 72 | P0.3/AD3 |
| 33 | P1.2/CT2I/INT4 | 73 | P0.2/AD2 |
| 34 | P1.3/CT3I/INT5 | 74 | P0.1/AD1 |
| 35 | P1.4/T2 | 75 | P0.0/AD0 |
| 36 | P1.5/RT2 | 76 | $AV_{DD2}$ |
| 37 | P1.6 | 77 | $AV_{SS2}$ |
| 38 | P1.7 | 78 | XTAL3 |
| 39 | SCL | 79 | XTAL4 |
| 40 | SDA | 80 | SELXTAL1 |

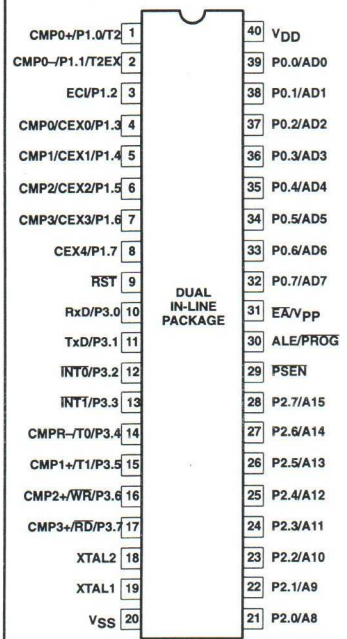
* only 89CE558/89CE559 with alternative function

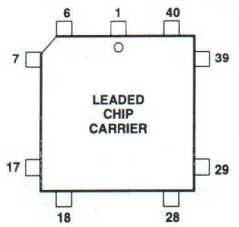
## 80C592/83C592/87C592



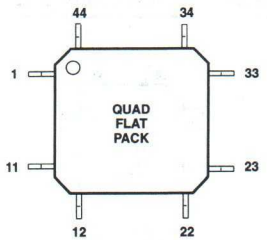

| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P5.0/ADC0 | 35 | $V_{SS}$ |
| 2 | $V_{DD}$ | 36 | P2.0/A08 |
| 3 | STADC | 37 | P2.1/A09 |
| 4 | $\overline{PWM0}$ | 38 | P2.2/A10 |
| 5 | $\overline{PWM1}$ | 39 | P2.3/A11 |
| 6 | $\overline{EW}$ | 40 | P2.4/A12 |
| 7 | P4.0/CMSR0 | 41 | P2.5/A13 |
| 8 | P4.1/CMRS1 | 42 | P2.6/A14 |
| 9 | P4.2CMSR2 | 43 | P2.7/A15 |
| 10 | P4.3/CMSR3 | 44 | $\overline{PSEN}$ |
| 11 | P4.4/CMSR4 | 45 | ALE/$\overline{PROG}$ |
| 12 | P4.5/CMSR5 | 46 | $\overline{EA}$/$V_{PP}$ |
| 13 | P4.6/CMT0 | 47 | P0.7/AD07 |
| 14 | P4.7/CMT1 | 48 | P0.6/AD06 |
| 15 | RST | 49 | P0.5/AD05 |
| 16 | P1.0/CT0I/INT2 | 50 | P0.4/AD04 |
| 17 | P1.1/CT1I/INT3 | 51 | P0.3/AD03 |
| 18 | P1.2/CT2I/INT4 | 52 | P0.2/AD02 |
| 19 | P1.3/CT3I/INT5 | 53 | P0.1/AD01 |
| 20 | P1.4/T2 | 54 | P0.0/AD00 |
| 21 | P1.5/RT2 | 55 | REF |
| 22 | $CV_{SS}$ | 56 | CRX1 |
| 23 | P1.6/CTX0 | 57 | CRX0 |
| 24 | P1.7/CTX1 | 58 | AVref– |
| 25 | P3.0/RxD | 59 | AVref+ |
| 26 | P3.1/TxD | 60 | $AV_{SS}$ |
| 27 | P3.2/$\overline{INT0}$ | 61 | $AV_{DD}$ |
| 28 | P3.3/$\overline{INT1}$ | 62 | P5.7/ADC7 |
| 29 | P3.4/T0 | 63 | P5.6/ADC6 |
| 30 | P3.5/T1 | 64 | P5.5/ADC5 |
| 31 | P3.6/$\overline{WR}$ | 65 | P5.4/ADC4 |
| 32 | P3.7/$\overline{RD}$ | 66 | P5.3/ADC3 |
| 33 | XTAL2 | 67 | P5.2/ADC2 |
| 34 | XTAL1 | 68 | P5.1/ADC1 |

## 80C575/83C575/87C575



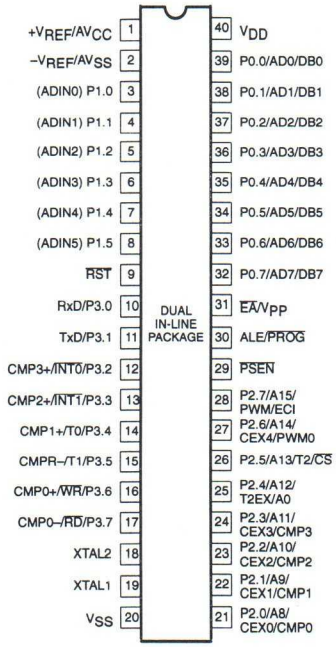



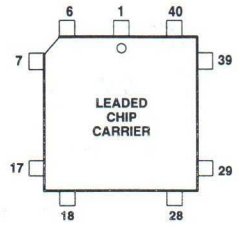

| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC | 23 | NC |
| 2 | T2/P1.0/CMP0+ | 24 | P2.0/A8 |
| 3 | T2EX/P1.1/CMP0– | 25 | P2.1/A9 |
| 4 | P1.2/ECI | 26 | P2.2/A10 |
| 5 | P1.3/CMP0/CEX0 | 27 | P2.3/A11 |
| 6 | P1.4/CMP1/CEX1 | 28 | P2.4/A12 |
| 7 | P1.5/CMP2/CEX2 | 29 | P2.5/A13 |
| 8 | P1.6/CMP3/CEX3 | 30 | P2.6/A14 |
| 9 | P1.7/CEX4 | 31 | P2.7/A15 |
| 10 | RST | 32 | PSEN |
| 11 | RxD/P3.0 | 33 | ALE/PROG |
| 12 | NC | 34 | NC |
| 13 | TxD/P3.1 | 35 | EA/VPP |
| 14 | INT0/P3.2 | 36 | P0.7/AD7 |
| 15 | INT1/P3.3 | 37 | P0.6/AD6 |
| 16 | T0/P3.4/CMPR– | 38 | P0.5/AD5 |
| 17 | T1/P3.5/CMP1+ | 39 | P0.4/AD4 |
| 18 | WR/P3.6/CMP2+ | 40 | P0.3/AD3 |
| 19 | RD/P3.7/CMP3+ | 41 | P0.2/AD2 |
| 20 | XTAL2 | 42 | P0.1/AD1 |
| 21 | XTAL1 | 43 | P0.0/AD0 |
| 22 | VSS | 44 | VCC |



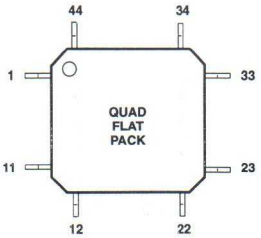

| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5/CMP2/CEX2 | 23 | P2.5/A13 |
| 2 | P1.6/CMP3/CEX3 | 24 | P2.6/A14 |
| 3 | P1.7/CEX4 | 25 | P2.7/A15 |
| 4 | RST | 26 | PSEN |
| 5 | RxD/P3.0 | 27 | ALE/PROG |
| 6 | NC | 28 | NC |
| 7 | TxD/P3.1 | 29 | EA/VPP |
| 8 | INT0/P3.2 | 30 | P0.7/AD7 |
| 9 | INT1/P3.3 | 31 | P0.6/AD6 |
| 10 | T0/P3.4/CMPR– | 32 | P0.5/AD5 |
| 11 | T1/P3.5/CMP1+ | 33 | P0.4/AD4 |
| 12 | WR/P3.6/CMP2+ | 34 | P0.3/AD3 |
| 13 | RD/P3.7/CMP3+ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | VSS | 38 | VCC |
| 17 | NC | 39 | NC |
| 18 | P2.0/A8 | 40 | T2/P1.0/CMP0+ |
| 19 | P2.1/A9 | 41 | T2EX/P1.1/CMP0– |
| 20 | P2.2/A10 | 42 | P1.2/ECI |
| 21 | P2.3/A11 | 43 | P1.3/CMP0/CEX0 |
| 22 | P2.4/A12 | 44 | P1.4/CMP1/CEX1 |

## 83C576/87C576



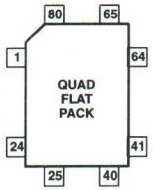




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | NC | 23 | NC |
| 2 | +VREF/AVCC | 24 | P2.0/A8/CEX0/CMP0 |
| 3 | –VREF/AVSS | 25 | P2.1/A9/CEX1/CMP1 |
| 4 | ADIN0/P1.0 | 26 | P2.2/A10/CEX2/CMP2 |
| 5 | ADIN1/P1.1 | 27 | P2.3/A11/CEX3/CMP3 |
| 6 | ADIN2/P1.2 | 28 | P2.4/A12/T2EX/A0 |
| 7 | ADIN3/P1.3 | 29 | P2.5/A13/T2/CS |
| 8 | ADIN4/P1.4 | 30 | P2.6/A14/CEX4/PWM0 |
| 9 | ADIN5/P1.5 | 31 | P2.7/A15/PWM1/ECI |
| 10 | RST | 32 | PSEN |
| 11 | RxD/P3.0 | 33 | ALE/PROG |
| 12 | NC | 34 | NC |
| 13 | TxD/P3.1 | 35 | EA/VPP |
| 14 | INT0/P3.2/CMP3+ | 36 | P0.7/AD7/DB7 |
| 15 | INT1/P3.3/CMP2+ | 37 | P0.6/AD6/DB6 |
| 16 | T0/P3.4/CMP1+ | 38 | P0.5/AD5/DB5 |
| 17 | T1/P3.5/CMPR– | 39 | P0.4/AD4/DB4 |
| 18 | WR/P3.6/CMP0+ | 40 | P0.3/AD3/DB3 |
| 19 | RD/P3.7/CMP0– | 41 | P0.2/AD2/DB2 |
| 20 | XTAL2 | 42 | P0.1/AD1/DB1 |
| 21 | XTAL1 | 43 | P0.0/AD0/DB0 |
| 22 | VSS | 44 | VCC |




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | ADIN3/P1.3 | 23 | P2.5/A13/T2/CS |
| 2 | ADIN4/P1.4 | 24 | P2.6/A14/CEX4/PWM0 |
| 3 | ADIN5/P1.5 | 25 | P2.7/A15/PWM1/ECI |
| 4 | RST | 26 | PSEN |
| 5 | RxD/P3.0 | 27 | ALE/PROG |
| 6 | NC | 28 | NC |
| 7 | TXD/P3.1 | 29 | EA/VPP |
| 8 | INT0/P3.2 | 30 | P0.7/AD7/DB7 |
| 9 | INT1/P3.3 | 31 | P0.6/AD6/DB6 |
| 10 | T0/P3.4/CMPR– | 32 | P0.5/AD5/DB5 |
| 11 | T1/P3.5/CMP1+ | 33 | P0.4/AD4/DB4 |
| 12 | WR/P3.6/CMP2+ | 34 | P0.3/AD3/DB3 |
| 13 | RD/P3.7CMP3+ | 35 | P0.2/AD2/DB2 |
| 14 | XTAL2 | 36 | P0.1/AD1/DB1 |
| 15 | XTAL1 | 37 | P0.0/AD0/DB0 |
| 16 | VSS | 38 | VCC |
| 17 | NC | 39 | NC |
| 18 | P2.0/A8/CEX0/CMP0 | 40 | +VREF/AVCC |
| 19 | P2.1/A9/CEX1/CMP1 | 41 | –VREF/AVSS |
| 20 | P2.2/A10/CEX2/CMP2 | 42 | ADIN0/P1.0 |
| 21 | P2.3/A11/CEX3/CMP3 | 43 | ADIN1/P1.1 |
| 22 | P2.4/A12/T2EX/A0 | 44 | ADIN2/P1.2 |

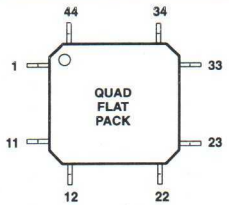
## P8xCE598




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | AVref– | 41 | P3.0/RXD |
| 2 | AVref+ | 42 | P3.1/TXD |
| 3 | $AV_{SS}$ | 43 | P3.2/$\overline{INT0}$ |
| 4 | $AV_{DD}$ | 44 | P3.3/$\overline{INT1}$ |
| 5 | P5.7/ADC7 | 45 | P3.4/T0 |
| 6 | P5.6/ADC6 | 46 | P3.5/T1 |
| 7 | P5.5/ADC5 | 47 | P3.6/$\overline{WR}$ |
| 8 | P5.4/ADC4 | 48 | P3.7/$\overline{RD}$ |
| 9 | P5.3/ADC3 | 49 | NC |
| 10 | P5.2/ADC2 | 50 | NC |
| 11 | P5.1/ADC1 | 51 | XTAL2 |
| 12 | P5.0/ADC0 | 52 | XTAL1 |
| 13 | $V_{SS1}$ | 53 | $V_{DD3}$ |
| 14 | $V_{DD1}$ | 54 | $V_{SS3}$ |
| 15 | STADC | 55 | P2.0/A08 |
| 16 | $\overline{PWM0}$ | 56 | P2.1/A09 |
| 17 | $\overline{PWM1}$ | 57 | P2.2/A10 |
| 18 | $\overline{EW}$ | 58 | P2.3/A11 |
| 19 | P4.0/CMSR0 | 59 | P2.4/A12 |
| 20 | P4.1/CMSR1 | 60 | P2.5/A13 |
| 21 | P4.2/CMSR2 | 61 | P2.6/A14 |
| 22 | P4.3/CMSR3 | 62 | P2.7/A15 |
| 23 | NC | 63 | $\overline{PSEN}$ |
| 24 | P4.4/CMSR4 | 64 | ALE/$\overline{PROG}$ * |
| 25 | P4.5/CMSR5 | 65 | $\overline{EA}/V_{PP}$ * |
| 26 | P4.6/CMT0 | 66 | NC |
| 27 | P4.7/CMT1 | 67 | NC |
| 28 | $V_{SS2}$ | 68 | P0.7/AD7 |
| 29 | $V_{DD2}$ | 69 | P0.6/AD6 |
| 30 | RST | 70 | P0.5/AD5 |
| 31 | P1.0/CT0I/INT2 | 71 | P0.4/AD4 |
| 32 | P1.1/CT1I/INT3 | 72 | P0.3/AD3 |
| 33 | P1.2/CT2I/INT4 | 73 | P0.2/AD2 |
| 34 | P1.3/CT3I/INT5 | 74 | P0.1/AD1 |
| 35 | P1.4/T2 | 75 | P0.0/AD0 |
| 36 | P1.5/RT2 | 76 | $AV_{DD4}$ |
| 37 | $CV_{SS}$ | 77 | $AV_{SS4}$ |
| 38 | P1.6/CTX0 | 78 | REF |
| 39 | P1.7/CTX1 | 79 | CRX1 |
| 40 | $CV_{DD}$ | 80 | CRX0 |

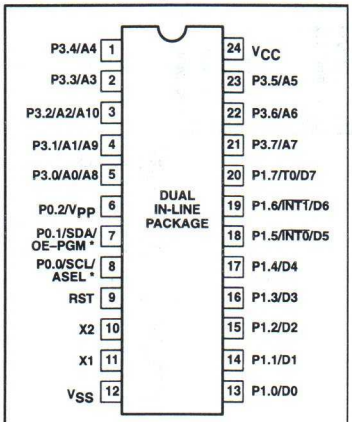
* only 87CE598 with alternative function

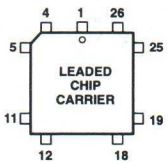
## 80CE654/83CE654




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P1.5 | 23 | P2.5/A13 |
| 2 | P1.6/SCL | 24 | P2.6/A14 |
| 3 | P1.7/SDA | 25 | P2.7/A15 |
| 4 | RST | 26 | $\overline{PSEN}$ |
| 5 | P3.0/RxD | 27 | ALE |
| 6 | $V_{SS4}$ | 28 | $V_{SS2}$ |
| 7 | P3.1/TxD | 29 | $\overline{EA}$ |
| 8 | P3.2/$\overline{INT0}$ | 30 | P0.7/AD7 |
| 9 | P3.3/$\overline{INT1}$ | 31 | P0.6/AD6 |
| 10 | P3.4/T0 | 32 | P0.5/AD5 |
| 11 | P3.5/T1 | 33 | P0.4/AD4 |
| 12 | P3.6/$\overline{WR}$ | 34 | P0.3/AD3 |
| 13 | P3.7/$\overline{RD}$ | 35 | P0.2/AD2 |
| 14 | XTAL2 | 36 | P0.1/AD1 |
| 15 | XTAL1 | 37 | P0.0/AD0 |
| 16 | $V_{SS1}$ | 38 | $V_{DD2}$ |
| 17 | $V_{DD1}$ | 39 | $V_{SS3}$ |
| 18 | P2.0/A8 | 40 | P1.0 |
| 19 | P2.1/A9 | 41 | P1.1 |
| 20 | P2.2/A10 | 42 | P1.2 |
| 21 | P2.3/A11 | 43 | P1.3 |
| 22 | P2.4/A12 | 44 | P1.4 |

### 83C748/87C748, 83C750/87C750, 83C751/87C751

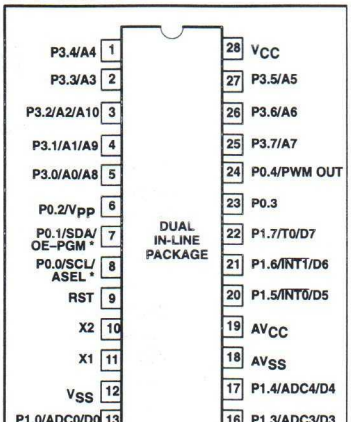


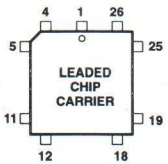




| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P3.4/A4 | 15 | P1.0/D0 |
| 2 | P3.3/A3 | 16 | P1.1/D1 |
| 3 | P3.2/A2/A10 | 17 | P1.2/D2 |
| 4 | P3.1/A1/A9 | 18 | P1.3/D3 |
| 5 | N.C. | 19 | P1.4/D4 |
| 6 | P3.0/A0/A8 | 20 | P1.5/INT0/D5 |
| 7 | P0.2/$V_{PP}$ | 21 | N.C. |
| 8 | P0.1/SDA/OE–PGM * | 22 | N.C. |
| 9 | P0.0/SCL/ASEL * | 23 | P1.6/INT1/D6 |
| 10 | N.C. | 24 | P1.7/T0/D7 |
| 11 | RST | 25 | P3.7/A7 |
| 12 | X2 | 26 | P3.6/A6 |
| 13 | X1 | 27 | P3.5/A5 |
| 14 | $V_{SS}$ | 28 | $V_{CC}$ |

* P0.1 and P0.0 have the alternate functions SCL and SDA, respectively, on the 83C751/87C751.

### 83C749/87C749, 83C752/87C752

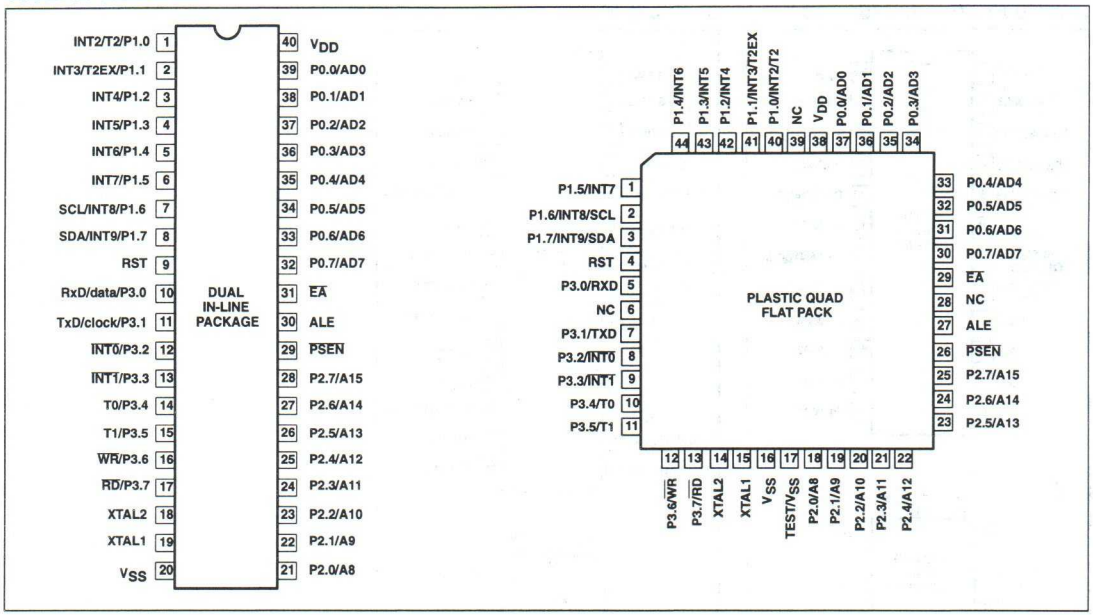



| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | P3.4/A4 | 15 | P1.2/ADC2/D2 |
| 2 | P3.3/A3 | 16 | P1.3/ADC3/D3 |
| 3 | P3.2/A2/A10 | 17 | P1.4/ADC4/D4 |
| 4 | P3.1/A1/A9 | 18 | $AV_{SS}$ |
| 5 | P3.0/A0/A8 | 19 | $AV_{CC}$ |
| 6 | P0.2/$V_{PP}$ | 20 | P1.5/INT0/D5 |
| 7 | P0.1/SDA/OE–PGM * | 21 | P1.6/INT1/D6 |
| 8 | P0.0/SCL/ASEL * | 22 | P1.7/T0/D7 |
| 9 | RST | 23 | P0.3 |
| 10 | X2 | 24 | P0.4/PWM OUT |
| 11 | X1 | 25 | P3.7/A7 |
| 12 | $V_{SS}$ | 26 | P3.6/A6 |
| 13 | P1.0/ADC0/D0 | 27 | P3.5/A5 |
| 14 | P1.1/ADC1/D1 | 28 | $V_{CC}$ |

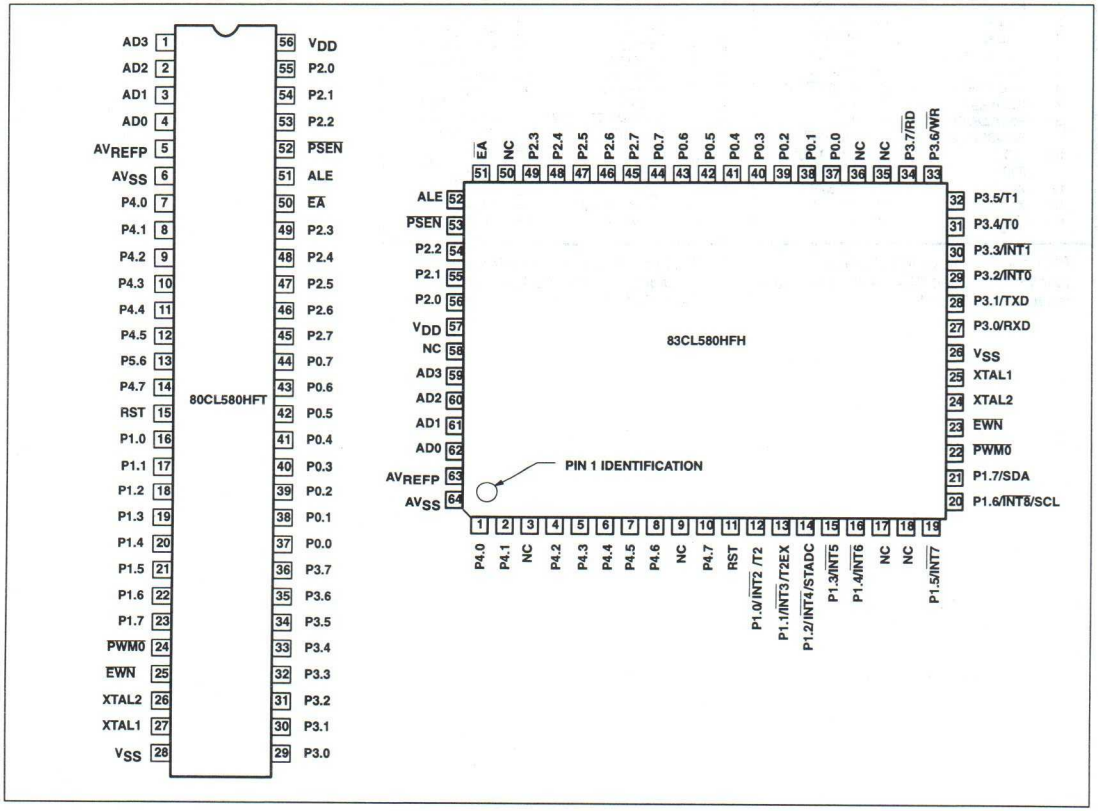
* P0.1 and P0.0 have the alternate functions SCL and SDA, respectively, on the 83C752/87C752.

## 83CL781*/83CL782*




* P1.6 and P1.7 have the alternate functions SCL and SDA, respectively, on the 83CL781 and 83CL782.

## 80CL580/83CL580



# North American Sales Offices, Representatives and Distributors

## PHILIPS SEMICONDUCTORS
811 East Arques Avenue
P.O. Box 3409
Sunnyvale, CA 94088-3409

## ALABAMA
**Huntsville**
Philips Semiconductors
Phone: (205) 464-0111
(205) 464-9101

Elcom, Inc.
Phone: (205) 830-4001

## ARIZONA
**Scottsdale**
Thom Luke Sales, Inc.
Phone: (602) 451-5400

**Tempe**
Philips Semiconductors
Phone: (602) 820-2225

## CALIFORNIA
**Calabasas**
Philips Semiconductors
Phone: (818) 880-6304

**Irvine**
Philips Semiconductors
Phone: (714) 453-0770

**Loomis**
B.A.E. Sales, Inc.
Phone: (916) 652-6777

**San Diego**
Philips Semiconductors
Phone: (619) 560-0242

**San Jose**
B.A.E. Sales, Inc.
Phone: (408) 452-8133

**Sunnyvale**
Philips Semiconductors
Phone: (408) 991-3737

## COLORADO
**Englewood**
Philips Semiconductors
Phone: (303) 792-9011

Thom Luke Sales, Inc.
Phone: (303) 649-9717

## CONNECTICUT
**Wallingford**
JEBCO
Phone: (203) 265-1318

## FLORIDA
**Clearwater**
Conley and Assoc., Inc.
Phone: (813) 572-8895

**Oviedo**
Conley and Assoc., Inc.
Phone: (407) 365-3283

## GEORGIA
**Norcross**
Elcom, Inc.
Phone: (404) 447-8200

## ILLINOIS
**Itasca**
Philips Semiconductors
Phone: (708) 250-0050

**Schaumburg**
Micro-Tex, Inc.
Phone: (708) 885-8200

## INDIANA
**Indianapolis**
Mohrfield Marketing, Inc.
Phone: (317) 546-6969

**Kokomo**
Philips Semiconductors
Phone: (317) 459-5355

## MARYLAND
**Columbia**
Third Wave Solutions, Inc.
Phone: (410) 290-5990

## MASSACHUSETTS
**Chelmsford**
JEBCO
Phone: (508) 256-5800

**Westford**
Philips Semiconductors
Phone: (508) 692-6211

## MICHIGAN
**Novi**
Philips Semiconductors
Phone: (810) 347-1700

Mohrfield Marketing, Inc.
Phone: (810) 348-5799

## MINNESOTA
**Bloomington**
High Technology Sales
Phone: (612) 844-9933

## MISSOURI
**Bridgeton**
Centech, Inc.
Phone: (314) 291-4230

**Raytown**
Centech, Inc.
Phone: (816) 358-8100

## NEW JERSEY
**Toms River**
Philips Semiconductors
Phone: (908) 505-1200
(908) 240-1479

## NEW YORK
**Ithaca**
Bob Dean, Inc.
Phone: (607) 257-0007

**Rockville Centre**
S-J Associates
Phone: (516) 536-4242

**Wappingers Falls**
Philips Semiconductors
Phone: (914) 297-4074

Bob Dean, Inc.
Phone: (914) 297-6406

## NORTH CAROLINA
**Cary**
Philips Semiconductors
Phone: (919) 462-1332

**Charlotte**
Elcom, Inc.
Phone: (704) 543-1229

**Greensboro**
Elcom, Inc.
Phone: (919) 273-8887

**Matthews**
Elcom, Inc.
Phone: (704) 847-4323

## OHIO
**Columbus**
S-J Associates, Inc.
Phone: (614) 885-6700

**Kettering**
S-J Associates, Inc.
Phone: (513) 298-7322

**Solon**
S-J Associates, Inc.
Phone: (216) 349-2700

## OREGON
**Beaverton**
Philips Semiconductors
Phone: (503) 627-0110

Western Technical Sales
Phone: (503) 644-8860

## PENNSYLVANIA
**Erie**
S-J Associates, Inc.
Phone: (216) 888-7004

**Hatboro**
Delta Technical Sales, Inc.
Phone: (215) 957-0600

**Pittsburgh**
S-J Associates, Inc.
Phone: (216) 349-2700

## SOUTH CAROLINA
**Greenville**
Elcom, Inc.
Phone: (803) 370-9119

## TENNESSEE
**Dandridge**
Philips Semiconductors
Phone: (615) 397-5053

## TEXAS
**Austin**
Philips Semiconductors
Phone: (512) 339-9945

Synergistic Sales, Inc.
Phone: (512) 346-2122

**Houston**
Philips Semiconductors
Phone: (713) 999-1316

Synergistic Sales, Inc.
Phone: (713) 937-1990

**Richardson**
Philips Semiconductors
Phone: (972) 644-1610
(972) 705-9555

Synergistic Sales, Inc.
Phone: (972) 644-3500

## UTAH
**Salt Lake City**
Electrodyne
Phone: (801) 264-8050

## WASHINGTON
**Bellevue**
Western Technical Sales
Phone: (206) 641-3900

**Spokane**
Western Technical Sales
Phone: (509) 922-7600

## WISCONSIN
**Waukesha**
Micro-Tex, Inc.
Phone: (414) 542-5352

## CANADA
## PHILIPS SEMICONDUCTORS CANADA, LTD.

**Calgary, Alberta**
Philips Semiconductors/
Components, Inc.
Phone: (403) 293-5969

Tech-Trek, Ltd.
Phone: (403) 241-1719

**Kanata, Ontario**
Philips Semiconductors
Phone: (613) 599-8720

Tech-Trek, Ltd.
Phone: (613) 599-8787

**Montreal, Quebec**
Philips Semiconductors/
Components, Inc.
Phone: (514) 424-7320

**Mississauga, Ontario**
Tech-Trek, Ltd.
Phone: (416) 238-0366

**Richmond, B.C.**
Tech-Trek, Ltd.
Phone: (604) 276-8735

**Scarborough, Ontario**
Philips Semiconductors/
Components, Ltd.
(416) 292-5161

**Ville St. Laurent, Quebec**
Tech-Trek, Ltd.
Phone: (514) 337-7540

## MEXICO
**Anzures Section**
Philips Components
Phone: +9–5 (800) 234-7381

**El Paso, TX**
Philips Components
Phone: (915) 775-4020

## PUERTO RICO
**Caguas**
Mectron Group
Phone: (809) 746-3522

## DISTRIBUTORS

**Contact one of our local distributors:**
Allied Electronics
Anthem Electronics
Arrow/Schweber Electronics
Future Electronics (Canada only)
Hamilton Hallmark
Marshall Industries
Newark Electronics
Penstock
Richardson Electronics
Wyle Electronics
Zeus Electronics

09/12/96

©Philips Electronics North America Corporation 1996 Printed in U.S.A.

# Philips Semiconductors - a worldwide company


**Argentina:** see South America

**Australia:** 34 Waterloo Road, NORTH RYDE, NSW 2113,
Tel. +61 2 9805 4455, Fax. +61 2 9805 4466

**Austria:** Computerstr. 6, A-1101 WIEN, P.O. Box 213,
Tel. +43 1 60 101, Fax. +43 1 60 101 1210

**Belarus:** Hotel Minsk Business Center, Bld. 3, r. 1211, Volodarski Str. 6,
220050 MINSK, Tel. +375 172 200 733, Fax. +375 172 200 773

**Belgium:** see The Netherlands

**Brazil:** see South America

**Bulgaria:** Philips Bulgaria Ltd., Energoproject, 15th floor,
51 James Bourchier Blvd., 1407 SOFIA,
Tel. +359 2 689 211, Fax. +359 2 689 102

**Canada:** PHILIPS SEMICONDUCTORS/COMPONENTS,
Tel. +1 800 234 7381

**China/Hong Kong:** 501 Hong Kong Industrial Technology Centre,
72 Tat Chee Avenue, Kowloon Tong, HONG KONG,
Tel. +852 2319 7888, Fax. +852 2319 7700

**Colombia:** see South America

**Czech Republic:** see Austria

**Denmark:** Prags Boulevard 80, PB 1919, DK-2300 COPENHAGEN S,
Tel. +45 32 88 2636, Fax. +45 31 57 1949

**Finland:** Sinikalliontie 3, FIN-02630 ESPOO,
Tel. +358 615 800, Fax. +358 615 80920

**France:** 4 Rue du Port-aux-Vins, BP317, 92156 SURESNES Cedex,
Tel. +33 1 40 99 6161, Fax. +33 1 40 99 6427

**Germany:** Hammerbrookstraβe 69, D-20097 HAMBURG,
Tel. +49 40 23 53 60, Fax. +49 40 23 536 300

**Greece:** No. 15, 25th March Street, GR 17778 TAVROS,
Tel. +30 1 4894 339/911, Fax. +30 1 4814 240

**Hungary:** see Austria

**India:** Philips INDIA Ltd., Shivsagar Estate, A Block, Dr. Annie Besant Rd.,
Worli, MUMBAI 400 018, Tel. +91 22 4938 541, Fax. +91 22 4938 722

**Indonesia:** see Singapore

**Ireland:** Newstead, Clonskeagh, DUBLIN 14,
Tel. +353 1 7640 000, Fax. +353 1 7640 200

**Israel:** RAPAC Electronics, 7 Kehilat Saloniki St, TEL AVIV 61180,
Tel. +972 3 645 0444, Fax. +972 3 649 1007

**Italy:** PHILIPS SEMICONDUCTORS, Piazza IV Novembre 3,
20124 MILANO, Tel. +39 2 6752 2531, Fax. +39 2 6752 2557

**Japan:** Philips Bldg. 13-37, Kohnan 2-chome, Minato-ku, TOKYO 108,
Tel. +81 3 3740 5130, Fax. +81 3 3740 5077

**Korea:** Philips House, 260-199 Itaewon-dong, Yongsan-ku, SEOUL,
Tel. +82 2 709 1412, Fax. +82 2 709 1415

**Malaysia:** No. 76 Jalan Universiti, 46200 PETALING JAYA, SELANGOR,
Tel. +60 3 750 5214, Fax. +60 3 757 4880

**Mexico:** 5900 Gateway East, Suite 200, EL PASO, TEXAS 79905,
Tel. +9-5 800 234 7381

**Middle East:** see Italy

**Netherlands:** Postbus 90050, 5600 PB EINDHOVEN, Bldg. VB,
Tel. +31 40 27 82785, Fax. +31 40 27 88399

**New Zealand:** 2 Wagener Place, C.P.O. Box 1041, AUCKLAND,
Tel. +64 9 849 4160, Fax. +64 9 849 7811

**Norway:** Box 1, Manglerud 0612, OSLO,
Tel. +47 22 74 8000, Fax. +47 22 74 8341

**Philippines:** Philips Semiconductors Philippines Inc.,
106 Valero St. Salcedo Village, P.O. Box 2108 MCC, MAKATI,
Metro MANILA, Tel. +63 2 816 6380, Fax. +63 2 817 3474

**Poland:** Ul. Lukiska 10, PL 04-123 WARSZAWA,
Tel. +48 22 612 2831, Fax. +48 22 612 2327

**Portugal:** see Spain

**Romania:** see Italy

**Russia:** Philips Russia, Ul. Usatcheva 35A, 119048 MOSCOW,
Tel. +7 095 926 5361, Fax +7 095 564 8323

**Singapore:** Lorong 1, Toa Payoh, SINGAPORE 1231,
Tel. +65 350 2538, Fax. +65 251 6500

**Slovakia:** see Austria

**Slovenia:** see Italy

**South Africa:** S.A. PHILIPS Pty Ltd., 195-215 Main Road Martindale,
2092 JOHANNESBURG, P.O. Box 7430, Johannesburg 2000,
Tel. +27 11 470 5911, Fax. +27 11 470 5494

**South America:** Rua do Rocio 220, 5th Floor, Suite 51,
04552-903 São Paulo, SÃO PAULO-SP, Brazil,
Tel. +55 11 821 2333, Fax. +55 11 829 1849

**Spain:** Balmes 22, 08007 BARCELONA,
Tel. +34 3 301 6312, Fax. +34 3 301 4107

**Sweden:** Kottbygatan 7, Akalla. S-16485 STOCKHOLM,
Tel. +46 8 632 2000, Fax. +46 8 632 2745

**Switzerland:** Allmendstrasse 140, CH-8027 ZÜRICH,
Tel. +41 1 488 2686, Fax. +41 1 481 7730

**Taiwan:** PHILIPS TAIWAN Ltd., 23-30F, 66,
Chung Hsiao West Road, Sec. 1, P.O. Box 22978, TAIPEI 100,
Tel. +886 2 382 4443, Fax. +886 2 382 4444

**Thailand:** PHILIPS ELECTRONICS (THAILAND) Ltd.,
209/2 Sanpavuth-Bangna Road Prakanong, BANGKOK 10260,
Tel. +66 2 745 4090, Fax. +66 2 398 0793

**Turkey:** Talatpasa Cad. No. 5, 80640 GÜLTEPE/ISTANBUL,
Tel. +90 212 279 2770, Fax. +90 212 282 6707

**Ukraine:** PHILIPS UKRAINE, 4 Patrice Lumumba str., Building B, Floor 7,
252042 KIEV, Tel. +380 44 264 2776, Fax. +380 44 268 0461

**United Kingdom:** Philips Semiconductors Ltd., 276 Bath Road, Hayes,
MIDDLESEX UB3 5BX, Tel. +44 181 730 5000, Fax. +44 181 754 8421

**United States:** 811 East Arques Avenue, SUNNYVALE, CA 94088-3409,
Tel. +1 800 234 7381

**Uruguay:** see South America

**Vietnam:** see Singapore

**Yugoslavia:** PHILIPS, Trg N. Pasica 5/v, 11000 BEOGRAD,
Tel. +381 11 825 344, Fax. +381 11 635 777

**For all other countries apply to:** Philips Semiconductors, Marketing and Sales Communications,
Building BE-p, P.O. Box 218, 5600 MD EINDHOVEN, The Netherlands, Fax . +31 40 27 24825

**Internet:** http://www.semiconductors.philips.com



©Philips Electronics N.V. 1996

SCA51

All rights are reserved. Reproduction in whole or in part is prohibited without the prior written consent of the copyright owner.

The information presented in this document does not form part of any quotation or contract, is believed to be accurate and reliable and may be changed without notice. No liability will be accepted by the publisher for any consequence of its use. Publication thereof does not convey nor imply any license under patent- or industrial or intellectual property rights.



Printed in the USA 457031/47.5M/CR5/pp1472 Date of release: 08–96 Document order number: 9397 750 00963



Let's make things better.

Philips Semiconductors

PHILIPS