# DATA SHEET

## P89C51X2/52X2/54X2/58X2

80C51 8-bit Flash microcontroller family

4K/8K/16K/32K Flash
128B/256B RAM

Preliminary data
Supersedes data of 2002 Feb 28

2002 Jun 05

Philips
Semiconductors

PHILIPS

## DESCRIPTION

The Philips microcontrollers described in this data sheet are high-performance static 80C51 designs. They are manufactured in an advanced CMOS process and contain a non-volatile Flash program memory. They support both 12-clock and 6-clock operation.

The P89C51X2 and P89C52X2/54X2/58X2 contain 128 byte RAM and 256 byte RAM respectively, 32 I/O lines, three 16-bit counter/timers, a six-source, four-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 devices are static designs which offer a wide range of operating frequencies down to zero. Two software selectable modes of power reduction — idle mode and power-down mode — are available. 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. Since the design is static, the clock can be stopped without loss of user data. Then the execution can be resumed from the point the clock was stopped.

## SELECTION TABLE

For applications requiring more Flash and RAM, as well as more on-chip peripherals, see the P89C66x and P89C51Rx2 data sheets.

| Type | Memory | | | | Timers | | | | Serial Interfaces | | | | ADC bits/ch. | I/O Pins | Interrupts (External) | Program Security | Default Clock Rate | Optional Clock Rate | Max. Freq. at 6-clk / 12-clk (MHz) | Freq. Range at 3V (MHz) | Freq. Range at 5V (MHz) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | RAM | ROM | OTP | Flash | # of Timers | PWM | PCA | WD | UART | I2C | CAN | SPI | | | | | | | | | |
| P89C58X2 | 256B | – | – | 32K | 3 | – | – | – | ✓ | – | – | – | – | 32 | 6 (2) | ✓ | 12–clk | 6-clk | 20/33 | – | 0–20/33 |
| P89C54X2 | 256B | – | – | 16K | 3 | – | – | – | ✓ | – | – | – | – | 32 | 6 (2) | ✓ | 12–clk | 6-clk | 20/33 | – | 0–20/33 |
| P89C52X2 | 256B | – | – | 8K | 3 | – | – | – | ✓ | – | – | – | – | 32 | 6 (2) | ✓ | 12–clk | 6-clk | 20/33 | – | 0–20/33 |
| P89C51X2 | 128B | – | – | 4K | 3 | – | – | – | ✓ | – | – | – | – | 32 | 6 (2) | ✓ | 12–clk | 6-clk | 20/33 | – | 0–20/33 |

**NOTE:**
1. $I^2C$ = Inter-Integrated Circuit Bus; CAN = Controller Area Network; SPI = Serial Peripheral Interface; PCA = Programmable Counter Array; ADC = Analog-to-Digital Converter; PWM = Pulse Width Modulation

## DEVICE COMPARISON TABLE

| Item | P89C5xX2 devices (this data sheet) | P89C5xBx devices (separate data sheet) | P89C5xUxx devices (discontinued) |
|---|---|---|---|
| Type Description | P89C5xX2Bxx/P89C5xX2Fxx | P89C5xBx | P89C5xUBxx/P89C5xUFxx |
| PROGRAMMING ALGORITHM | When using a parallel programmer, be sure to select P89C5xX2 devices. IF DEVICES ARE NOT YET SELECTABLE ASK YOUR PROGRAMMER VENDOR FOR A SOFTWARE UPDATE | When using a parallel programmer, be sure to select P89C5xBx devices. IF DEVICES ARE NOT YET SELECTABLE ASK YOUR PROGRAMMER VENDOR FOR A SOFTWARE UPDATE | When using a parallel programmer, be sure to select P89C5xUxxx devices. |
| Quad Flat Package type | LQFP package (P89C5xX2xBD) | LQFP package (P89C5xBBD) | PQFP package (P89C5xUxBB) |
| Package identifiers | PLCC = A<br>LQFP = BD<br>PDIP = N | PLCC = A<br>LQFP = BD<br>PDIP = P | PLCC = AA<br>LQFP = BB<br>PDIP = PN |
| Flash Memory program and erase cycles | 10,000 program and erase cycles | 10,000 program and erase cycles | 100 program and erase cycles |

## FEATURES

- 80C51 Central Processing Unit
  - 4 Kbytes Flash (P89C51X2)
  - 8 Kbytes Flash (P89C52X2)
  - 16 Kbytes Flash (P89C54X2)
  - 32 Kbytes Flash (P89C58X2)
  - 128 byte RAM (P89C51X2)
  - 256 byte RAM (P89C52/54X2/58X2)
  - Boolean processor
  - Fully static operation
- 12-clock operation with selectable 6-clock operation (via software or via parallel programmer)
- Memory addressing capability
  - Up to 64 Kbytes ROM and 64 Kbytes RAM
- Power control modes:
  - Clock can be stopped and resumed
  - Idle mode
  - Power-down mode
- Two speed ranges
  - 0 to 20 MHz with 6-clock operation
  - 0 to 33 MHz with 12-clock operation
- LQFP, PLCC or DIP package
- Extended temperature ranges
- Dual Data Pointers
- Three security bits
- Four interrupt priority levels
- Six interrupt sources
- Four 8-bit I/O ports
- Full-duplex enhanced UART
  - Framing error detection
  - Automatic address recognition
- Three 16-bit timers/counters T0, T1 (standard 80C51) and additional T2 (capture and compare)
- Programmable clock-out pin
- Asynchronous port reset
- Low EMI (inhibit ALE, slew rate controlled outputs, and 6-clock mode)
- Wake-up from Power Down by an external interrupt

## P89C51X2 ORDERING INFORMATION (4 KBYTE FLASH)

| Type number | Package | | | Temperature Range (°C) |
|---|---|---|---|---|
| | Name | Description | Version | |
| P89C51X2BA | PLCC44 | plastic leaded chip carrier; 44 leads | SOT187-2 | 0 to +70 |
| P89C51X2BN | DIP40 | plastic dual in-line package; 40 leads (600 mil) | SOT129-1 | 0 to +70 |
| P89C51X2BBD | LQFP44 | plastic low profile quad flat package; 44 leads | SOT389-1 | 0 to +70 |
| P89C51X2FA | PLCC44 | plastic leaded chip carrier; 44 leads | SOT187-2 | –40 to +85 |

## P89C52X2 ORDERING INFORMATION (8 KBYTE FLASH)

| Type number | Package | | | Temperature Range (°C) |
|---|---|---|---|---|
| | Name | Description | Version | |
| P89C52X2BA | PLCC44 | plastic leaded chip carrier; 44 leads | SOT187-2 | 0 to +70 |
| P89C52X2BN | DIP40 | plastic dual in-line package; 40 leads (600 mil) | SOT129-1 | 0 to +70 |
| P89C52X2BBD | LQFP44 | plastic low profile quad flat package; 44 leads | SOT389-1 | 0 to +70 |
| P89C52X2FA | PLCC44 | plastic leaded chip carrier; 44 leads | SOT187-2 | –40 to +85 |
| P89C52X2FN | DIP40 | plastic dual in-line package; 40 leads (600 mil) | SOT129-1 | –40 to +85 |
| P89C52X2FBD | LQFP44 | plastic low profile quad flat package; 44 leads | SOT389-1 | –40 to +85 |

## P89C54X2 ORDERING INFORMATION (16 KBYTE FLASH)

| Type number | Package | | | Temperature Range (°C) |
|---|---|---|---|---|
| | Name | Description | Version | |
| P89C54X2BA | PLCC44 | plastic lead chip carrier; 44 leads | SOT187-2 | 0 to +70 |
| P89C54X2BN | DIP40 | plastic dual in-line package; 40 leads (600 mil) | SOT129-1 | 0 to +70 |
| P89C54X2BBD | LQFP44 | plastic low profile quad flat package; 44 leads | SOT389-1 | 0 to +70 |
| P89C54X2FA | PLCC44 | plastic lead chip carrier; 44 leads | SOT187-2 | –40 to +85 |

## P89C58X2 ORDERING INFORMATION (32 KBYTE FLASH)

| Type number | Package | | | Temperature Range (°C) |
|---|---|---|---|---|
| | Name | Description | Version | |
| P89C58X2BA | PLCC44 | plastic lead chip carrier; 44 leads | SOT187-2 | 0 to +70 |
| P89C58X2BN | DIP40 | plastic dual in-line package; 40 leads (600 mil) | SOT129-1 | 0 to +70 |
| P89C58X2BBD | LQFP44 | plastic low profile quad flat package; 44 leads | SOT389-1 | 0 to +70 |
| P89C58X2FA | PLCC44 | plastic lead chip carrier; 44 leads | SOT187-2 | –40 to +85 |

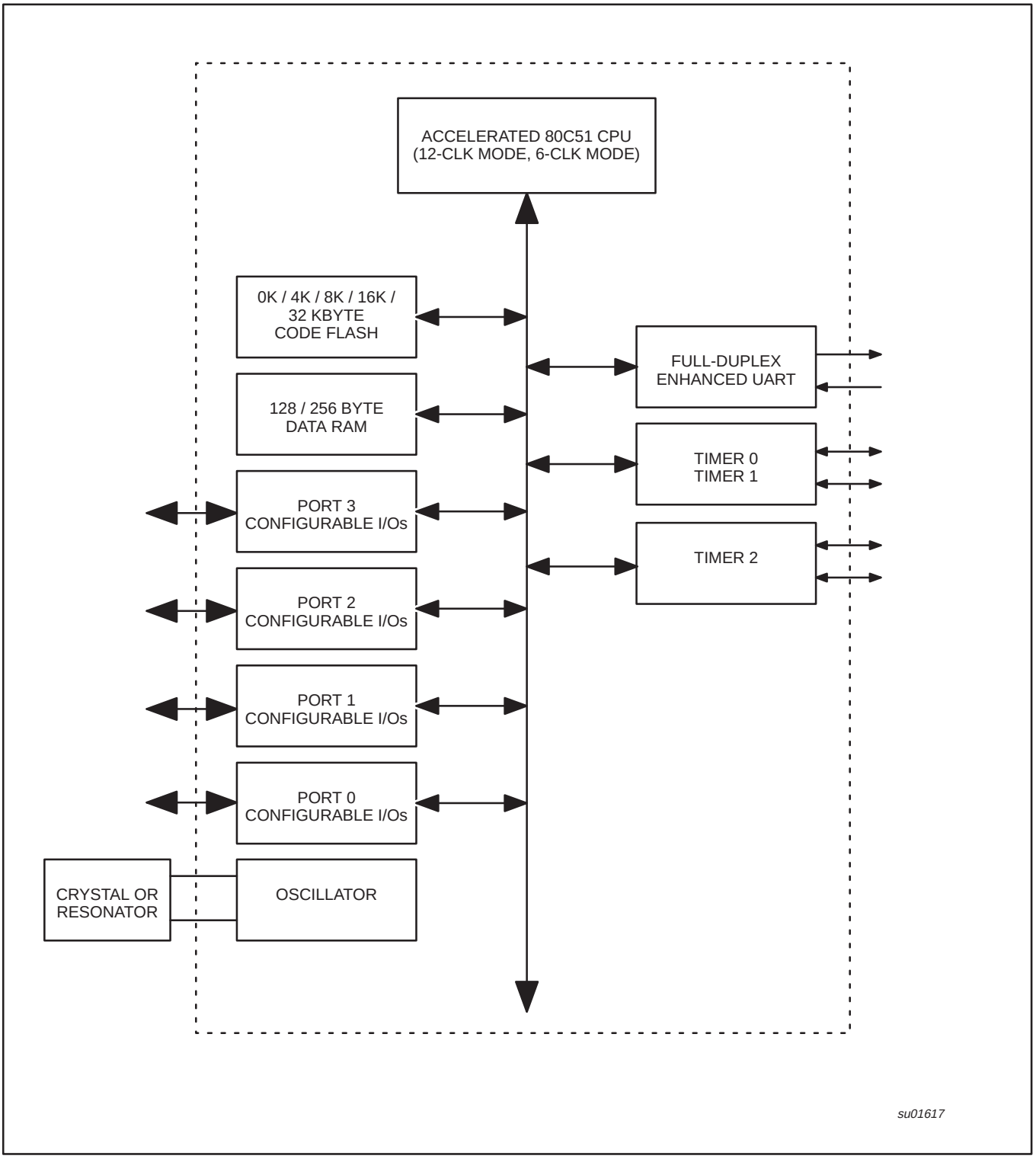
## PART NUMBER DERIVATION

| Memory | Temperature Range | Package |
|---|---|---|
| P89C51X2<br>9 = Flash<br>1 = 128 BYTES RAM 4 KBYTES FLASH<br>2 = 256 BYTES RAM 8 KBYTES FLASH<br>4 = 256 BYTES RAM 16 KBYTES FLASH<br>8 = 256 BYTES RAM 32 KBYTES FLASH<br>X2 = 6-clock mode available | B = 0 °C TO +70 °C<br>F = –40 °C TO +85 °C | A = PLCC<br>N = DIP<br>BD = LQFP |

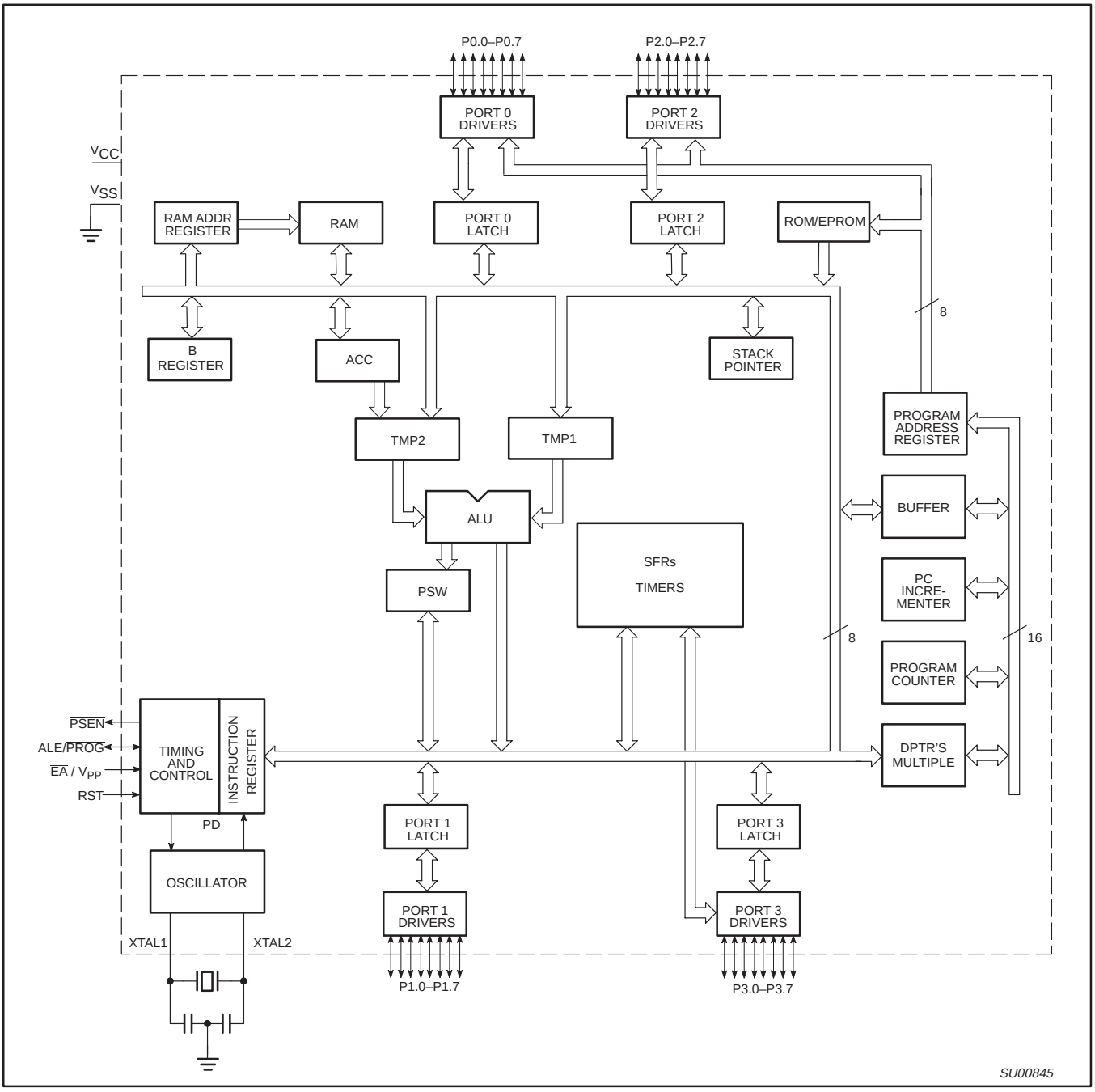
The following table illustrates the correlation between operating mode, power supply and maximum external clock frequency:

| Operating Mode | Power Supply | Maximum Clock Frequency |
|---|---|---|
| 6-clock | 5 V ± 10% | 20 MHz |
| 12-clock | 5 V ± 10% | 33 MHz |

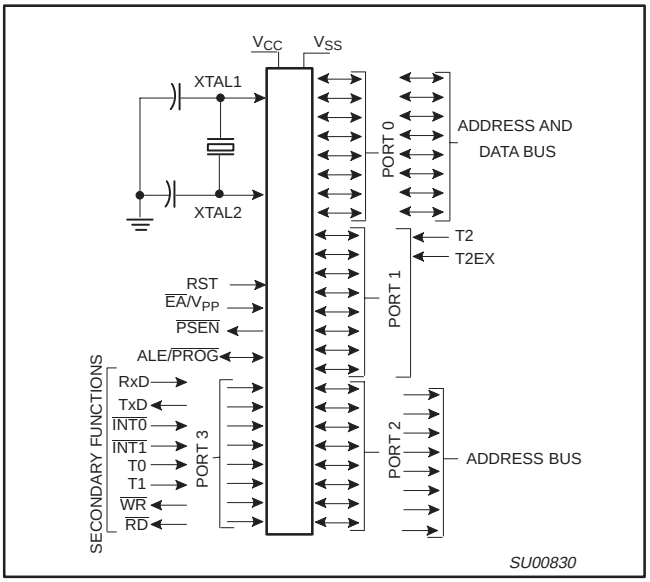
## BLOCK DIAGRAM 1



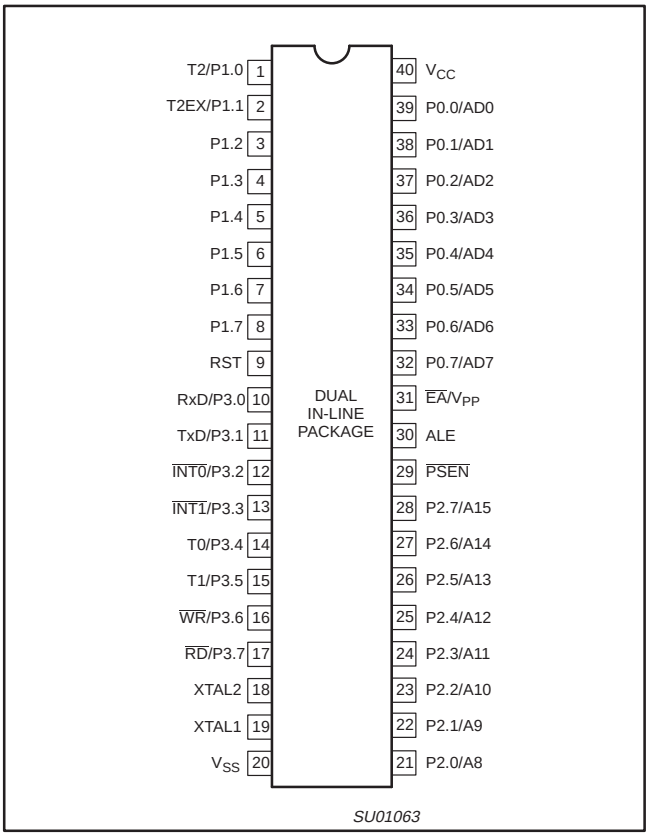
## BLOCK DIAGRAM 2 (CPU-ORIENTED)



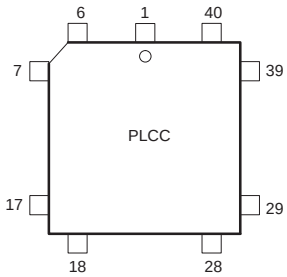
## LOGIC SYMBOL




## PLASTIC DUAL IN-LINE PACKAGE PIN CONFIGURATIONS




## PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS

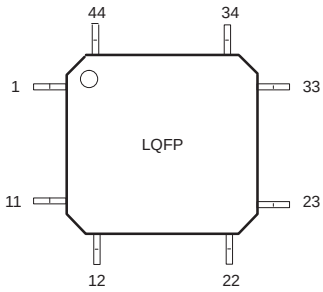



| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | NIC* | 16 | P3.4/T0 | 31 | P2.7/A15 |
| 2 | P1.0/T2 | 17 | P3.5/T1 | 32 | PSEN |
| 3 | P1.1/T2EX | 18 | P3.6/WR | 33 | ALE |
| 4 | P1.2 | 19 | P3.7/RD | 34 | NIC* |
| 5 | P1.3 | 20 | XTAL2 | 35 | EA/VPP |
| 6 | P1.4 | 21 | XTAL1 | 36 | P0.7/AD7 |
| 7 | P1.5 | 22 | VSS | 37 | P0.6/AD6 |
| 8 | P1.6 | 23 | NIC* | 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 | NIC* | 27 | P2.3/A11 | 42 | P0.1/AD1 |
| 13 | P3.1/TxD | 28 | P2.4/A12 | 43 | P0.0/AD0 |
| 14 | P3.2/INT0 | 29 | P2.5/A13 | 44 | VCC |
| 15 | P3.3/INT1 | 30 | P2.6/A14 | | |

* NO INTERNAL CONNECTION

SU01062

## LOW PROFILE QUAD FLAT PACK PIN FUNCTIONS




| Pin | Function | Pin | Function | Pin | Function |
|---|---|---|---|---|---|
| 1 | P1.5 | 16 | VSS | 31 | P0.6/AD6 |
| 2 | P1.6 | 17 | NIC* | 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 | NIC* | 21 | P2.3/A11 | 36 | P0.1/AD1 |
| 7 | P3.1/TxD | 22 | P2.4/A12 | 37 | P0.0/AD0 |
| 8 | P3.2/INT0 | 23 | P2.5/A13 | 38 | VCC |
| 9 | P3.3/INT1 | 24 | P2.6/A14 | 39 | NIC* |
| 10 | P3.4/T0 | 25 | P2.7/A15 | 40 | P1.0/T2 |
| 11 | P3.5/T1 | 26 | PSEN | 41 | P1.1/T2EX |
| 12 | P3.6/WR | 27 | ALE | 42 | P1.2 |
| 13 | P3.7/RD | 28 | NIC* | 43 | P1.3 |
| 14 | XTAL2 | 29 | EA/VPP | 44 | P1.4 |
| 15 | XTAL1 | 30 | P0.7/AD7 | | |

* NO INTERNAL CONNECTION

SU01487

## PIN DESCRIPTIONS

| MNEMONIC | PIN NUMBER DIP | PLCC | LQFP | TYPE | NAME AND FUNCTION |
|---|---|---|---|---|---|
| $V_{SS}$ | 20 | 22 | 16 | I | **Ground:** 0 V 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 received code bytes during Flash 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 for Port 1 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 |
| 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 Flash 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 | **$\overline{\text{INT0}}$ (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/$\overline{\text{PROG}}$ | 30 | 33 | 27 | 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 (12-clk) or 1/3 (6-clk Mode) 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 ($\overline{\text{PROG}}$) during Flash programming. 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 device 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. |
| $\overline{\text{EA}}$/$V_{PP}$ | 31 | 35 | 29 | I | **External Access Enable/Programming Supply Voltage:** $\overline{\text{EA}}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H to 0FFFH/1FFFH/3FFFH/7FFFH. If $\overline{\text{EA}}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than the on-chip Flash. This pin also receives the 5 V / 12 V programming supply voltage ($V_{PP}$) during Flash programming. 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 $V_{CC}$ + 0.5 V or $V_{SS}$ – 0.5 V, respectively.

**Table 1. 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 | – | – | – | – | WUPD | 0 | – | DPS | xxx000x0B |
| B* | B register | F0H | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | 00H |
| CKCON | Clock Control Register | 8FH | – | – | – | – | – | – | – | X2 | xxx00000B |
| 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 | $\overline{EA}$ | – | 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 |
| IPH# | Interrupt Priority High | B7H | – | – | PT2H | PSH | PT1H | PX1H | PT0H | PX0H | 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 | 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 | 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 | 000000x0B |
| 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{T}$2 | CP/$\overline{RL}$2 | 00H |
| T2MOD# | Timer 2 Mode Control | C9H | – | – | – | – | – | – | T2OE | DCEN | xxxxxx00B |
| TH0 | Timer High 0 | 8CH | | | | | | | | | 00H |
| TH1 | Timer High 1 | 8DH | | | | | | | | | 00H |
| TH2# | Timer High 2 | CDH | | | | | | | | | 00H |
| TL0 | Timer Low 0 | 8AH | | | | | | | | | 00H |
| TL1 | Timer Low 1 | 8BH | | | | | | | | | 00H |
| TL2# | Timer Low 2 | CCH | | | | | | | | | 00H |
| TMOD | Timer Mode | 89H | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 | 00H |

**NOTE:**
Unused register bits that are not defined should not be set by the user's program. If violated, the device could function incorrectly.
* SFRs are bit addressable.
\# SFRs are modified from or added to the 80C51 SFRs.
– Reserved bits.
1. Reset value depends on reset source.

## FLASH EPROM MEMORY

### General Description

The P89C51X2/P89C52X2/P89C54X2/P89C58X2 FLASH reliably stores memory contents even after 10,000 erase and program cycles. The cell is designed to optimize the erase and programming mechanisms. In addition, the combination of advanced tunnel oxide processing and low internal electric fields for erase and programming operations produces reliable cycling.

### Features

- FLASH EPROM internal program memory with Chip Erase
- Up to 64 kbyte external program memory if the internal program memory is disabled ($\overline{EA}$ = 0)
- Programmable security bits
- 10,000 minimum erase/program cycles for each byte
- 10 year minimum data retention
- Programming support available from many popular vendors

## OSCILLATOR CHARACTERISTICS

Using the oscillator, 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. However, minimum and maximum high and low times specified in the data sheet must be observed.

### Clock Control Register (CKCON)

This device provides control of the 6-clock/12-clock mode by both an SFR bit (bit X2 in register CKCON) and a Flash bit (bit FX2, located in the Security Block). When X2 is 0, 12-clock mode is activated. By setting this bit to 1, the system is switching to 6-clock mode. Having this option implemented as SFR bit, it can be accessed anytime and changed to either value. Changing X2 from 0 to 1 will result in executing user code at twice the speed, since all system time intervals will be divided by 2. Changing back from 6-clock to 12-clock mode will slow down running code by a factor of 2.

The Flash clock control bit (FX2) activates the 6-clock mode when programmed using a parallel programmer, superceding the X2 bit (CKCON.0). Please also see Table 2 below.

### Table 2.

| FX2 clock mode bit (can only be set by parallel programmer) | X2 bit (CKCON.0) | CPU clock mode |
|---|---|---|
| erased | 0 | 12-clock mode (default) |
| erased | 1 | 6-clock mode |
| programmed | X | 6-clock mode |

### Programmable Clock-Out Pin

A 50% duty cycle clock can be programmed to be output 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 61 Hz to 4 MHz at a 16 MHz operating frequency in 12-clock mode (122 Hz to 8 MHz in 6-clock mode).

To configure the Timer/Counter 2 as a clock generator, bit C/$\overline{T}$2 (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:

$$\frac{\text{Oscillator Frequency}}{n \times (65536 - \text{RCAP2H, RCAP2L})}$$

Where:

n = 2 in 6-clock mode, 4 in 12-clock mode.

(RCAP2H,RCAP2L) = the content of RCAP2H and RCAP2L taken as a 16-bit unsigned integer.

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.

## RESET

A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods in 12-clock and 12 oscillator periods in 6-clock mode), while the oscillator is running. To insure a reliable 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, unless it has been set to 6-clock operation using a parallel programmer.

## LOW POWER MODES

### Stop Clock Mode

The static design enables the clock speed to be reduced down to 0 MHz (stopped). When the oscillator is stopped, the RAM and Special Function Registers retain their values. This mode allows step-by-step utilization and permits reduced system power consumption by lowering the clock frequency down to any value. For lowest power consumption the Power Down mode is suggested.

### Idle Mode

In idle mode (see Table 3), 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 3) 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 down to 2.0 V and care must be taken to return $V_{CC}$ to the minimum specified operating voltages before 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. WUPD (AUXR1.3–Wakeup from Power Down) enables or disables the wakeup from power down with external interrupt. Where:

WUPD = 0: Disable
WUPD = 1: Enable

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 10 ms).

To terminate Power Down with an external interrupt, $\overline{INT0}$ or $\overline{INT1}$ 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 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.

## ONCE™ Mode

The ONCE (“On-Circuit Emulation”) Mode facilitates testing and debugging of systems without the device having to be removed from the circuit. The ONCE Mode is invoked in the following way:

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 device 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 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 |

## TIMER 0 AND TIMER 1 OPERATION

## Timer 0 and Timer 1

The “Timer” or “Counter” function is selected by control bits C/$\overline{T}$ 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 2 shows the Mode 0 operation.

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 TFn. The counted input is enabled to the Timer when TRn = 1 and either GATE = 0 or $\overline{INTn}$ = 1. (Setting GATE = 1 allows the Timer to be controlled by external input $\overline{INTn}$, to facilitate pulse width measurements). TRn is a control bit in the Special Function Register TCON (Figure 3).

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

Mode 0 operation is the same for Timer 0 as for Timer 1. 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 (TLn) with automatic reload, as shown in Figure 4. Overflow from TLn not only sets TFn, but also reloads TLn with the contents of THn, which is preset by software. The reload leaves THn unchanged.

Mode 2 operation is the same for Timer 0 as for Timer 1.

**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 5. TL0 uses the Timer 0 control bits: C/$\overline{T}$, GATE, TR0, and TF0 as well as pin $\overline{INT0}$. 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.

**TMOD** Address = 89H Reset Value = 00H

Not Bit Addressable

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 |
| **TIMER 1** | | | | **TIMER 0** | | | |

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| TMOD.3/ TMOD.7 | GATE | Gating control when set. Timer/Counter “n” is enabled only while “$\overline{INTn}$” pin is high and “TRn” control pin is set. when cleared Timer “n” is enabled whenever “TRn” control bit is set. |
| TMOD.2/ TMOD.6 | C/$\overline{T}$ | Timer or Counter Selector cleared for Timer operation (input from internal system clock.) Set for Counter operation (input from “Tn” input pin). |

| M1 | M0 | OPERATING |
|---|---|---|
| 0 | 0 | 8048 Timer: “TLn” serves as 5-bit prescaler. |
| 0 | 1 | 16-bit Timer/Counter: “THn” and “TLn” are cascaded; there is no prescaler. |
| 1 | 0 | 8-bit auto-reload Timer/Counter: “THn” holds a value which is to be reloaded into “TLn” 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. |

SU01580

**Figure 1. Timer/Counter 0/1 Mode Control (TMOD) Register**

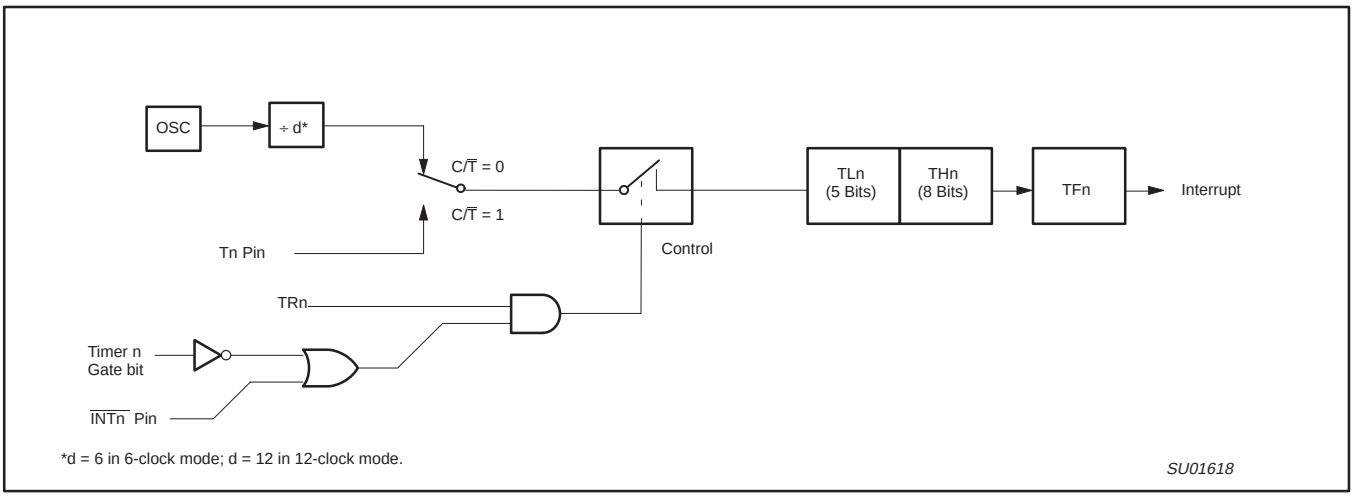



**Figure 2. Timer/Counter 0/1 Mode 0: 13-Bit Timer/Counter**

**TCON** Address = 88H Reset Value = 00H

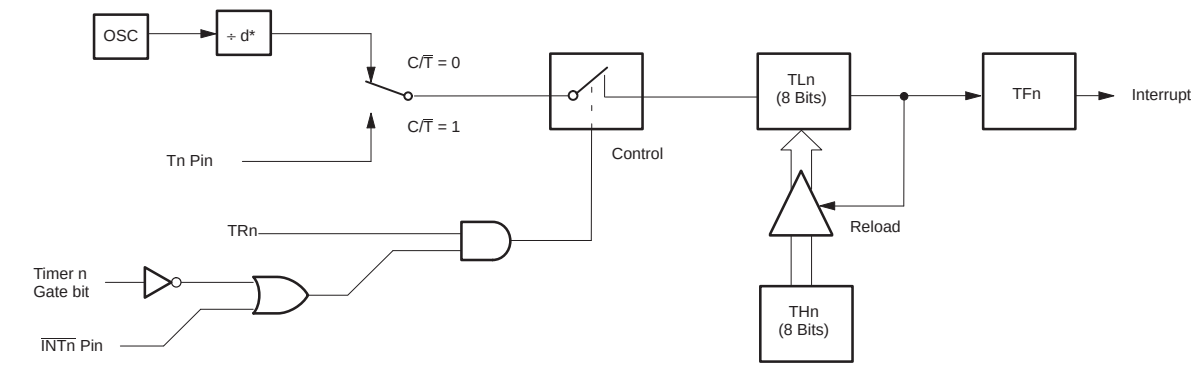
Bit Addressable

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |

| 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. |

*SU01516*

**Figure 3. Timer/Counter 0/1 Control (TCON) Register**




*d = 6 in 6-clock mode; d = 12 in 12-clock mode.

*SU01619*

**Figure 4. Timer/Counter 0/1 Mode 2: 8-Bit Auto-Reload**

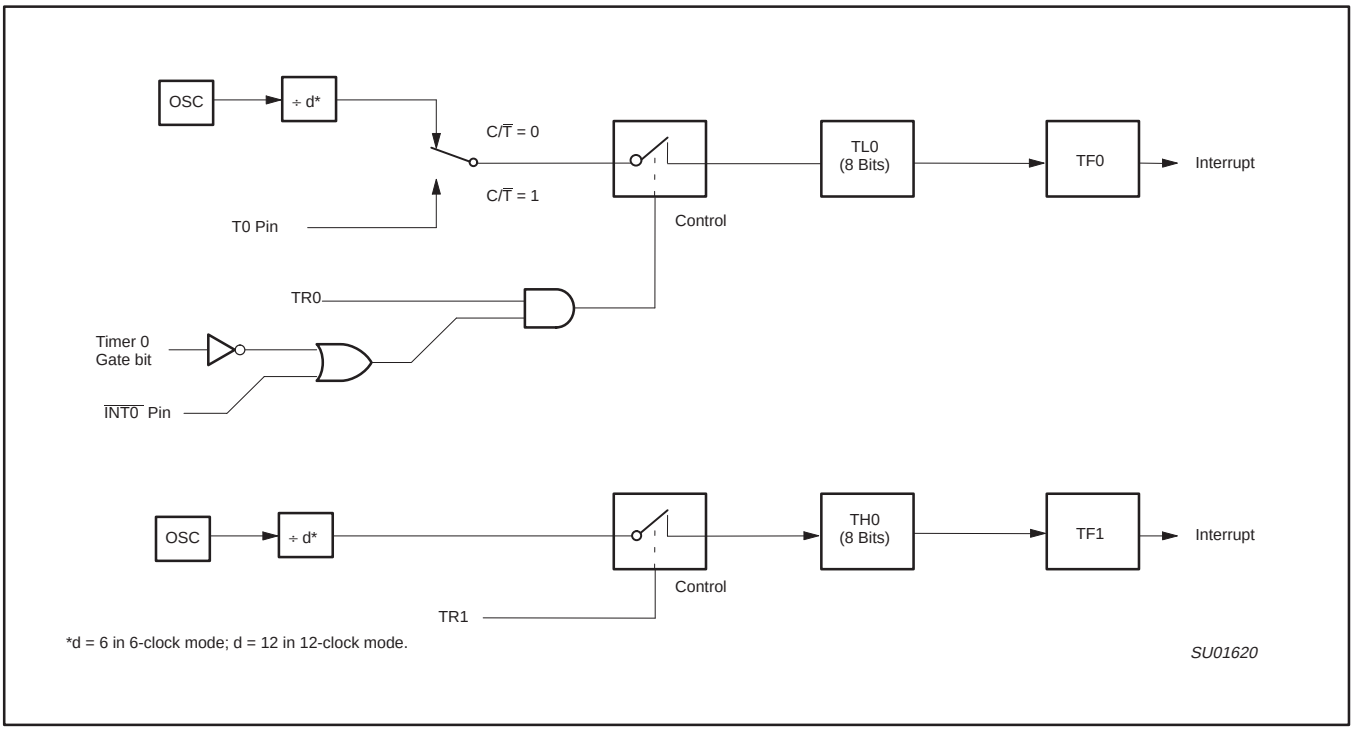



**Figure 5. Timer/Counter 0 Mode 3: Two 8-Bit Counters**

## 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/$\overline{T}$2 in the special function register T2CON (see Figure 6). 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 4.

### 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/$\overline{T}$2 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). 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 7 (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 (12-clock Mode) or osc/6 (6-clock Mode) 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/$\overline{T}$2 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 8). After reset, DCEN=0 which means Timer 2 will default to counting up. If DCEN is set, Timer 2 can count up or down depending on the value of the T2EX pin.

Figure 9 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.

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 10 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.

A logic 0 applied to pin T2EX causes Timer 2 to count down. The timer will underflow when TL2 and TH2 become equal to the value stored in RCAP2L and RCAP2H. A 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.

**Table 4. Timer 2 Operating Modes**

| RCLK + TCLK | CP/$\overline{RL}$2 | 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) |

**T2CON** Address = C8H Reset Value = 00H
Bit Addressable

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/$\overline{T}$2 | CP/$\overline{RL}$2 |

| 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{T}$2 | T2CON.1 | Timer or counter select. (Timer 2)<br>0 = Internal timer (OSC/12 in 12-clock mode or OSC/6 in 6-clock mode)<br>1 = External event counter (falling edge triggered). |
| CP/$\overline{RL}$2 | 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. |

*SU01621*

**Figure 6. Timer/Counter 2 (T2CON) Control Register**

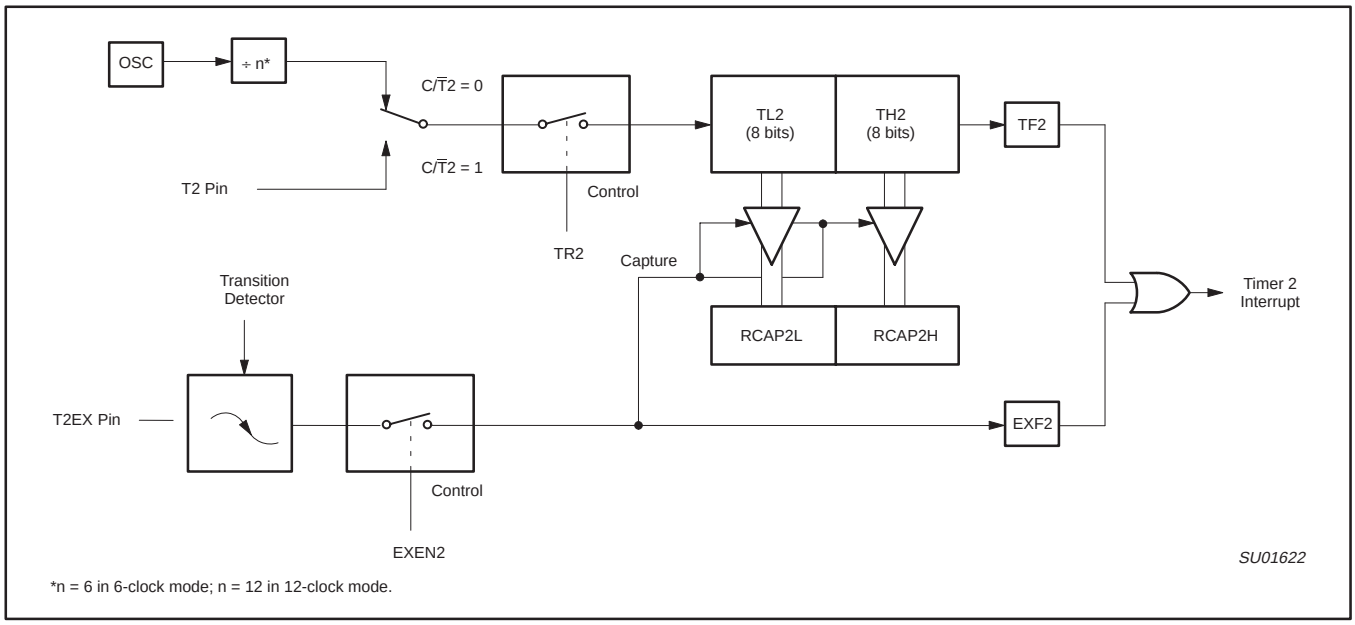



**Figure 7. Timer 2 in Capture Mode**

**T2MOD** Address = 0C9H Reset Value = XXXX XX00B

Not Bit Addressable

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | T2OE | DCEN |

| Symbol | Position | Function |
|---|---|---|
| — | | Not implemented, reserved for future use.* |
| T2OE | T2MOD.1 | Timer 2 Output Enable bit. |
| DCEN | T2MOD.0 | 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.

SU01519

**Figure 8. Timer 2 Mode (T2MOD) Control Register**

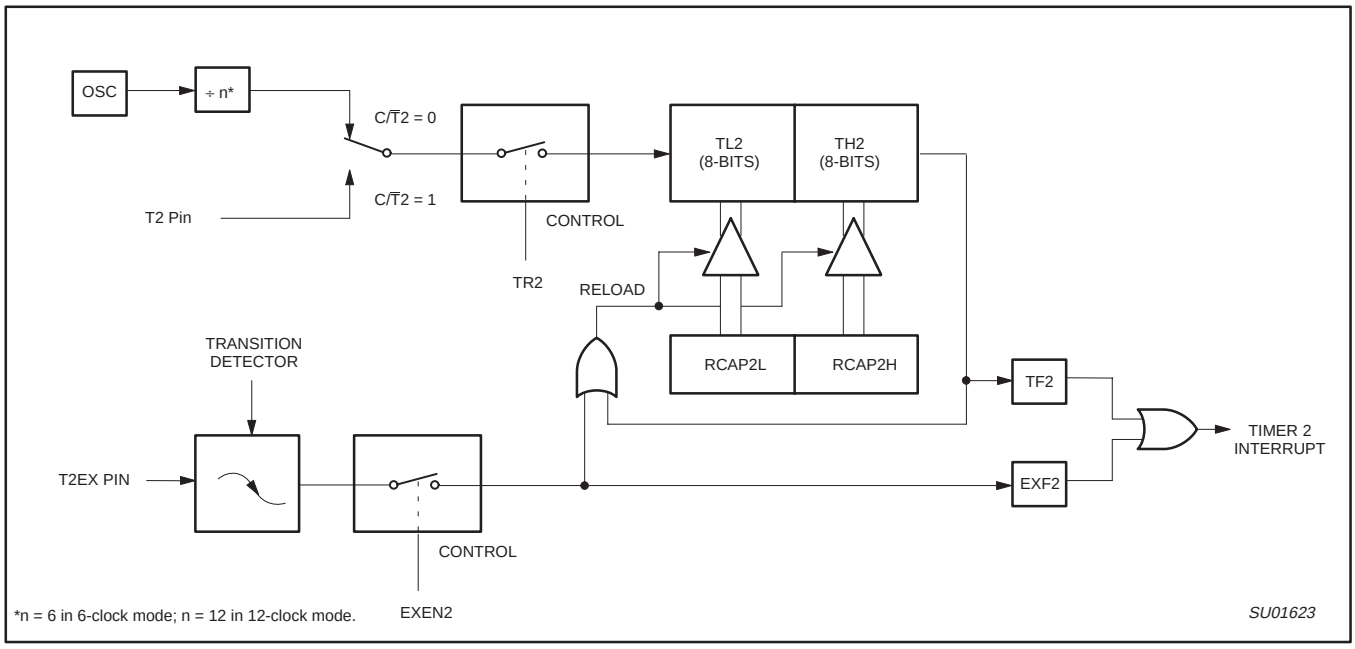



**Figure 9. Timer 2 in Auto-Reload Mode (DCEN = 0)**

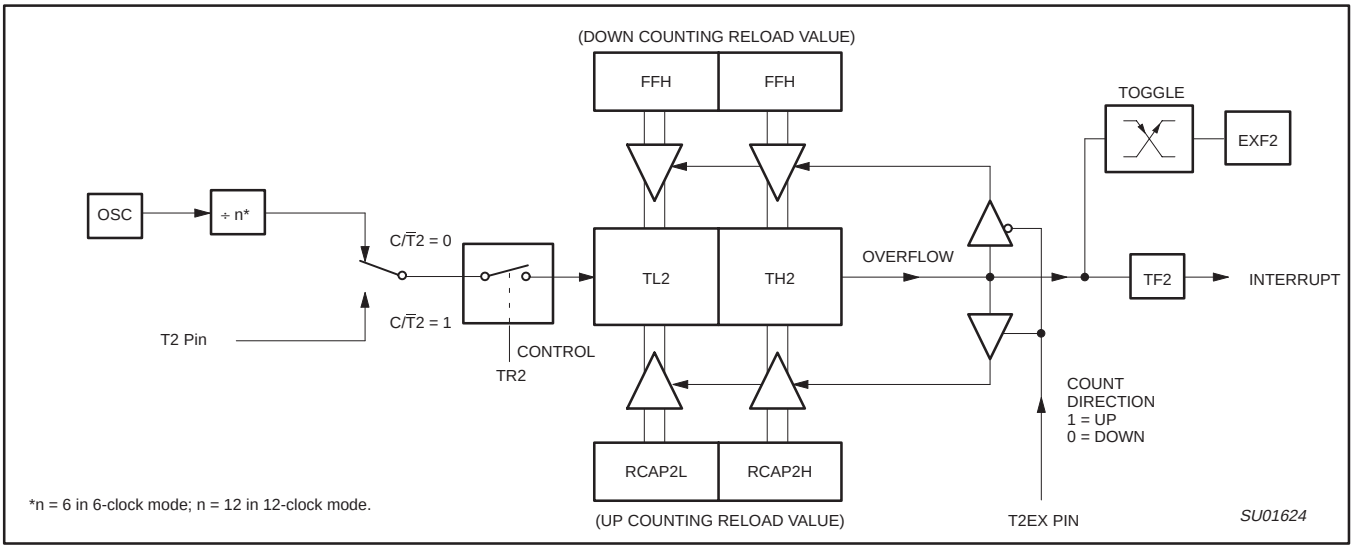



**Figure 10. Timer 2 Auto Reload Mode (DCEN = 1)**

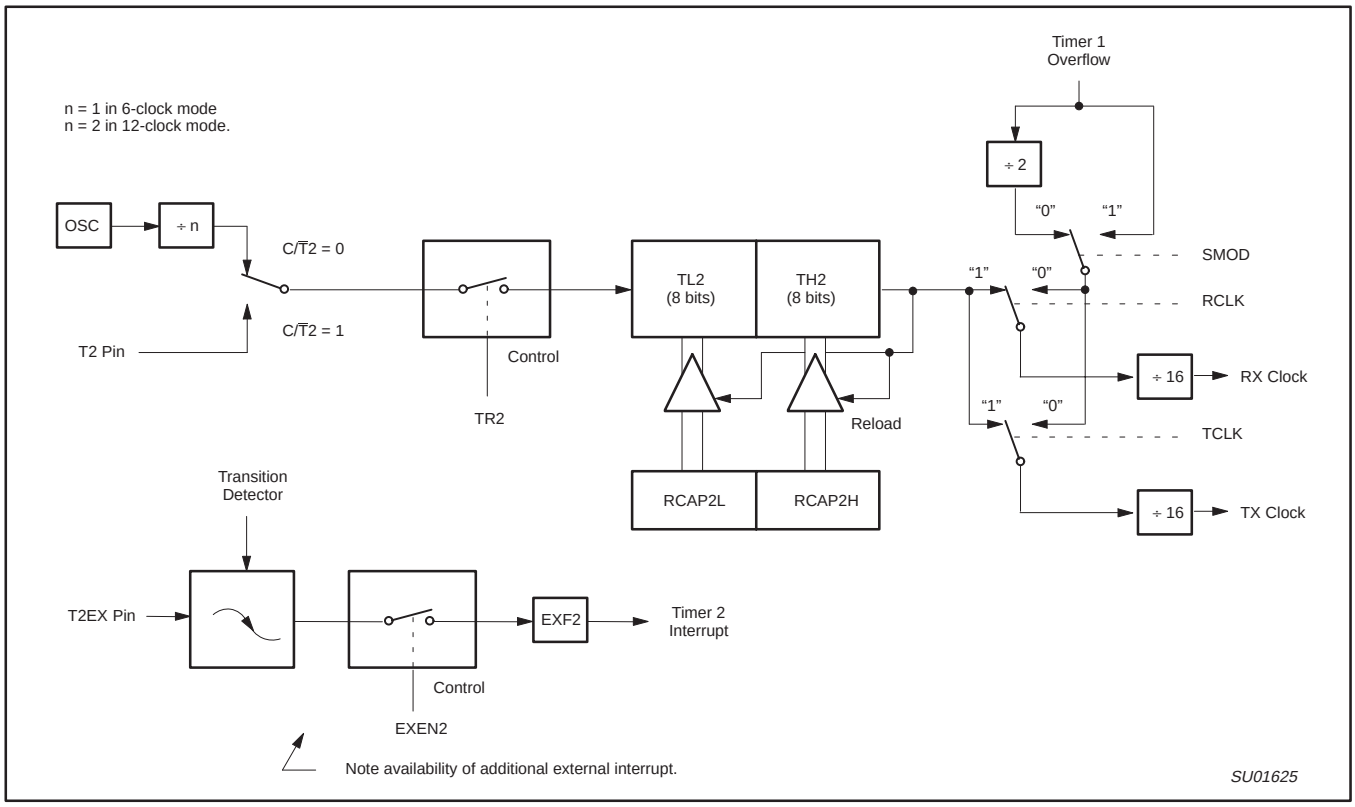



**Figure 11. Timer 2 in Baud Rate Generator Mode**

## Baud Rate Generator Mode

Bits TCLK and/or RCLK in T2CON (Table 4) 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 11 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/$\overline{T}$2=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., 1/6 the oscillator frequency in 6-clock mode or 1/12 the oscillator frequency in 12-clock mode). As a baud rate generator, it increments at the oscillator frequency in 6-clock mode or at 1/2 the oscillator frequency in 12-clock mode. Thus the baud rate formula is as follows:

$$\text{Modes 1 and 3 Baud Rates} = \frac{\text{Oscillator Frequency}}{[n \times [65536 - (\text{RCAP2H, RCAP2L})]]}$$

Where:

n = 16 in 6-clock mode, 32 in 12-clock mode.

(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 11 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 5 shows commonly used baud rates and how they can be obtained from Timer 2.

### Table 5. Timer 2 Generated Commonly Used Baud Rates

| Baud Rate | | Osc Freq | Timer 2 | |
|---|---|---|---|---|
| 12-clk mode | 6-clk mode | | RCAP2H | RCAP2L |
| 375 K | 750 K | 12 MHz | FF | FF |
| 9.6 K | 19.2 K | 12 MHz | FF | D9 |
| 4.8 K | 9.6 K | 12 MHz | FF | B2 |
| 2.4 K | 4.8 K | 12 MHz | FF | 64 |
| 1.2 K | 2.4 K | 12 MHz | FE | C8 |
| 300 | 600 | 12 MHz | FB | 1E |
| 110 | 220 | 12 MHz | F2 | AF |
| 300 | 600 | 6 MHz | FD | 8F |
| 110 | 220 | 6 MHz | F9 | 57 |

## 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:

$$\text{Baud Rate} = \frac{f_{OSC}}{[n \times [65536 - (RCAP2H, RCAP2L)]]}$$

Where:

n = 16 in 6-clock mode, 32 in 12-clock mode.

$f_{OSC}$= Oscillator Frequency

To obtain the reload value for RCAP2H and RCAP2L, the above equation can be rewritten as:

$$RCAP2H, RCAP2L = 65536 - \left(\frac{f_{OSC}}{n \times \text{Baud Rate}}\right)$$

## 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 6 for set-up of Timer 2 as a timer. Also see Table 7 for set-up of Timer 2 as a counter.

### Table 6. 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 7. 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.

## FULL-DUPLEX ENHANCED UART

### Standard UART operation

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 (in 12-clock mode) or 1/6 the oscillator frequency (in 6-clock mode).

**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 (in 12-clock mode) or 1/16 or 1/32 the oscillator frequency (in 6-clock mode).

**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 12. 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 (in 12-clock mode) or / 6 (in 6-clock mode). 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), and the port pins in 12-clock mode, the baud rate is 1/64 the oscillator frequency. If SMOD = 1, the baud rate is 1/32 the oscillator frequency. In 6-clock mode, the baud rate is 1/32 or 1/16 the oscillator frequency, respectively.

Mode 2 Baud Rate =

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

Where:

n = 64 in 12-clock mode, 32 in 6-clock mode

The baud rates in Modes 1 and 3 are determined by the Timer 1 or Timer 2 overflow rate.

**Using Timer 1 to Generate Baud Rates**

When Timer 1 is used as the baud rate generator (T2CON.RCLK = 0, T2CON.TCLK = 0), 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 =

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

Where:

n = 32 in 12-clock mode, 16 in 6-clock mode

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 =

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

Where:

n = 32 in 12-clock mode, 16 in 6-clock mode

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 13 lists various commonly used baud rates and how they can be obtained from Timer 1.

**SCON** Address = 98H Reset Value = 00H
Bit Addressable

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 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 | $f_{OSC}/12$ (12-clock mode) or $f_{OSC}/6$ (6-clock mode) |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{OSC}/64$ or $f_{OSC}/32$ (12-clock mode) or $f_{OSC}/32$ or $f_{OSC}/16$ (6-clock mode) |
| 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 Rl 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.

*SU01626*

**Figure 12. Serial Port Control (SCON) Register**

| Baud Rate | | | $f_{OSC}$ | SMOD | Timer 1 | | |
|---|---|---|---|---|---|---|---|
| Mode | 12-clock mode | 6-clock mode | | | C/$\overline{T}$ | Mode | Reload Value |
| Mode 0 Max | 1.67 MHz | 3.34 MHz | 20 MHz | X | X | X | X |
| Mode 2 Max | 625 k | 1250 k | 20 MHz | 1 | X | X | X |
| Mode 1, 3 Max | 104.2 k | 208.4 k | 20 MHz | 1 | 0 | 2 | FFH |
| Mode 1, 3 | 19.2 k | 38.4 k | 11.059 MHz | 1 | 0 | 2 | FDH |
| | 9.6 k | 19.2 k | 11.059 MHz | 0 | 0 | 2 | FDH |
| | 4.8 k | 9.6 k | 11.059 MHz | 0 | 0 | 2 | FAH |
| | 2.4 k | 4.8 k | 11.059 MHz | 0 | 0 | 2 | F4H |
| | 1.2 k | 2.4 k | 11.059 MHz | 0 | 0 | 2 | E8H |
| | 137.5 | 275 | 11.986 MHz | 0 | 0 | 2 | 1DH |
| | 110 | 220 | 6 MHz | 0 | 0 | 2 | 72H |
| | 110 | 220 | 12 MHz | 0 | 0 | 1 | FEEBH |

**Figure 13. 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 at 1/12 the oscillator frequency (12-clock mode) or 1/6 the oscillator frequency (6-clock mode).

Figure 14 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 or Timer 2 overflow rate.

Figure 15 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 deactivate 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 (12-clock mode) or 1/16 or 1/32 the oscillator frequency (6-clock mode) the oscillator frequency in Mode 2. Mode 3 may have a variable baud rate generated from Timer 1 or Timer 2.

Figures 16 and 17 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.

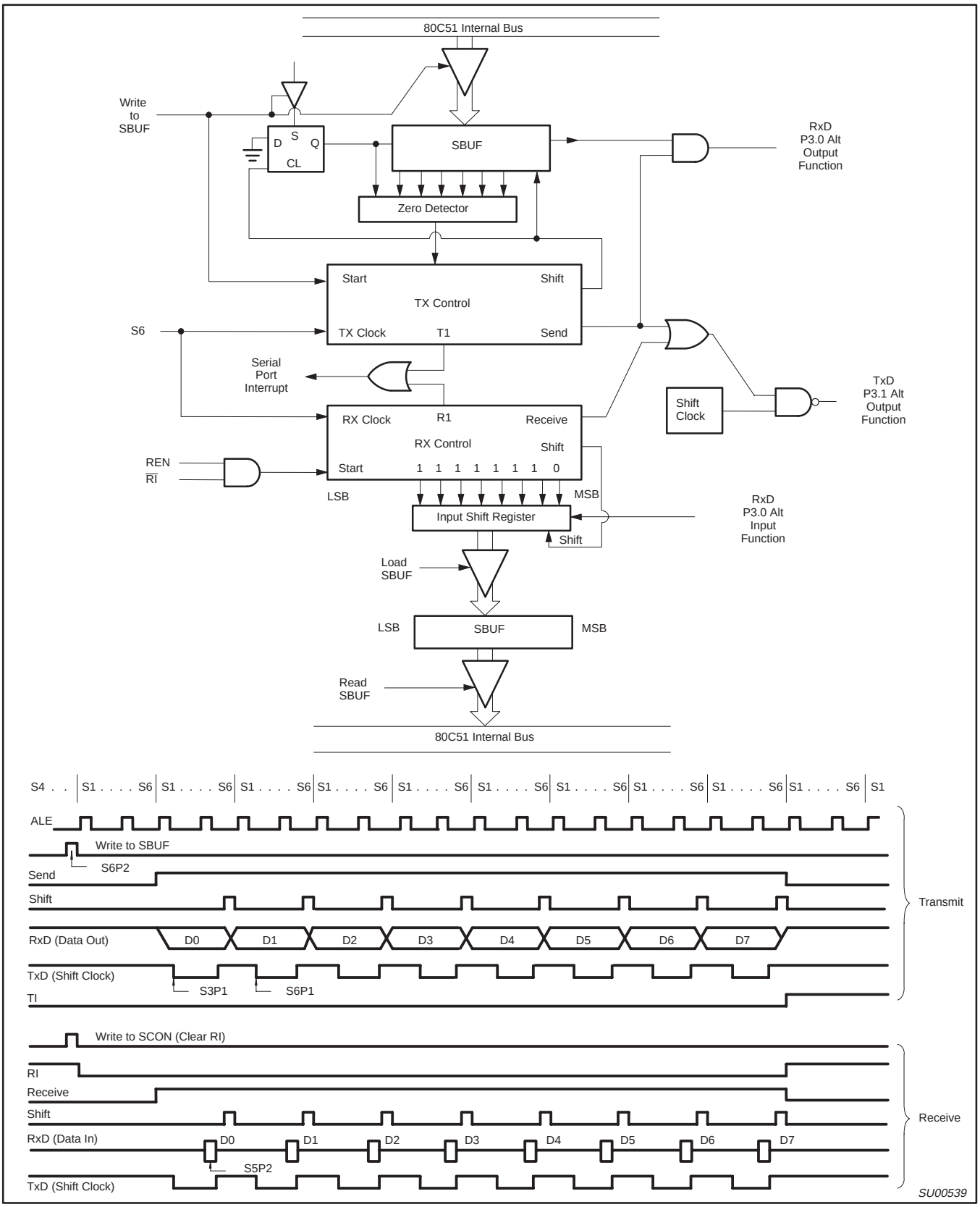



**Figure 14. Serial Port Mode 0**

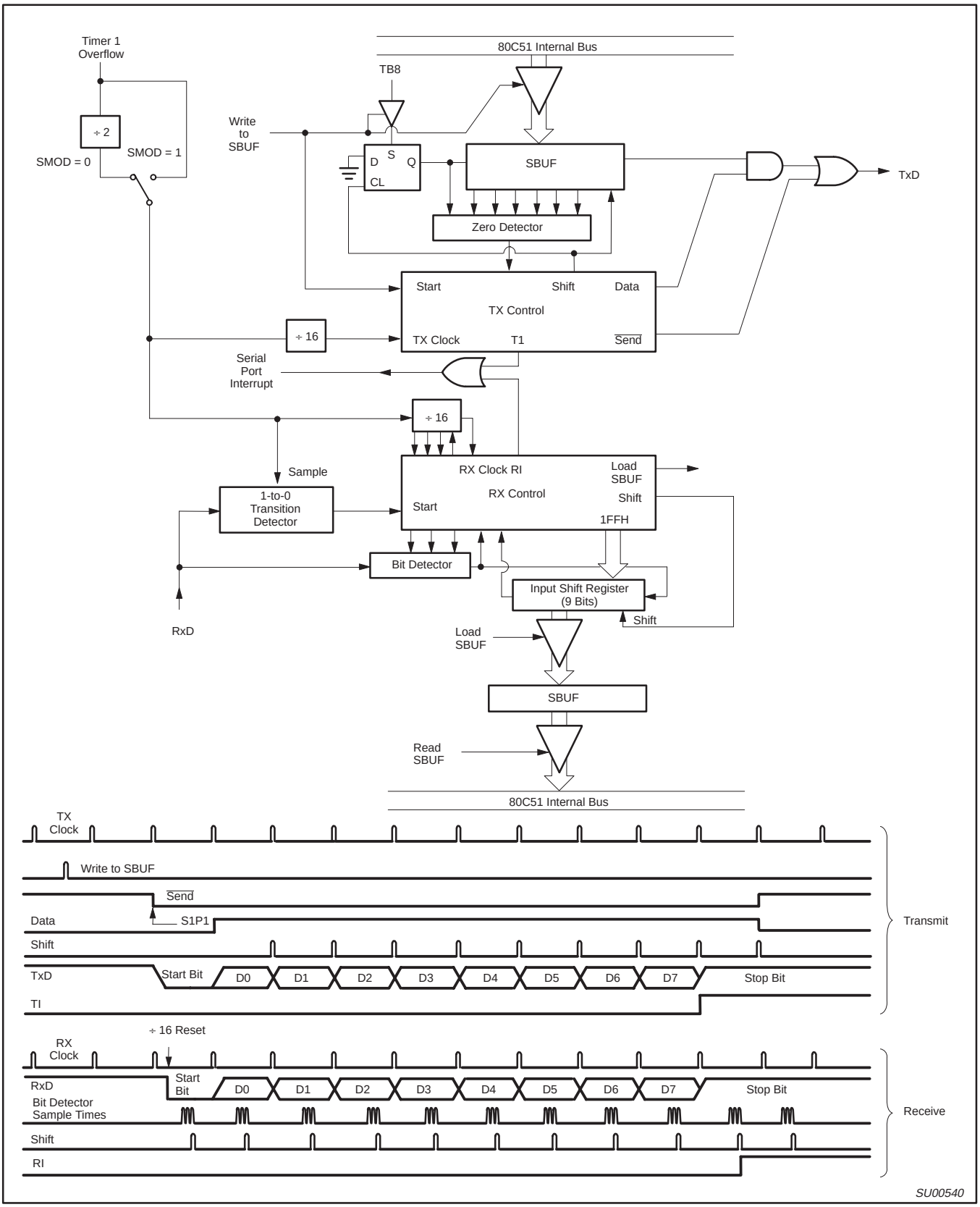



**Figure 15. Serial Port Mode 1**

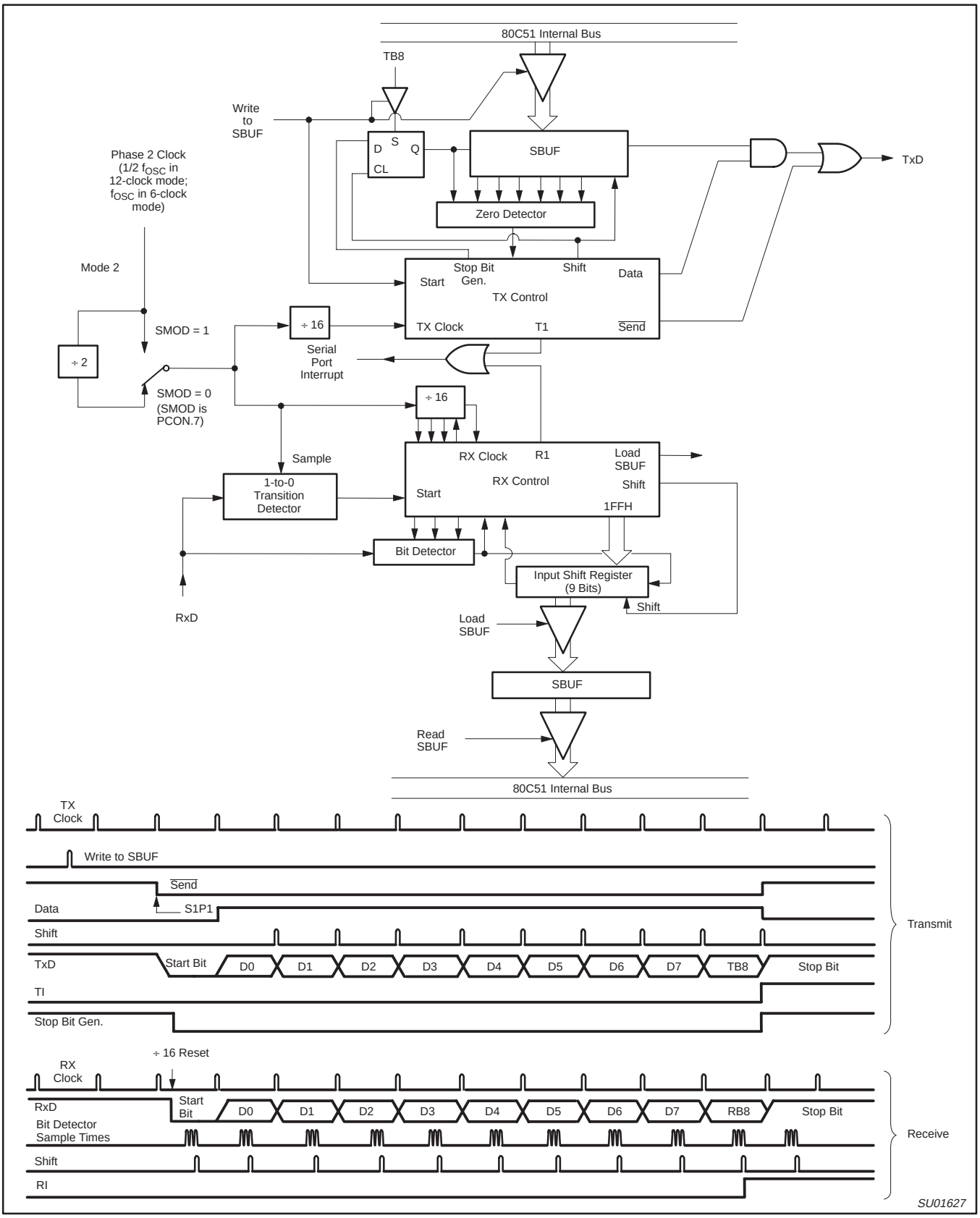



**Figure 16. Serial Port Mode 2**

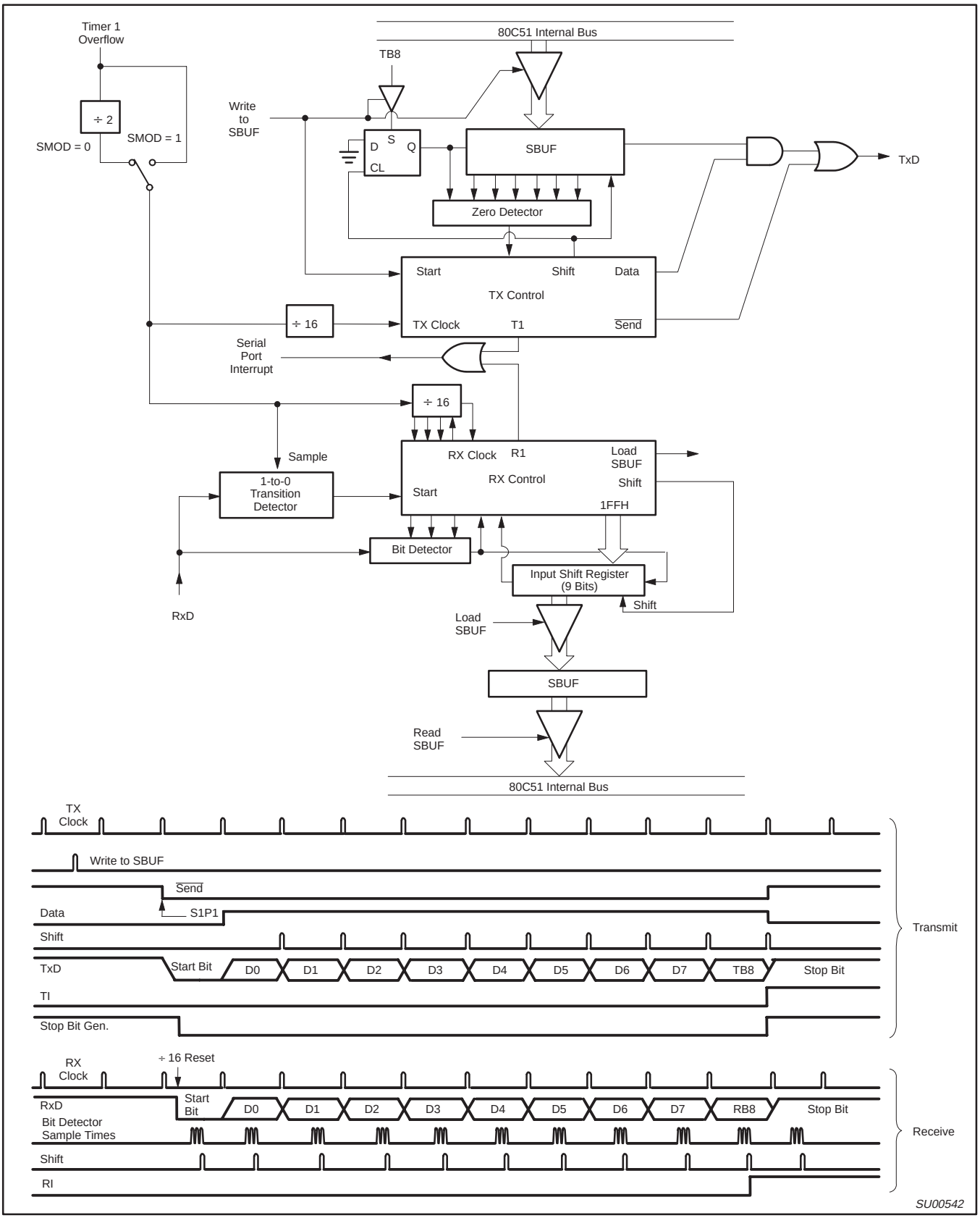



**Figure 17. Serial Port Mode 3**

## Enhanced UART operation

In addition to the standard operation modes, the UART can perform framing error detect by looking for missing stop bits, and automatic address recognition. The UART also fully supports multiprocessor communication.

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 18). 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.

### 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.

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 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 trended 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

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SM0/FE | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |

(SMOD0 = 0/1)*

| Symbol | Position | Function |
|---|---|---|
| FE | SCON.7 | 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 | SCON.7 | Serial Port Mode Bit 0, (SMOD0 must = 0 to access bit SM0) |
| SM1 | SCON.6 | Serial Port Mode Bit 1 |

| SM0 | SM1 | Mode | Description | Baud Rate** |
|---|---|---|---|---|
| 0 | 0 | 0 | shift register | $f_{OSC}/12$ (12-clk mode) or $f_{OSC}/6$ (6-clk mode) |
| 0 | 1 | 1 | 8-bit UART | variable |
| 1 | 0 | 2 | 9-bit UART | $f_{OSC}/64$ or $f_{OSC}/32$ or $f_{OSC}/16$ (6-clock mode) or $f_{OSC}/32$ (12-clock mode) |
| 1 | 1 | 3 | 9-bit UART | variable |

| Symbol | Position | Function |
|---|---|---|
| SM2 | SCON.5 | 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 | SCON.4 | Enables serial reception. Set by software to enable reception. Clear by software to disable reception. |
| TB8 | SCON.3 | The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired. |
| RB8 | SCON.2 | 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.<br>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, in any serial transmission. 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, in any serial reception (except see SM2). Must be cleared by software. |

**NOTES:**
*SMOD0 is located at PCON.6.
**$f_{OSC}$ = oscillator frequency

*SU01628*

**Figure 18. SCON: Serial Port Control Register**

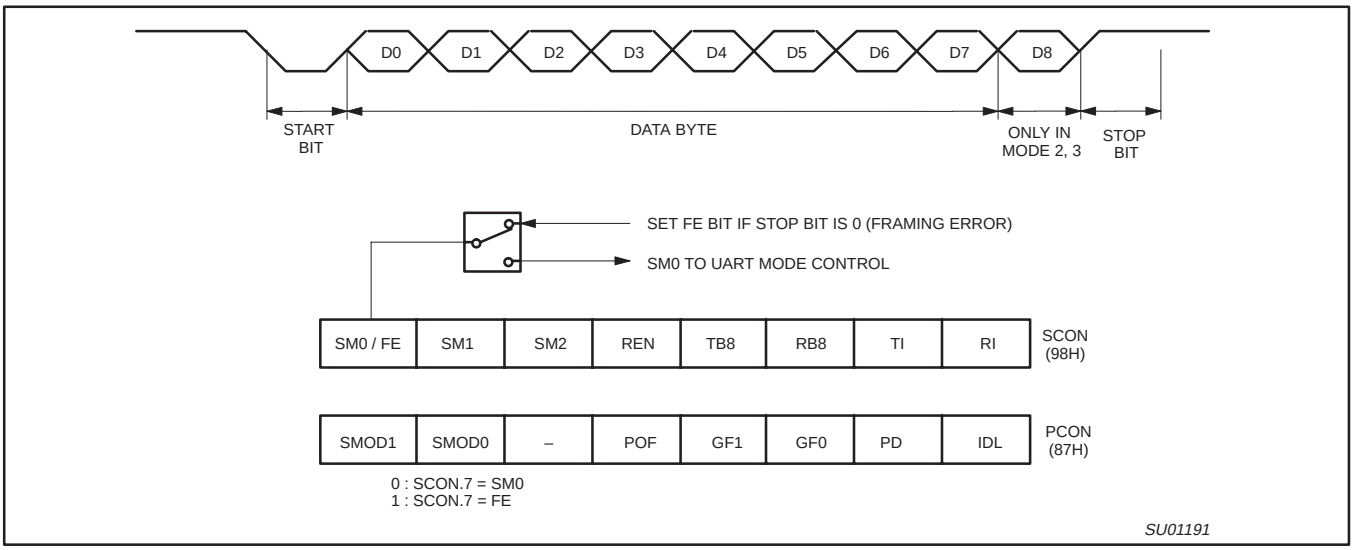



**Figure 19. UART Framing Error Detection**

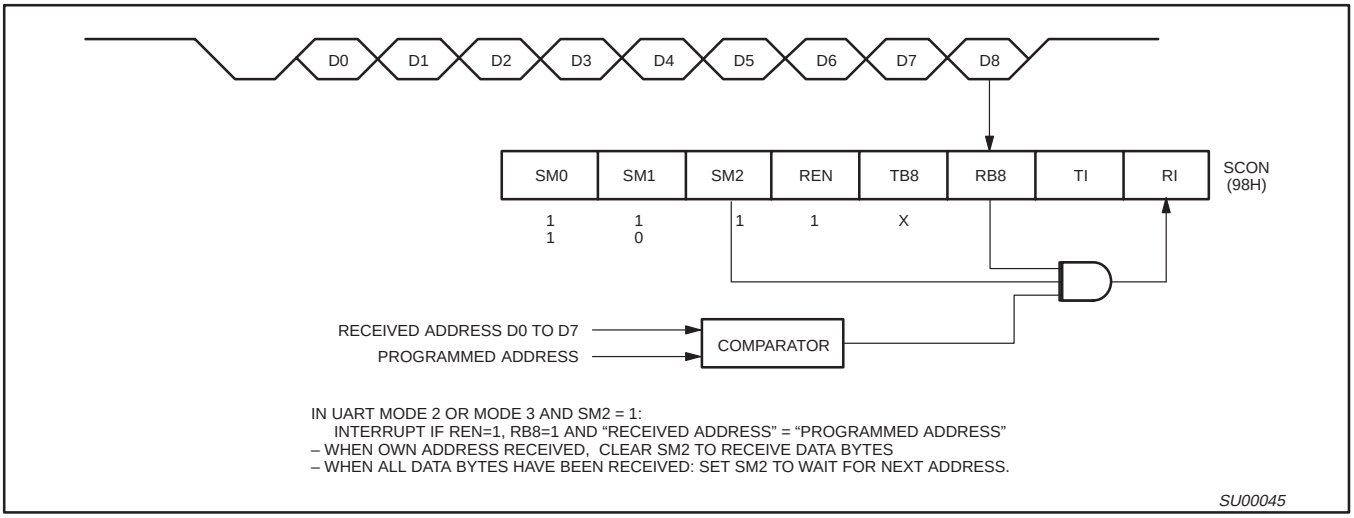



**Figure 20. UART Multiprocessor Communication, Automatic Address Recognition**

## Interrupt Priority Structure

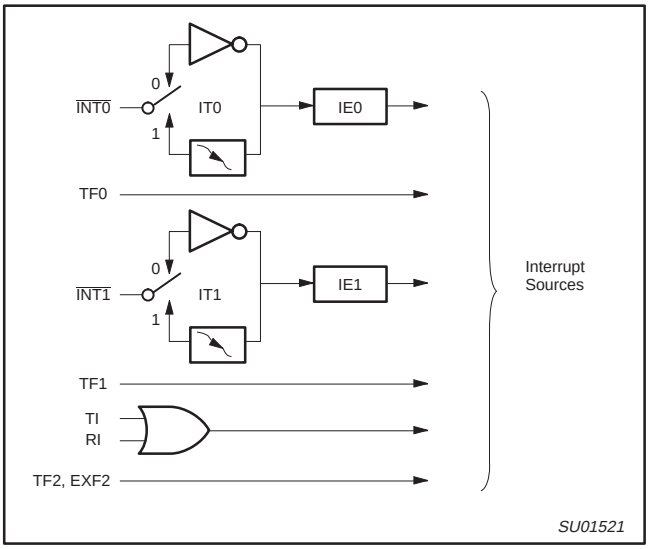



**Figure 21. Interrupt Sources**

## Interrupts

The devices described in this data sheet provide six interrupt sources. These are shown in Figure 21. 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 22). IE also contains a global disable bit, $\overline{EA}$, which disables all interrupts at once.

### Priority Level Structure

Each interrupt source can also be individually programmed to one of four priority levels by setting or clearing bits in Special Function Registers IP (Figure 23) and IPH (Figure 24). A lower-priority interrupt can itself be interrupted by a higher-priority interrupt, but not by another interrupt of the same level. A high-priority level 3 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 (External Int 0) | (highest) |
| 2. TF0 (Timer 0) | |
| 3. IE1 (External Int 1) | |
| 4. TF1 (Timer 1) | |
| 5. RI+TI (UART) | |
| 6. TF2, EXF2 (Timer 2) | (lowest) |

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

The IP and IPH registers contain a number of unimplemented bits. User software should not write 1s to these positions, since they may be used in other 80C51 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.

**IE** Address = 0A8H Reset Value = 0X000000B

Bit Addressable

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 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 | — | Not implemented. Reserved for future use. |
| 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. |

SU01522

**Figure 22. Interrupt Enable (IE) Register**

**IP** Address = 0B8H Reset Value = xx000000B

Bit Addressable

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| — | — | PT2 | PS | PT1 | PX1 | PT0 | PX0 |

Priority Bit = 1 assigns higher priority
Priority Bit = 0 assigns lower priority

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IP.7 | — | Not implemented, reserved for future use. |
| IP.6 | — | Not implemented, reserved for future use. |
| 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. |

SU01523

**Figure 23. Interrupt Priority (IP) Register**

**IPH** Address = B7H Reset Value = xx000000B

Bit Addressable

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| — | — | PT2H | PSH | PT1H | PX1H | PT0H | PX0H |

Priority Bit = 1 assigns higher priority
Priority Bit = 0 assigns lower priority

| BIT | SYMBOL | FUNCTION |
|---|---|---|
| IPH.7 | — | Not implemented, reserved for future use. |
| IPH.6 | — | Not implemented, reserved for future use. |
| IPH.5 | PT2H | Timer 2 interrupt priority bit high. |
| IPH.4 | PSH | Serial Port interrupt priority bit high. |
| IPH.3 | PT1H | Timer 1 interrupt priority bit high. |
| IPH.2 | PX1H | External interrupt 1 priority bit high. |
| IPH.1 | PT0H | Timer 0 interrupt priority bit high. |
| IPH.0 | PX0H | External interrupt 0 priority bit high. |

SU01524

**Figure 24. Interrupt Priority HIGH (IPH) Register**

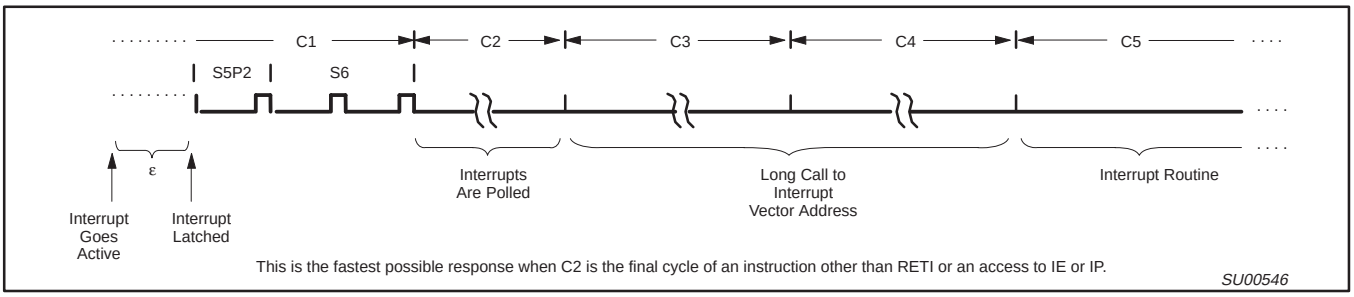



**Figure 25. Interrupt Response Timing Diagram**

The polling cycle/LCALL sequence is illustrated in Figure 25.

Note that if an interrupt of higher priority level goes active prior to S5P2 of the machine cycle labeled C3 in Figure 25, 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 in Table 8.

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 $\overline{\text{INT}}$x pin. If ITx = 1, external interrupt x is edge triggered. In this mode if successive samples of the $\overline{\text{INT}}$x 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 $\overline{\text{INT0}}$ and $\overline{\text{INT1}}$ 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 25 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.

As previously mentioned, the derivatives described in this data sheet have a four-level interrupt structure. The corresponding registers are IE, IP and IPH. (See Figures 22, 23, and 24.) The IPH (Interrupt Priority High) register makes the four-level interrupt structure possible.

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) |

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 8. Interrupt Table**

| SOURCE | POLLING PRIORITY | REQUEST BITS | HARDWARE CLEAR? | VECTOR ADDRESS |
|---|---|---|---|---|
| External interrupt 0 | 1 | IE0 | N (L)[1] Y (T)[2] | 03H |
| Timer 0 | 2 | TF0 | Y | 0BH |
| External interrupt 1 | 3 | IE1 | N (L) Y (T) | 13H |
| Timer 1 | 4 | TF1 | Y | 1BH |
| UART | 5 | RI, TI | N | 23H |
| Timer 2 | 6 | TF2, EXF2 | N | 2BH |

**NOTES:**
1. L = Level activated
2. T = Transition activated

## Reduced EMI

All port pins have slew rate controlled outputs. This is to limit noise generated by quickly switching output signals. The slew rate is factory set to approximately 10 ns rise and fall times.

## Reduced EMI Mode

The AO bit (AUXR.0) in the AUXR register when set disables the ALE output.

**AUXR (8EH)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | – | – | – | AO |

AUXR.0 AO Turns off ALE output.

## Dual DPTR

The dual DPTR structure (see Figure 26) enables a way to 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.

- New Register Name: AUXR1#
- SFR Address: A2H
- Reset Value: xxx000x0B

**AUXR1 (A2H)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | WUPD | 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.

Note that bit 2 is not writable and is always read as a zero. This allows the DPS bit to be quickly toggled simply by executing an INC DPTR instruction without affecting the WUPD bit.

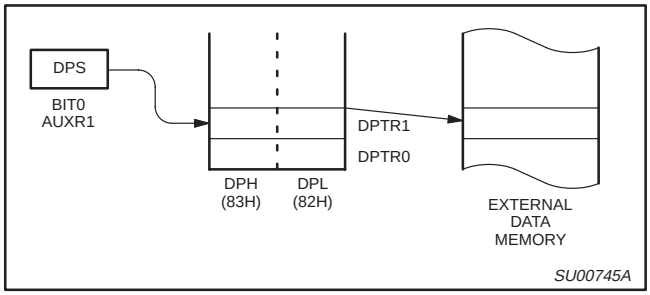



**Figure 26.**

**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 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 maximum.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to $V_{SS}$ unless otherwise noted.

## AC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0°C to +70°C or –40°C to +85°C

| SYMBOL | FIGURE | PARAMETER | OPERATING MODE | POWER SUPPLY VOLTAGE | CLOCK FREQUENCY RANGE | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | | | MIN | MAX | |
| $1/t_{CLCL}$ | 31 | Oscillator frequency | 6-clock | 5 V ± 10% | 0 | 20 | MHz |
| | | | 12-clock | 5 V ± 10% | 0 | 33 | MHz |

## DC ELECTRICAL CHARACTERISTICS

$T_{amb}$ = 0 °C to +70 °C or –40 °C to +85 °C; $V_{CC}$ = 5 V ±10%; $V_{SS}$ = 0 V (20/33 MHz max. CPU clock)

| SYMBOL | PARAMETER | TEST CONDITIONS | LIMITS | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP[1] | MAX | |
| $V_{IL}$ | Input low voltage[11] | 4.5 V < $V_{CC}$ < 5.5 V | –0.5 | | 0.2 $V_{CC}$–0.1 | 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[11] | – | 0.7 $V_{CC}$ | | $V_{CC}$+0.5 | V |
| $V_{OL}$ | Output low voltage, ports 1, 2, 3 [8] | $V_{CC}$ = 4.5 V; $I_{OL}$ = 1.6 mA[2] | – | | 0.4 | V |
| $V_{OL1}$ | Output low voltage, port 0, ALE, $\overline{PSEN}$ [7, 8] | $V_{CC}$ = 4.5 V; $I_{OL}$ = 3.2 mA[2] | – | | 0.4 | V |
| $V_{OH}$ | Output high voltage, ports 1, 2, 3 [3] | $V_{CC}$ = 4.5 V; $I_{OH}$ = –30 μA | $V_{CC}$ – 0.7 | | – | V |
| $V_{OH1}$ | Output high voltage (port 0 in external bus mode), ALE[9], $\overline{PSEN}$[3] | $V_{CC}$ = 4.5 V; $I_{OH}$ = –3.2 mA | $V_{CC}$ – 0.7 | | – | V |
| $I_{IL}$ | Logical 0 input current, ports 1, 2, 3 | $V_{IN}$ = 0.4 V | –1 | | –50 | μA |
| $I_{TL}$ | Logical 1-to-0 transition current, ports 1, 2, 3[6] | $V_{IN}$ = 2.0 V; 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 34): | | | | | |
| | Active mode (see Note 5) | | | | | |
| | Idle mode (see Note 5) | | | | | |
| | Power-down mode or clock stopped (see Figure 38 for conditions) | $T_{amb}$ = 0 °C to 70 °C | | 15 | 100 | μA |
| | | $T_{amb}$ = –40 °C to +85 °C | | | 125 | 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, 5 V.
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 > 100 pF), 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. $I_{OL}$ can exceed these conditions provided that no single output sinks more than 5 mA 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 2 V.
5. See Figures 35 through 38 for $I_{CC}$ test conditions and Figure 34 for $I_{CC}$ vs. Frequency.
   12-clock mode characteristics:
   Active mode: $I_{CC}$ (MAX) = (8.5 + 0.62 × FREQ. [MHz])mA
   Idle mode: $I_{CC}$ (MAX) = (3.5 + 0.18 × FREQ. [MHz])mA
6. This value applies to $T_{amb}$ = 0°C to +70°C. For $T_{amb}$ = –40°C to +85°C, $I_{TL}$ = –750 μA.
7. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100 pF, load capacitance for all other outputs = 80 pF.
8. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   Maximum $I_{OL}$ per port pin: 15 mA (*NOTE: This is 85 °C specification.)
   Maximum $I_{OL}$ per 8-bit port: 26 mA
   Maximum total $I_{OL}$ for all outputs: 71 mA
   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 25 pF. Pin capacitance of ceramic package is less than 15 pF (except $\overline{EA}$ is 25 pF).
11. To improve noise rejection a nominal 100 ns glitch rejection circuitry has been added to the RST pin, and a nominal 15 ns glitch rejection circuitry has been added to the $\overline{INT0}$ and $\overline{INT1}$ pins. Previous devices provided only an inherent 5 ns of glitch rejection.

## AC ELECTRICAL CHARACTERISTICS (12-CLOCK MODE)

$T_{amb}$ = 0 °C to +70 °C or –40 °C to +85 °C ; $V_{CC}$ = 5 V ±10%, $V_{SS}$ = 0 V[1,2,3,4]

| Symbol | Figure | Parameter | Limits | | 16 MHz Clock | | Unit |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 31 | Oscillator frequency | 0 | 33 | | | MHz |
| $t_{LHLL}$ | 27 | ALE pulse width | 2 $t_{CLCL}$–8 | | 117 | | ns |
| $t_{AVLL}$ | 27 | Address valid to ALE low | $t_{CLCL}$ –13 | | 49.5 | | ns |
| $t_{LLAX}$ | 27 | Address hold after ALE low | $t_{CLCL}$ –20 | | 42.5 | | ns |
| $t_{LLIV}$ | 27 | ALE low to valid instruction in | | 4 $t_{CLCL}$ –35 | | 215 | ns |
| $t_{LLPL}$ | 27 | ALE low to $\overline{PSEN}$ low | $t_{CLCL}$ –10 | | 52.5 | | ns |
| $t_{PLPH}$ | 27 | $\overline{PSEN}$ pulse width | 3 $t_{CLCL}$ –10 | | 177.5 | | ns |
| $t_{PLIV}$ | 27 | $\overline{PSEN}$ low to valid instruction in | | 3 $t_{CLCL}$ –35 | | 152.5 | ns |
| $t_{PXIX}$ | 27 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 27 | Input instruction float after $\overline{PSEN}$ | | $t_{CLCL}$ –10 | | 52.5 | ns |
| $t_{AVIV}$ | 27 | Address to valid instruction in | | 5 $t_{CLCL}$ –35 | | 277.5 | ns |
| $t_{PLAZ}$ | 27 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 28 | $\overline{RD}$ pulse width | 6 $t_{CLCL}$ –20 | | 355 | | ns |
| $t_{WLWH}$ | 29 | $\overline{WR}$ pulse width | 6 $t_{CLCL}$ –20 | | 355 | | ns |
| $t_{RLDV}$ | 28 | $\overline{RD}$ low to valid data in | | 5 $t_{CLCL}$ –35 | | 277.5 | ns |
| $t_{RHDX}$ | 28 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 28 | Data float after $\overline{RD}$ | | 2 $t_{CLCL}$ –10 | | 115 | ns |
| $t_{LLDV}$ | 28 | ALE low to valid data in | | 8 $t_{CLCL}$ –35 | | 465 | ns |
| $t_{AVDV}$ | 28 | Address to valid data in | | 9 $t_{CLCL}$ –35 | | 527.5 | ns |
| $t_{LLWL}$ | 28, 29 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 3 $t_{CLCL}$ –15 | 3 $t_{CLCL}$ +15 | 172.5 | 202.5 | ns |
| $t_{AVWL}$ | 28, 29 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 4 $t_{CLCL}$ –15 | | 235 | | ns |
| $t_{QVWX}$ | 29 | Data valid to $\overline{WR}$ transition | $t_{CLCL}$ –25 | | 37.5 | | ns |
| $t_{WHQX}$ | 29 | Data hold after $\overline{WR}$ | $t_{CLCL}$ –15 | | 47.5 | | ns |
| $t_{QVWH}$ | 29 | Data valid to $\overline{WR}$ high | 7 $t_{CLCL}$ –5 | | 432.5 | | ns |
| $t_{RLAZ}$ | 28 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 28, 29 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | $t_{CLCL}$ –10 | $t_{CLCL}$ +10 | 52.5 | 72.5 | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 31 | High time | 0.32 $t_{CLCL}$ | $t_{CLCL}$ – $t_{CLCX}$ | | | ns |
| $t_{CLCX}$ | 31 | Low time | 0.32 $t_{CLCL}$ | $t_{CLCL}$ – $t_{CHCX}$ | | | ns |
| $t_{CLCH}$ | 31 | Rise time | | 5 | | | ns |
| $t_{CHCL}$ | 31 | Fall time | | 5 | | | ns |
| **Shift register** | | | | | | | |
| $t_{XLXL}$ | 30 | Serial port clock cycle time | 12 $t_{CLCL}$ | | 750 | | ns |
| $t_{QVXH}$ | 30 | Output data setup to clock rising edge | 10 $t_{CLCL}$ –25 | | 600 | | ns |
| $t_{XHQX}$ | 30 | Output data hold after clock rising edge | 2 $t_{CLCL}$ –15 | | 110 | | ns |
| $t_{XHDX}$ | 30 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 30 | Clock rising edge to input data valid | | 10 $t_{CLCL}$ –133 | | 492 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100 pF, load capacitance for all outputs = 80 pF
3. Interfacing the microcontroller to devices with float time up to 45 ns is permitted. This limited bus contention will not cause damage to port 0 drivers.
4. Parts are guaranteed by design to operate down to 0 Hz.

## AC ELECTRICAL CHARACTERISTICS (6-CLOCK MODE)

$T_{amb}$ = 0 °C to +70 °C or –40 °C to +85 °C ; $V_{CC}$ = 5 V ±10%, $V_{SS}$ = 0 V[1,2,3,4,5]

| Symbol | Figure | Parameter | Limits | | 16 MHz Clock | | Unit |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $1/t_{CLCL}$ | 31 | Oscillator frequency | 0 | 20 | | | MHz |
| $t_{LHLL}$ | 27 | ALE pulse width | $t_{CLCL}$–8 | | 54.5 | | ns |
| $t_{AVLL}$ | 27 | Address valid to ALE low | 0.5 $t_{CLCL}$ –13 | | 18.25 | | ns |
| $t_{LLAX}$ | 27 | Address hold after ALE low | 0.5 $t_{CLCL}$ –20 | | 11.25 | | ns |
| $t_{LLIV}$ | 27 | ALE low to valid instruction in | | 2 $t_{CLCL}$ –35 | | 90 | ns |
| $t_{LLPL}$ | 27 | ALE low to $\overline{PSEN}$ low | 0.5 $t_{CLCL}$ –10 | | 21.25 | | ns |
| $t_{PLPH}$ | 27 | $\overline{PSEN}$ pulse width | 1.5 $t_{CLCL}$ –10 | | 83.75 | | ns |
| $t_{PLIV}$ | 27 | $\overline{PSEN}$ low to valid instruction in | | 1.5 $t_{CLCL}$ –35 | | 58.75 | ns |
| $t_{PXIX}$ | 27 | Input instruction hold after $\overline{PSEN}$ | 0 | | 0 | | ns |
| $t_{PXIZ}$ | 27 | Input instruction float after $\overline{PSEN}$ | | 0.5 $t_{CLCL}$ –10 | | 21.25 | ns |
| $t_{AVIV}$ | 27 | Address to valid instruction in | | 2.5 $t_{CLCL}$ –35 | | 121.25 | ns |
| $t_{PLAZ}$ | 27 | $\overline{PSEN}$ low to address float | | 10 | | 10 | ns |
| **Data Memory** | | | | | | | |
| $t_{RLRH}$ | 28 | $\overline{RD}$ pulse width | 3 $t_{CLCL}$ –20 | | 167.5 | | ns |
| $t_{WLWH}$ | 29 | $\overline{WR}$ pulse width | 3 $t_{CLCL}$ –20 | | 167.5 | | ns |
| $t_{RLDV}$ | 28 | $\overline{RD}$ low to valid data in | | 2.5 $t_{CLCL}$ –35 | | 121.25 | ns |
| $t_{RHDX}$ | 28 | Data hold after $\overline{RD}$ | 0 | | 0 | | ns |
| $t_{RHDZ}$ | 28 | Data float after $\overline{RD}$ | | $t_{CLCL}$ –10 | | 52.5 | ns |
| $t_{LLDV}$ | 28 | ALE low to valid data in | | 4 $t_{CLCL}$ –35 | | 215 | ns |
| $t_{AVDV}$ | 28 | Address to valid data in | | 4.5 $t_{CLCL}$ –35 | | 246.25 | ns |
| $t_{LLWL}$ | 28, 29 | ALE low to $\overline{RD}$ or $\overline{WR}$ low | 1.5 $t_{CLCL}$ –15 | 1.5 $t_{CLCL}$ +15 | 78.75 | 108.75 | ns |
| $t_{AVWL}$ | 28, 29 | Address valid to $\overline{WR}$ low or $\overline{RD}$ low | 2 $t_{CLCL}$ –15 | | 110 | | ns |
| $t_{QVWX}$ | 29 | Data valid to $\overline{WR}$ transition | 0.5 $t_{CLCL}$ –25 | | 6.25 | | ns |
| $t_{WHQX}$ | 29 | Data hold after $\overline{WR}$ | 0.5 $t_{CLCL}$ –15 | | 16.25 | | ns |
| $t_{QVWH}$ | 29 | Data valid to $\overline{WR}$ high | 3.5 $t_{CLCL}$ –5 | | 213.75 | | ns |
| $t_{RLAZ}$ | 28 | $\overline{RD}$ low to address float | | 0 | | 0 | ns |
| $t_{WHLH}$ | 28, 29 | $\overline{RD}$ or $\overline{WR}$ high to ALE high | 0.5 $t_{CLCL}$ –10 | 0.5 $t_{CLCL}$ +10 | 21.25 | 41.25 | ns |
| **External Clock** | | | | | | | |
| $t_{CHCX}$ | 31 | High time | 0.4 $t_{CLCL}$ | $t_{CLCL}$ – $t_{CLCX}$ | | | ns |
| $t_{CLCX}$ | 31 | Low time | 0.4 $t_{CLCL}$ | $t_{CLCL}$ – $t_{CHCX}$ | | | ns |
| $t_{CLCH}$ | 31 | Rise time | | 5 | | | ns |
| $t_{CHCL}$ | 31 | Fall time | | 5 | | | ns |
| **Shift register** | | | | | | | |
| $t_{XLXL}$ | 30 | Serial port clock cycle time | 6 $t_{CLCL}$ | | 375 | | ns |
| $t_{QVXH}$ | 30 | Output data setup to clock rising edge | 5 $t_{CLCL}$ –25 | | 287.5 | | ns |
| $t_{XHQX}$ | 30 | Output data hold after clock rising edge | $t_{CLCL}$ –15 | | 47.5 | | ns |
| $t_{XHDX}$ | 30 | Input data hold after clock rising edge | 0 | | 0 | | ns |
| $t_{XHDV}$ | 30 | Clock rising edge to input data valid | | 5 $t_{CLCL}$ –133 | | 179.5 | ns |

**NOTES:**
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$=100 pF, load capacitance for all outputs = 80 pF
3. Interfacing the microcontroller to devices with float time up to 45ns is permitted. This limited bus contention will not cause damage to port 0 drivers.
4. Parts are guaranteed by design to operate down to 0 Hz.
5. Data shown in the table are the best mathematical models for the set of measured values obtained in tests. If a particular parameter calculated at a customer specified frequency has a negative value, it should be considered equal to zero.

## 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.

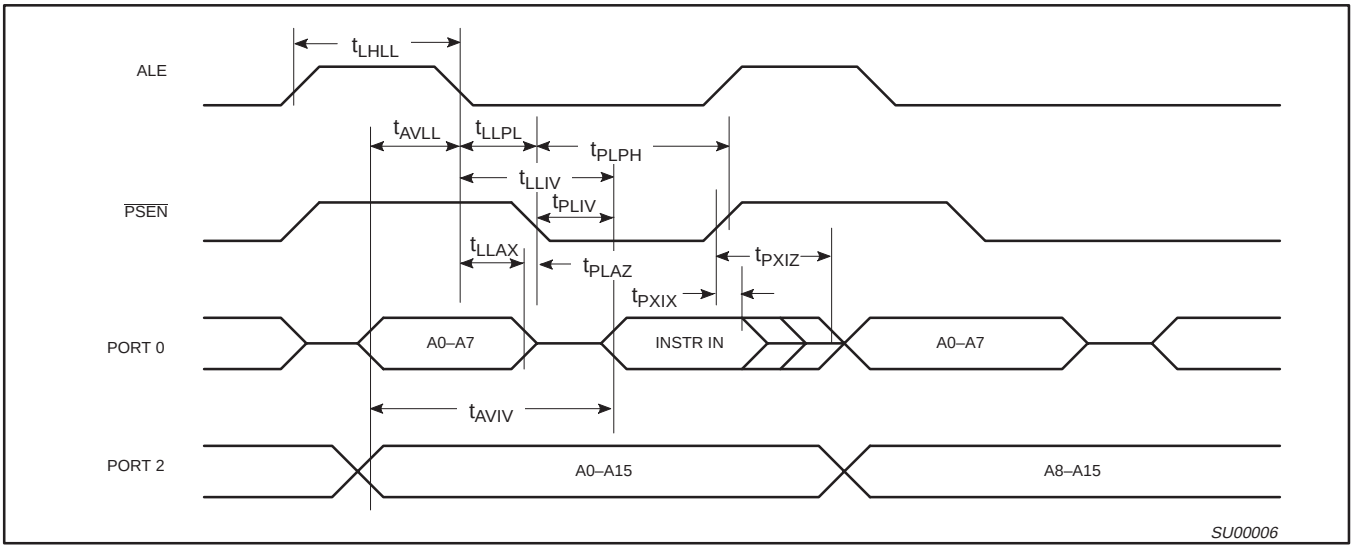



**Figure 27. External Program Memory Read Cycle**

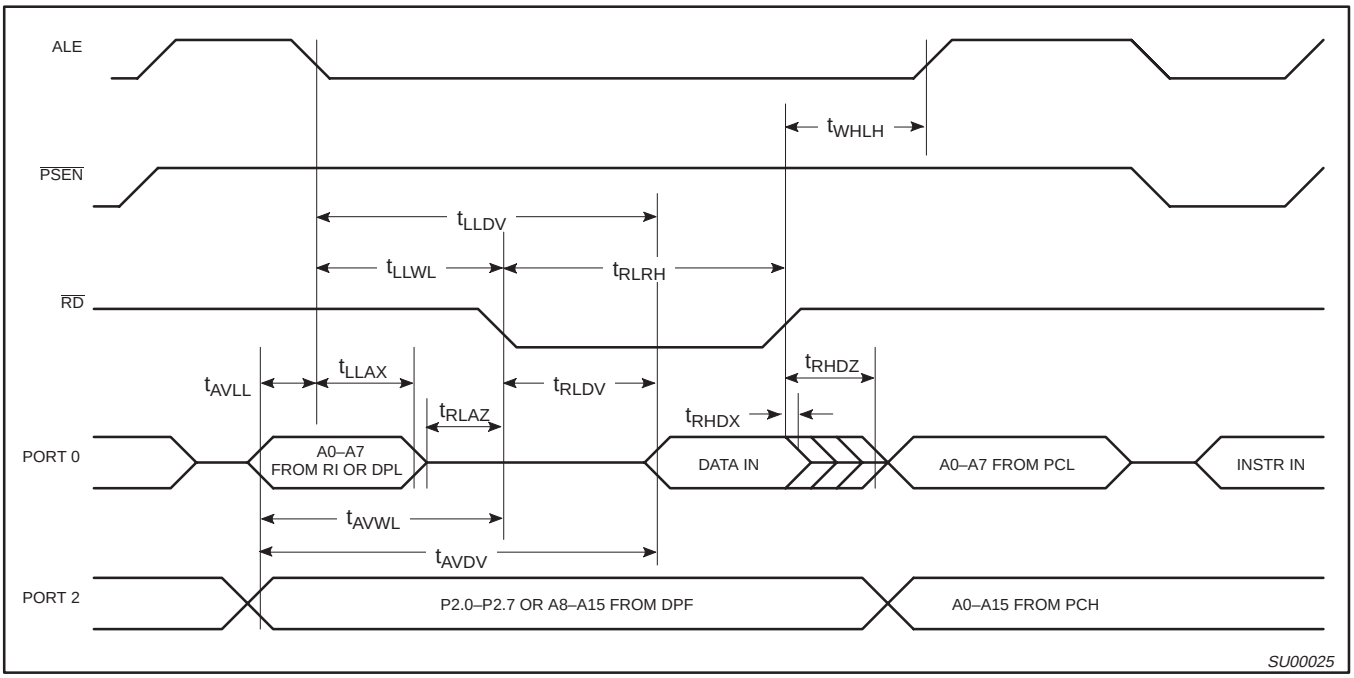



**Figure 28. External Data Memory Read Cycle**

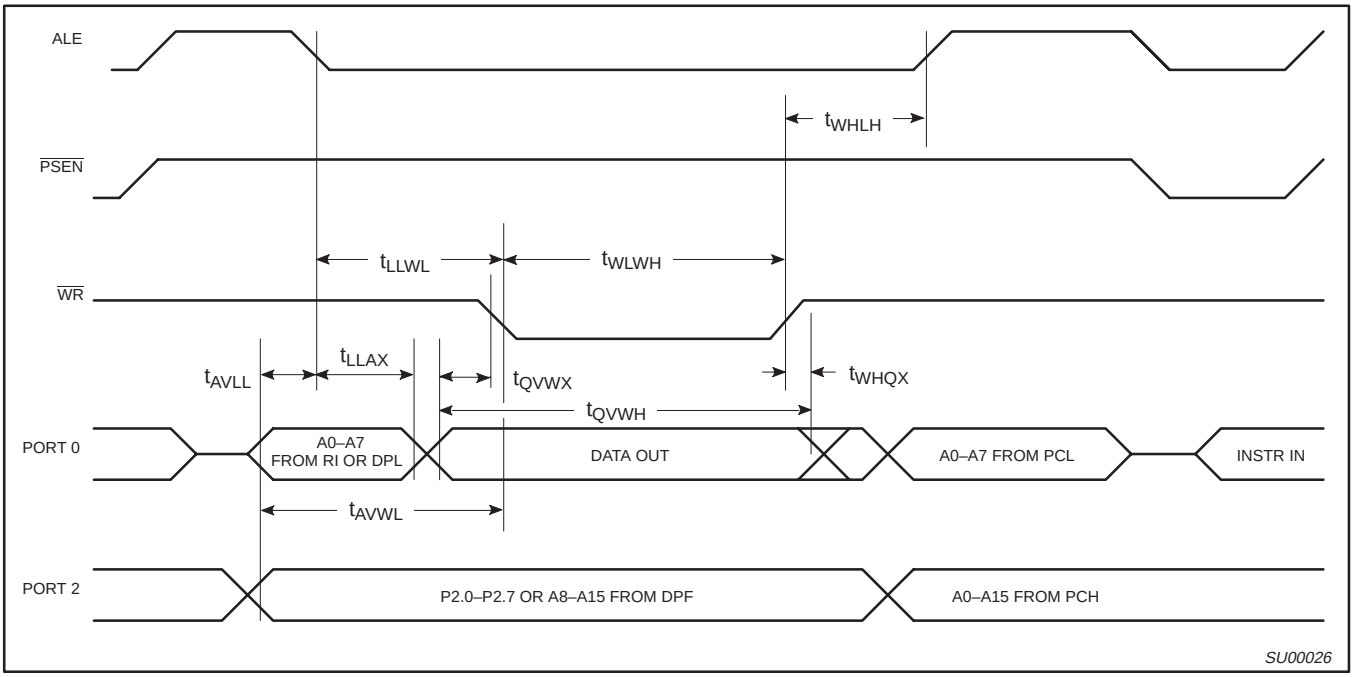



**Figure 29. External Data Memory Write Cycle**

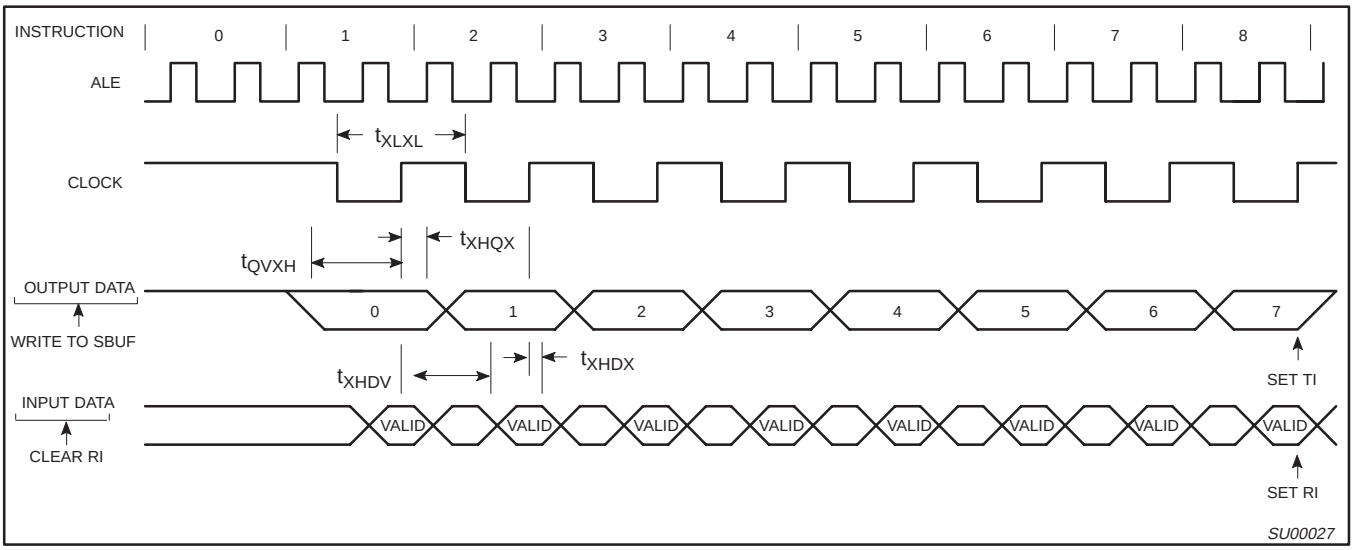



**Figure 30. Shift Register Mode Timing**

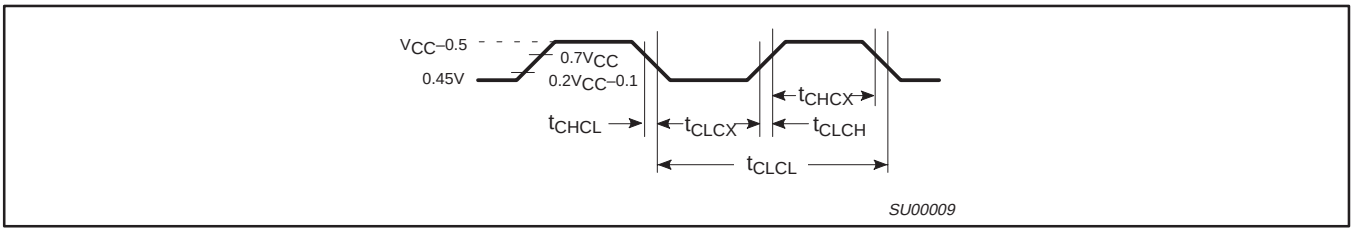
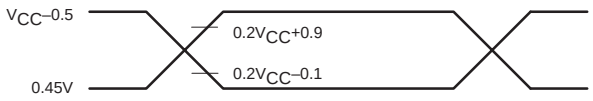



**Figure 31. External Clock Drive**




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'.

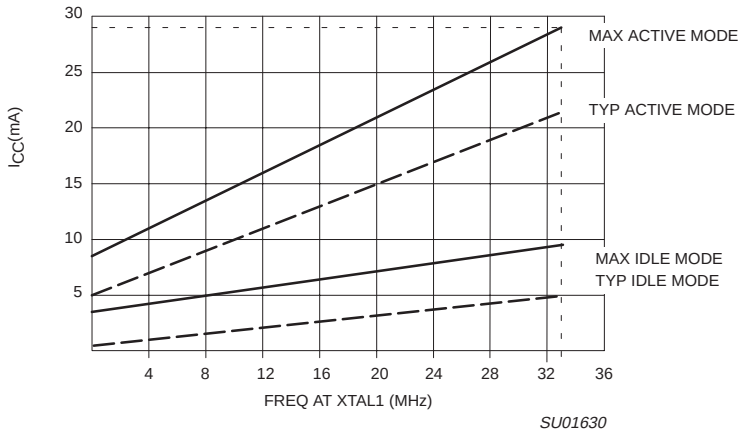
*SU00717*

**Figure 32. 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 33. Float Waveform**




*SU01630*

**Figure 34. $I_{CC}$ vs. FREQ for 12-clock operation**
**Valid only within frequency specifications**

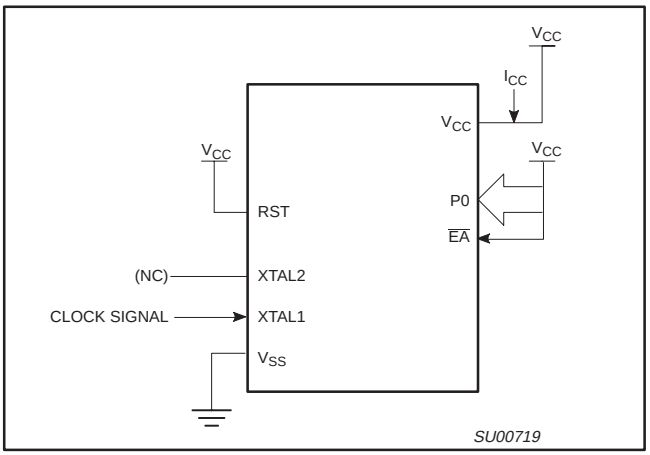



**Figure 35. $I_{CC}$ Test Condition, Active Mode**
**All other pins are disconnected**

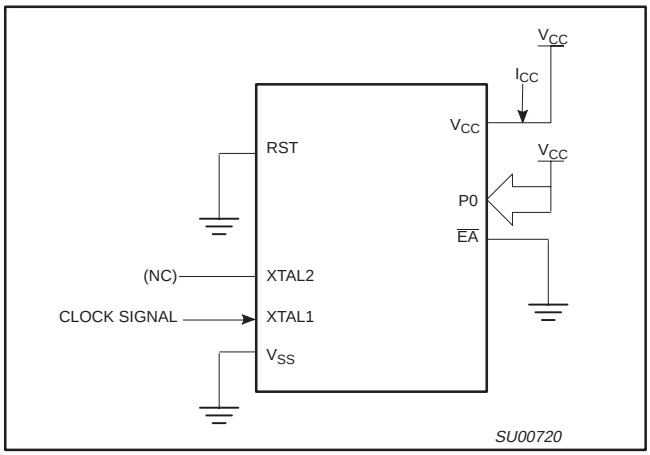



**Figure 36. $I_{CC}$ Test Condition, Idle Mode**
**All other pins are disconnected**

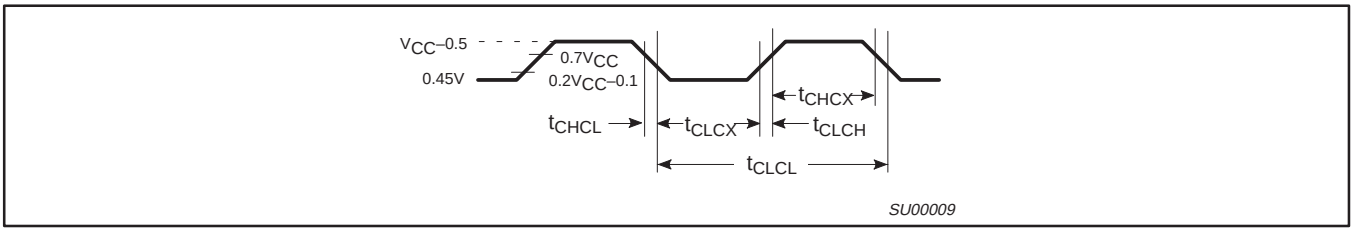



**Figure 37. Clock Signal Waveform for $I_{CC}$ Tests in Active and Idle Modes**
**$t_{CLCH} = t_{CHCL} = 5ns$**

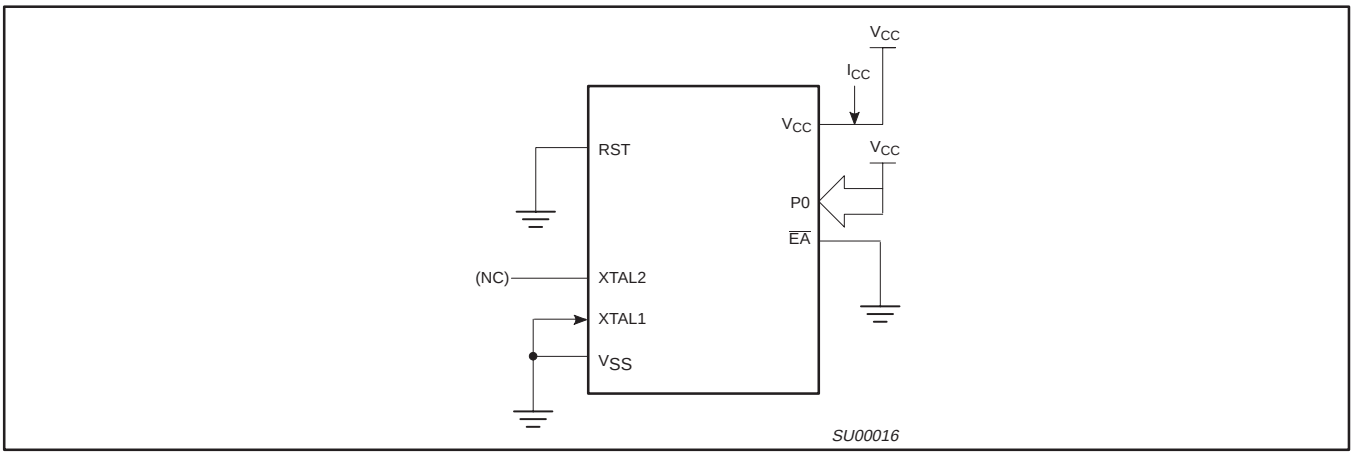



**Figure 38. $I_{CC}$ Test Condition, Power Down Mode**
**All other pins are disconnected. $V_{CC}$ = 2 V to 5.5 V**

## Security

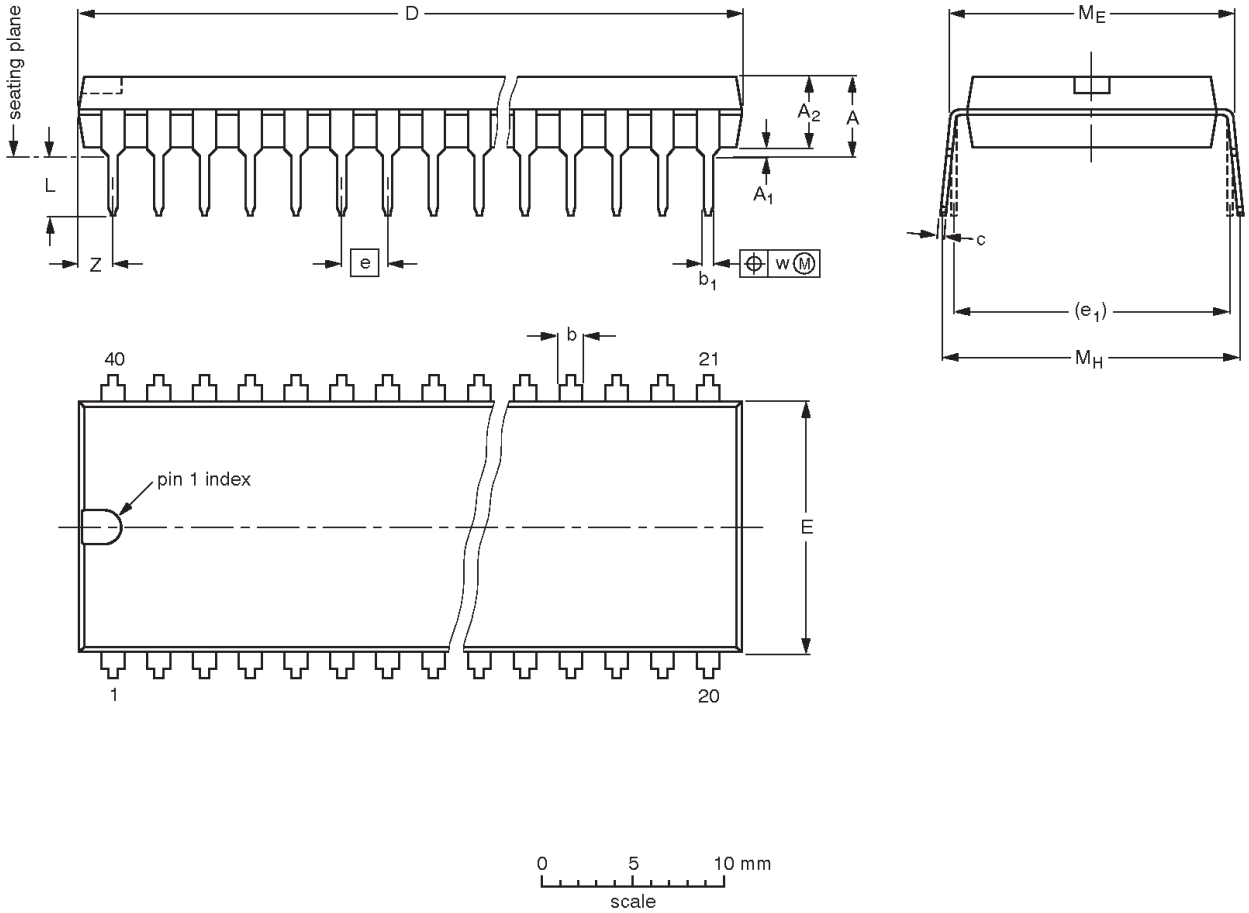
The security feature protects against software piracy and prevents the contents of the FLASH from being read. The Security Lock bits are located in FLASH. The P89C51X2/P89C52X2/P89C54X2/P89C58X2 has 3 programmable security lock bits that will provide different levels of protection for the on-chip code and data (see Table 9). Unlike the ROM and OTP versions, the security lock bits are independent. LB3 includes the security protection of LB1.

### Table 9.

| SECURITY LOCK BITS[1] | PROTECTION DESCRIPTION |
|---|---|
| Level | |
| LB1 | MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory. |
| LB2 | Program verification is disabled |
| LB3 | External execution is disabled. |

**NOTE:**
1. The security lock bits are independent.

## DIP40: plastic dual in-line package; 40 leads (600 mil) SOT129-1




**DIMENSIONS (inch dimensions are derived from the original mm dimensions)**

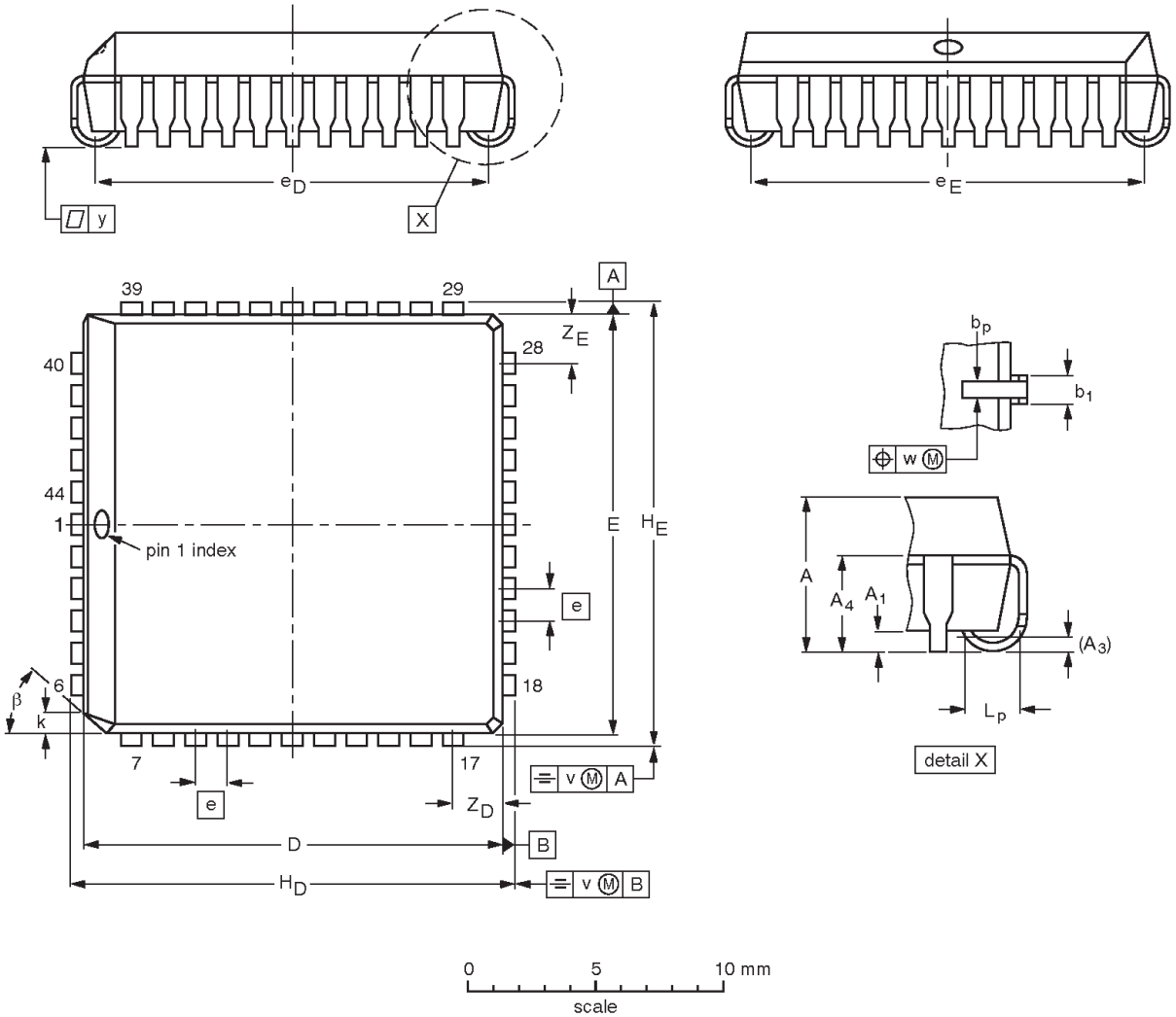
| UNIT | A max. | $A_1$ min. | $A_2$ max. | b | $b_1$ | c | D (1) | E (1) | e | $e_1$ | L | $M_E$ | $M_H$ | w | Z (1) max. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 4.7 | 0.51 | 4.0 | 1.70<br>1.14 | 0.53<br>0.38 | 0.36<br>0.23 | 52.50<br>51.50 | 14.1<br>13.7 | 2.54 | 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-015 | SC-511-40 | | | ~~95-01-14~~<br>99-12-27 |

**PLCC44: plastic leaded chip carrier; 44 leads** **SOT187-2**




**DIMENSIONS (mm dimensions are derived from the original inch dimensions)**

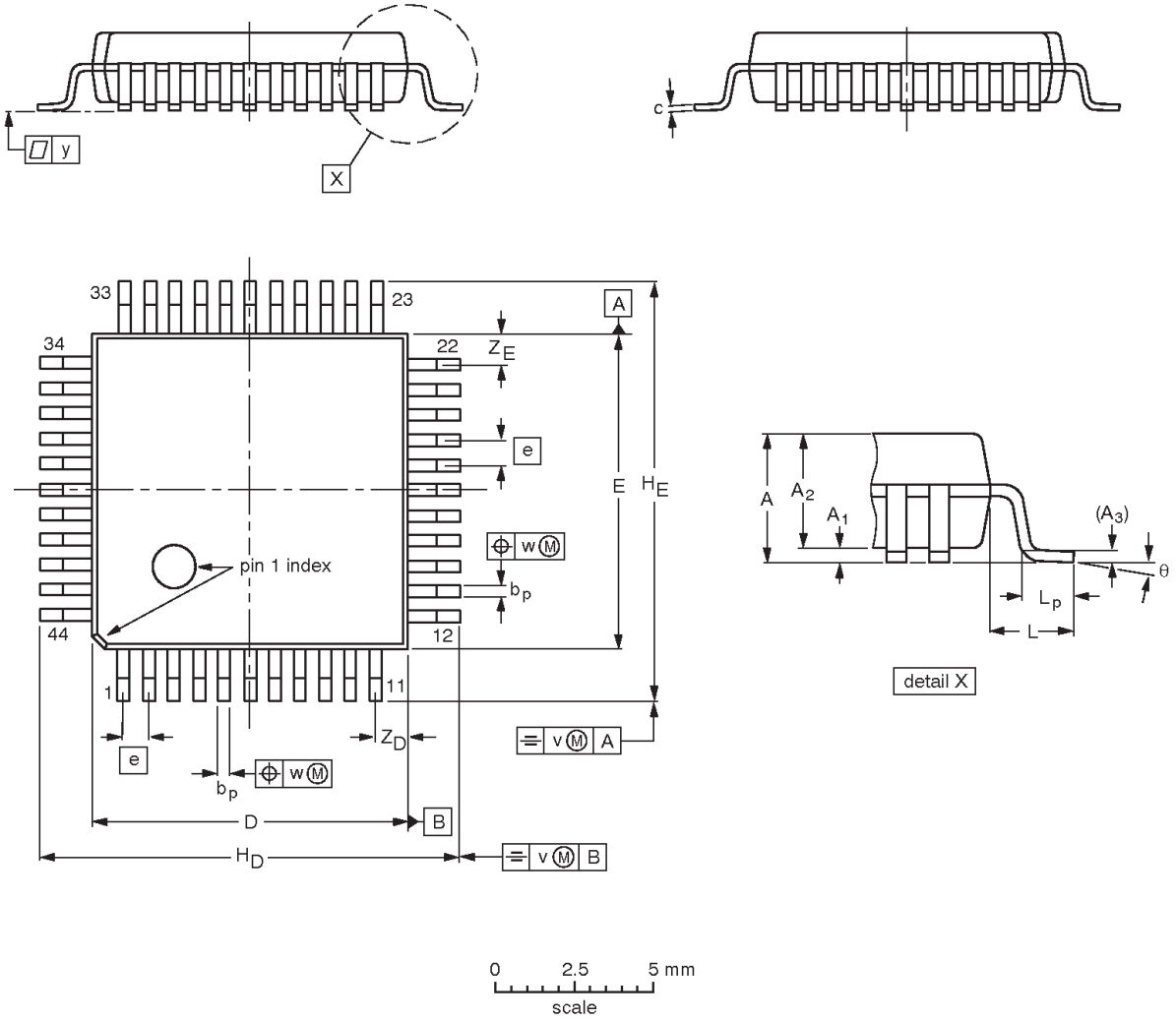
| UNIT | A | $A_1$ min. | $A_3$ | $A_4$ max. | $b_p$ | $b_1$ | D(1) | E(1) | e | $e_D$ | $e_E$ | $H_D$ | $H_E$ | k | $L_p$ | v | w | y | $Z_D$(1) max. | $Z_E$(1) max. | β |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 4.57<br>4.19 | 0.51 | 0.25 | 3.05 | 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 | 1.44<br>1.02 | 0.18 | 0.18 | 0.1 | 2.16 | 2.16 | 45° |
| inches | 0.180<br>0.165 | 0.02 | 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 | 0.63<br>0.59 | 0.63<br>0.59 | 0.695<br>0.685 | 0.695<br>0.685 | 0.048<br>0.042 | 0.057<br>0.040 | 0.007 | 0.007 | 0.004 | 0.085 | 0.085 | |

**Note**

1. Plastic or metal protrusions of 0.25 mm (0.01 inch) maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | JEITA | | | |
| SOT187-2 | 112E10 | MS-018 | EDR-7319 | | | ~~99-12-27~~<br>01-11-14 |

**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 | $L_p$ | 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 | 0.20 | 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 | 136E08 | MS-026 | | | | ~~99-12-17~~<br>00-01-19 |

## REVISION HISTORY

| Date | CPCN | Description |
|---|---|---|
| 2002 Jun 06 | 9397 750 09928 | Added device comparison table |
| 2002 Feb 28 | 9397 750 09537 | Initial release |

## Data sheet status

| Data sheet status [1] | Product status [2] | Definitions |
|---|---|---|
| Objective data | Development | This data sheet contains data from the objective specification for product development. Philips Semiconductors reserves the right to change the specification in any manner without notice. |
| Preliminary data | Qualification | This data sheet contains data from the preliminary specification. Supplementary data will be published at a later date. Philips Semiconductors reserves the right to change the specification without notice, in order to improve the design and supply the best possible product. |
| Product data | Production | This data sheet contains data from the product specification. Philips Semiconductors reserves the right to make changes at any time in order to improve the design, manufacturing and supply. Changes will be communicated according to the Customer Product/Process Change Notification (CPCN) procedure SNW-SQ-650A. |

[1] Please consult the most recently issued data sheet before initiating or completing a design.

[2] The product status of the device(s) described in this data sheet may have changed since this data sheet was published. The latest information is available on the Internet at URL http://www.semiconductors.philips.com.

## Definitions

**Short-form specification —** The data in a short-form specification is extracted from a full data sheet with the same type number and title. For detailed information see the relevant data sheet or data handbook.

**Limiting values definition —** Limiting values given are in accordance with the Absolute Maximum Rating System (IEC 60134). Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Characteristics sections of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability.

**Application information —** Applications that are described herein for any of these products are for illustrative purposes only. Philips Semiconductors make no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

## Disclaimers

**Life support —** These products are not designed for use in life support appliances, devices or systems where malfunction of these products can reasonably be expected to result in personal injury. Philips Semiconductors customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Philips Semiconductors for any damages resulting from such application.

**Right to make changes —** Philips Semiconductors reserves 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.

## Contact information

For additional information please visit
**http://www.semiconductors.philips.com**. Fax: **+31 40 27 24825**

For sales offices addresses send e-mail to:
**sales.addresses@www.semiconductors.philips.com**.

© Koninklijke Philips Electronics N.V. 2002
All rights reserved. Printed in U.S.A.

Date of release: 06-02

Document order number: 9397 750 09928

Let's make things better.

Philips Semiconductors

PHILIPS