
8-bit CMOS Microcontroller 0-60 MHz

1. Description

TEMIC TS80C52X2 is high performance CMOS ROM, OTP, EPROM and ROMless versions of the 80C51 CMOS single chip 8-bit microcontroller.

The TS80C52X2 retains all features of the TEMIC 80C51 with extended ROM/EPROM capacity (8 Kbytes), 256 bytes of internal RAM, a 6-source, 4-level interrupt system, an on-chip oscillator and three timer/counters.

In addition, the TS80C52X2 has a dual data pointer, a more versatile serial channel that facilitates multiprocessor communication (EUSART) and a X2 speed improvement mechanism.

The fully static design of the TS80C52X2 allows to reduce system power consumption by bringing the clock frequency down to any value, even DC, without loss of data.

The TS80C52X2 has 2 software-selectable modes of reduced activity for further reduction in power consumption. In the idle mode the CPU is frozen while the timers, the serial port and the interrupt system are still operating. In the power-down mode the RAM is saved and all other functions are inoperative.

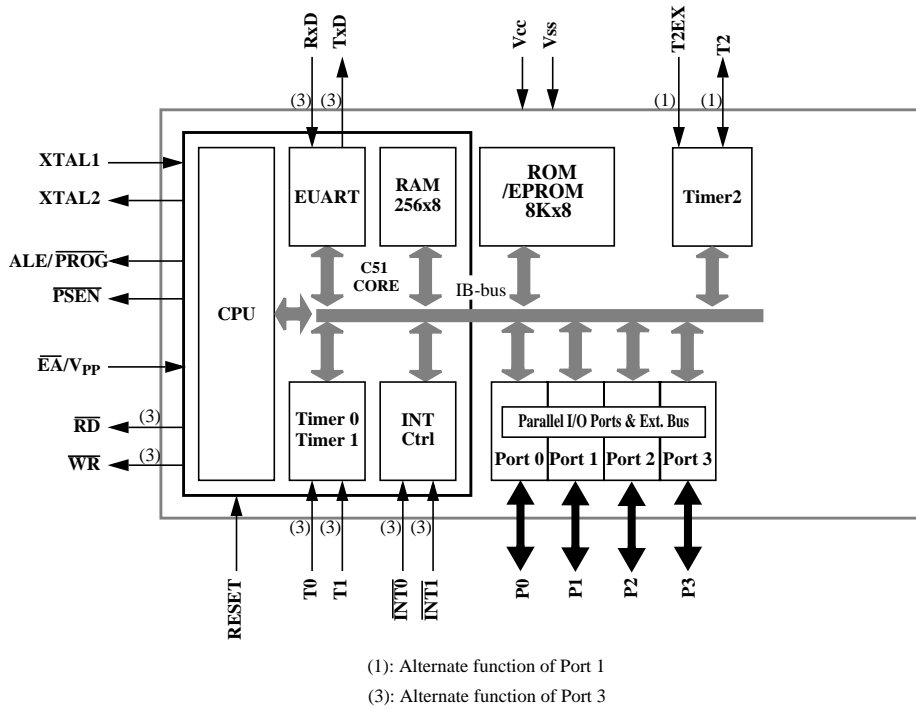
2. Features

- 80C52 Compatible
 - 8051 pin and instruction compatible
 - Four 8-bit I/O ports
 - Three 16-bit timer/counters
 - 256 bytes scratchpad RAM
- High-Speed Architecture
 - 40 MHz @ 5V, 30MHz @ 3V
 - X2 Speed Improvement capability (6 clocks/machine cycle)
 - 30 MHz @ 5V, 20 MHz @ 3V (Equivalent to 60 MHz @ 5V, 40 MHz @ 3V)
- Dual Data Pointer
- On-chip ROM/EPROM (8K-bytes)
- Programmable Clock Out and Up/Down Timer/Counter 2
- Asynchronous port reset
- Interrupt Structure with
 - 6 Interrupt sources,
 - 4 level priority interrupt system
- Full duplex Enhanced UART
 - Framing error detection
 - Automatic address recognition
- Low EMI (inhibit ALE)
- Power Control modes
 - Idle mode
 - Power-down mode
 - Power-off Flag
- Once mode (On-chip Emulation)
- Power supply: 4.5-5V, 2.7-5.5V
- Temperature ranges: Commercial (0 to 70°C) and Industrial (-40 to 85°C)
- Packages: PDIL40, PLCC44, VQFP44 1.4, PQFP F1 (13.9 footprint), CQPJ44 (window), CDIL40 (window)

Table 1. Memory size

| | ROM (bytes) | EPROM (bytes) | TOTAL RAM (bytes) |
|-----------|-------------|---------------|-------------------|
| TS80C32X2 | 0 | 0 | 256 |
| TS80C52X2 | 8k | 0 | 256 |
| TS87C52X2 | 0 | 8k | 256 |

3. Block Diagram



4. SFR Mapping

The Special Function Registers (SFRs) of the TS80C52X2 fall into the following categories:

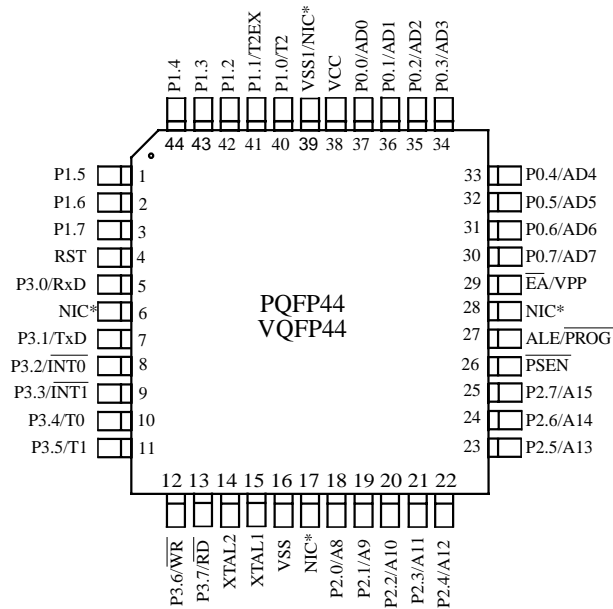
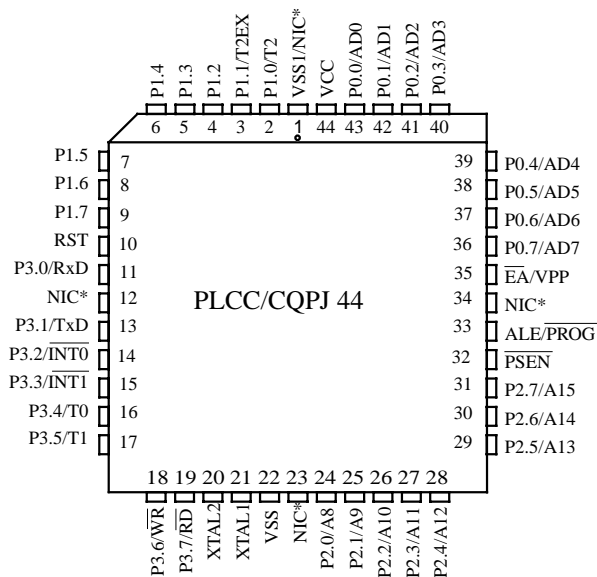
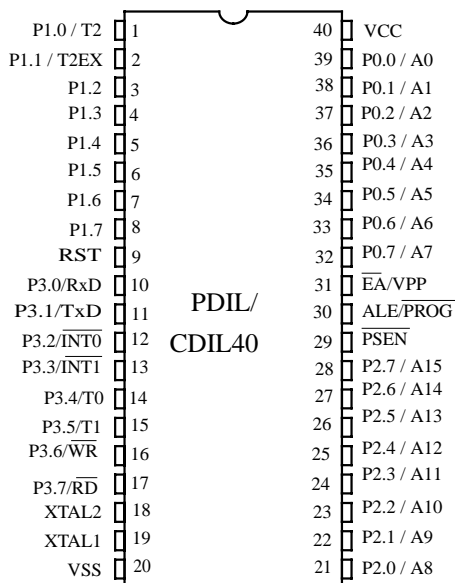
- C51 core registers: ACC, B, DPH, DPL, PSW, SP, AUXR1
- I/O port registers: P0, P1, P2, P3
- Timer registers: T2CON, T2MOD, TCON, TH0, TH1, TH2, TMOD, TL0, TL1, TL2, RCAP2L, RCAP2H
- Serial I/O port registers: SADDR, SADEN, SBUF, SCON
- Power and clock control registers: PCON
- Interrupt system registers: IE, IP, IPH
- Others: AUXR, CKCON

Table 2. All SFRs with their address and their reset value

| | Bit addressable | Non Bit addressable | | | | | | | | |
|-----|--------------------|---------------------|---------------------|---------------------|------------------|------------------|-------------------|--------------------|------------------|-----|
| | | 0/8 | 1/9 | 2/A | 3/B | 4/C | 5/D | 6/E | | 7/F |
| F8h | | | | | | | | | | FFh |
| F0h | B 0000 0000 | | | | | | | | | F7h |
| E8h | | | | | | | | | | EFh |
| E0h | ACC 0000 0000 | | | | | | | | | E7h |
| D8h | | | | | | | | | | DFh |
| D0h | PSW 0000 0000 | | | | | | | | | D7h |
| C8h | T2CON 0000 0000 | T2MOD XXXX XX00 | RCAP2L 0000 0000 | RCAP2H 0000 0000 | TL2 0000 0000 | TH2 0000 0000 | | | | CFh |
| C0h | | | | | | | | | | C7h |
| B8h | IP XX00 0000 | SADEN 0000 0000 | | | | | | | | BFh |
| B0h | P3 1111 1111 | | | | | | | | IPH XX00 0000 | B7h |
| A8h | IE 0X00 0000 | SADDR 0000 0000 | | | | | | | | AFh |
| A0h | P2 1111 1111 | | AUXR1 XXXX XXX0 | | | | | | | A7h |
| 98h | SCON 0000 0000 | SBUF XXXX XXXX | | | | | | | | 9Fh |
| 90h | P1 1111 1111 | | | | | | | | | 97h |
| 88h | TCON 0000 0000 | TMOD 0000 0000 | TL0 0000 0000 | TL1 0000 0000 | TH0 0000 0000 | TH1 0000 0000 | AUXR XXXXXXXX0 | CKCON XXXX XXX0 | | 8Fh |
| 80h | P0 1111 1111 | SP 0000 0111 | DPL 0000 0000 | DPH 0000 0000 | | | | PCON 00X1 0000 | | 87h |
| | 0/8 | 1/9 | 2/A | 3/B | 4/C | 5/D | 6/E | 7/F | | |

reserved

5. Pin Configuration



*NIC: No Internal Connection

Table 3. Pin Description for 40/44 pin packages

| MNEMONIC | PIN NUMBER | | | TYPE | NAME AND FUNCTION |
|------------------|------------|--------------|--------------|------|--|
| | DIL | LCC | VQFP 1.4 | | |
| V _{SS} | 20 | 22 | 16 | I | Ground: 0V reference |
| V _{SS1} | | 1 | 39 | I | Optional Ground: Contact the Sales Office for ground connection. |
| V _{CC} | 40 | 44 | 38 | I | Power Supply: This is the power supply voltage for normal, idle and power-down operation |
| P0.0-P0.7 | 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 pins must be polarized to V _{CC} or V _{SS} in order to prevent any parasitic current consumption. Port 0 is also the multiplexed low-order address and data bus during access to external program and data memory. In this application, it uses strong internal pull-up when emitting 1s. Port 0 also inputs the code bytes during EPROM programming. External pull-ups are required during program verification during which P0 outputs the code bytes. |
| 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. Port 1 also receives the low-order address byte during memory programming and verification. Alternate functions for Port 1 include: |
| P2.0-P2.7 | 1 | 2 | 40 | I/O | T2 (P1.0): Timer/Counter 2 external count input/Clockout |
| | 2 | 3 | 41 | I | T2EX (P1.1): Timer/Counter 2 Reload/Capture/Direction Control |
| | 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 pulled low will source current because of the internal pull-ups. 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 emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @Ri), port 2 emits the contents of the P2 SFR. Some Port 2 pins receive the high order address bits during EPROM programming and verification: P2.0 to P2.4 |
| 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 pulled low will source current because of the internal pull-ups. 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 | INT0 (P3.2): External interrupt 0 |
| | 13 | 15 | 9 | I | INT1 (P3.3): External interrupt 1 |
| | 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 | WR (P3.6): External data memory write strobe |
| | 17 | 19 | 13 | O | RD (P3.7): External data memory read strobe |
| Reset | 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} . |

Table 3. Pin Description for 40/44 pin packages

| MNEMONIC | PIN NUMBER | | | TYPE | NAME AND FUNCTION |
|---|------------|----|----|-------|---|
| ALE/ $\overline{\text{PROG}}$ | 30 | 33 | 27 | O (I) | 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 (1/3 in X2 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 EPROM programming. ALE can be disabled by setting SFR's AUXR.0 bit. With this bit set, ALE will be inactive during internal fetches. |
| $\overline{\text{PSEN}}$ | 29 | 32 | 26 | O | Program Store ENable: The read strobe to external program memory. When 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. |
| $\overline{\text{EA}}/\text{V}_{\text{PP}}$ | 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 and 3FFFH (RB) or 7FFFH (RC), or FFFFH (RD). If EA is held high, the device executes from internal program memory unless the program counter contains an address greater than 3FFFH (RB) or 7FFFH (RC) $\overline{\text{EA}}$ must be held low for ROMless devices. This pin also receives the 12.75V programming supply voltage (V_{PP}) during EPROM programming. If security level 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 |

6. TS80C52X2 Enhanced Features

In comparison to the original 80C52, the TS80C52X2 implements some new features, which are:

- The X2 option.
- The Dual Data Pointer.
- The 4 level interrupt priority system.
- The power-off flag.
- The ONCE mode.
- The ALE disabling.
- Some enhanced features are also located in the UART and the timer 2.

6.1 X2 Feature

The TS80C52X2 core needs only 6 clock periods per machine cycle. This feature called "X2" provides the following advantages:

- Divide frequency crystals by 2 (cheaper crystals) while keeping same CPU power.
- Save power consumption while keeping same CPU power (oscillator power saving).
- Save power consumption by dividing dynamically operating frequency by 2 in operating and idle modes.
- Increase CPU power by 2 while keeping same crystal frequency.

In order to keep the original C51 compatibility, a divider by 2 is inserted between the XTAL1 signal and the main clock input of the core (phase generator). This divider may be disabled by software.

6.1.1 Description

The clock for the whole circuit and peripheral is first divided by two before being used by the CPU core and peripherals. This allows any cyclic ratio to be accepted on XTAL1 input. In X2 mode, as this divider is bypassed, the signals on XTAL1 must have a cyclic ratio between 40 to 60%. Figure 1. shows the clock generation block diagram. X2 bit is validated on XTAL1÷2 rising edge to avoid glitches when switching from X2 to STD mode. Figure 2. shows the mode switching waveforms.

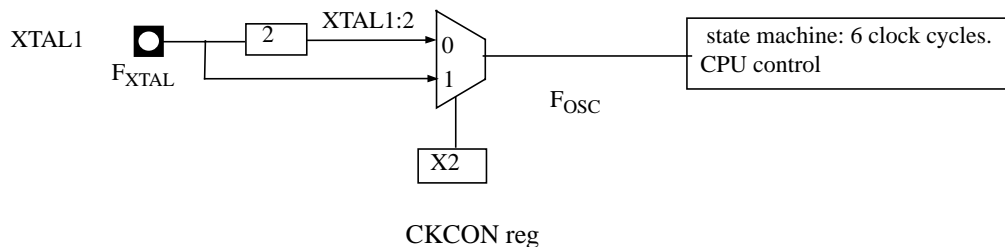


Figure 1. Clock Generation Diagram

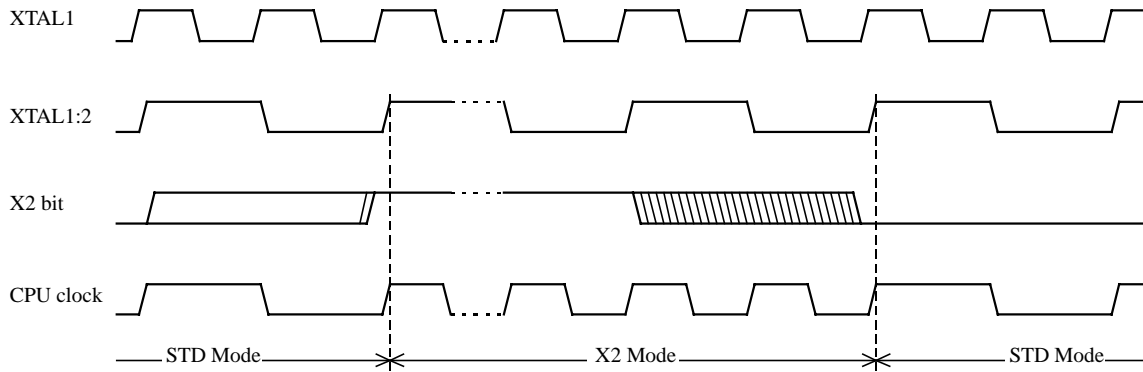


Figure 2. Mode Switching Waveforms

The X2 bit in the CKCON register (See Table 4.) allows to switch from 12 clock cycles per instruction to 6 clock cycles and vice versa. At reset, the standard speed is activated (STD mode). Setting this bit activates the X2 feature (X2 mode).

CAUTION

In order to prevent any incorrect operation while operating in X2 mode, user must be aware that all peripherals using clock frequency as time reference (UART, timers) will have their time reference divided by two. For example a free running timer generating an interrupt every 20 ms will then generate an interrupt every 10 ms. UART with 4800 baud rate will have 9600 baud rate.

Table 4. CKCON Register

CKCON - Clock Control Register (8Fh)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|--------------|---|---|---|---|---|-----------|
| - | - | - | - | - | - | - | X2 |
| Bit Number | Bit Mnemonic | Description | | | | | |
| 7 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 6 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 5 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 4 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 3 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 2 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 1 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 0 | X2 | CPU and peripheral clock bit Clear to select 12 clock periods per machine cycle (STD mode, $F_{OSC}=F_{XTAL}/2$). Set to select 6 clock periods per machine cycle (X2 mode, $F_{OSC}=F_{XTAL}$). | | | | | |

Reset Value = XXXX XXX0b

Not bit addressable

For further details on the X2 feature, please refer to ANM072 available on the web (<http://www.temic-semi.com>)

6.2 Dual Data Pointer Register Ddptr

The additional data pointer can be used to speed up code execution and reduce code size in a number of ways.

The dual DPTR structure is a way by which the chip will specify the address of an external data memory location. There are two 16-bit DPTR registers that address the external memory, and a single bit called DPS = AUXR1/bit0 (See Table 5.) that allows the program code to switch between them (Refer to Figure 3).

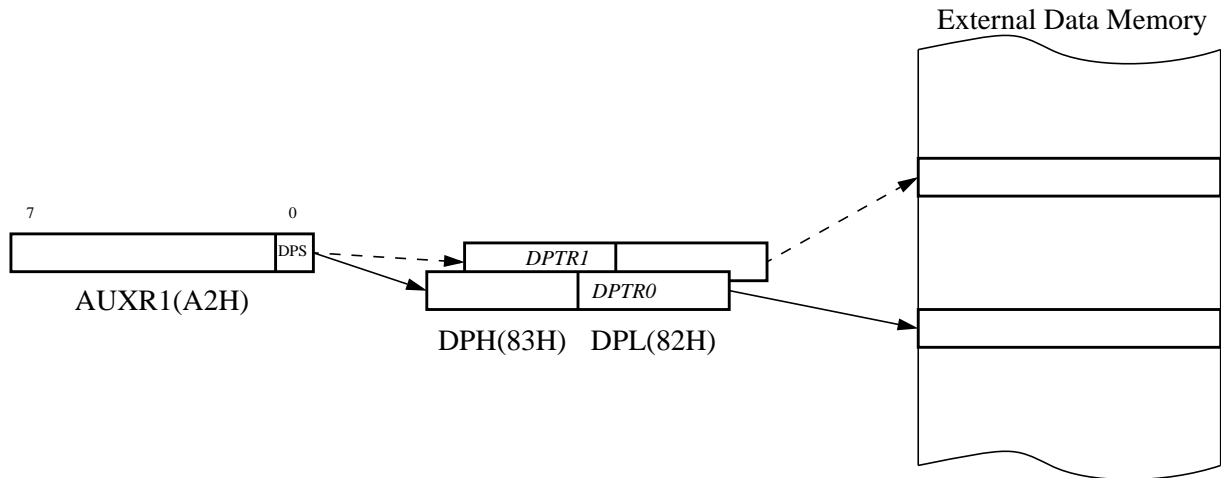


Figure 3. Use of Dual Pointer

Table 5. AUXR1: Auxiliary Register 1

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|-----|
| - | - | - | - | - | - | - | DPS |

| Bit Number | Bit Mnemonic | Description |
|------------|--------------|--|
| 7 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 6 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 5 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 4 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 3 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 2 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 1 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 0 | DPS | Data Pointer Selection Clear to select DPTR0. Set to select DPTR1. |

Reset Value = XXXX XXX0

Not bit addressable

Application

Software can take advantage of the additional data pointers to both increase speed and reduce code size, for example, block operations (copy, compare, search ...) are well served by using one data pointer as a 'source' pointer and the other one as a "destination" pointer.

ASSEMBLY LANGUAGE

```

; Block move using dual data pointers
; Destroys DPTR0, DPTR1, A and PSW
; note: DPS exits opposite of entry state
; unless an extra INC AUXR1 is added
;
00A2      AUXR1 EQU 0A2H
;
0000 909000 MOV DPTR,#SOURCE      ; address of SOURCE
0003 05A2  INC  AUXR1              ; switch data pointers
0005 90A000 MOV DPTR,#DEST        ; address of DEST
0008      LOOP:
0008 05A2  INC  AUXR1              ; switch data pointers
000A E0    MOVX A,@DPTR            ; get a byte from SOURCE
000B A3    INC  DPTR              ; increment SOURCE address
000C 05A2  INC  AUXR1              ; switch data pointers
000E F0    MOVX @DPTR,A           ; write the byte to DEST
000F A3    INC  DPTR              ; increment DEST address
0010 70F6  JNZ  LOOP              ; check for 0 terminator
0012 05A2  INC  AUXR1              ; (optional) restore DPS

```

INC is a short (2 bytes) and fast (12 clocks) way to manipulate the DPS bit in the AUXR1 SFR. However, note that the INC instruction does not directly force the DPS bit to a particular state, but simply toggles it. In simple routines, such as the block move example, only the fact that DPS is toggled in the proper sequence matters, not its actual value. In other words, the block move routine works the same whether DPS is '0' or '1' on entry. Observe that without the last instruction (INC AUXR1), the routine will exit with DPS in the opposite state.

6.3 Timer 2

The timer 2 in the TS80C52X2 is compatible with the timer 2 in the 80C52.

It is a 16-bit timer/counter: the count is maintained by two eight-bit timer registers, TH2 and TL2, connected in cascade. It is controlled by T2CON register (See Table 6) and T2MOD register (See Table 7). Timer 2 operation is similar to Timer 0 and Timer 1. $C/\overline{T2}$ selects $F_{OSC}/12$ (timer operation) or external pin T2 (counter operation) as the timer clock input. Setting TR2 allows TL2 to be incremented by the selected input.

Timer 2 has 3 operating modes: capture, autoreload and Baud Rate Generator. These modes are selected by the combination of RCLK, TCLK and $CP/\overline{RL2}$ (T2CON), as described in the TEMIC 8-bit Microcontroller Hardware description.

Refer to the TEMIC 8-bit Microcontroller Hardware description for the description of Capture and Baud Rate Generator Modes.

In TS80C52X2 Timer 2 includes the following enhancements:

- Auto-reload mode with up or down counter
- Programmable clock-output

6.3.1 Auto-Reload Mode

The auto-reload mode configures timer 2 as a 16-bit timer or event counter with automatic reload. If DCEN bit in T2MOD is cleared, timer 2 behaves as in 80C52 (refer to the TEMIC 8-bit Microcontroller Hardware description). If DCEN bit is set, timer 2 acts as an Up/down timer/counter as shown in Figure 4. In this mode the T2EX pin controls the direction of count.

When T2EX is high, timer 2 counts up. Timer overflow occurs at FFFFh which sets the TF2 flag and generates an interrupt request. The overflow also causes the 16-bit value in RCAP2H and RCAP2L registers to be loaded into the timer registers TH2 and TL2.

When T2EX is low, timer 2 counts down. Timer underflow occurs when the count in the timer registers TH2 and TL2 equals the value stored in RCAP2H and RCAP2L registers. The underflow sets TF2 flag and reloads FFFFh into the timer registers.

The EXF2 bit toggles when timer 2 overflows or underflows according to the the direction of the count. EXF2 does not generate any interrupt. This bit can be used to provide 17-bit resolution.

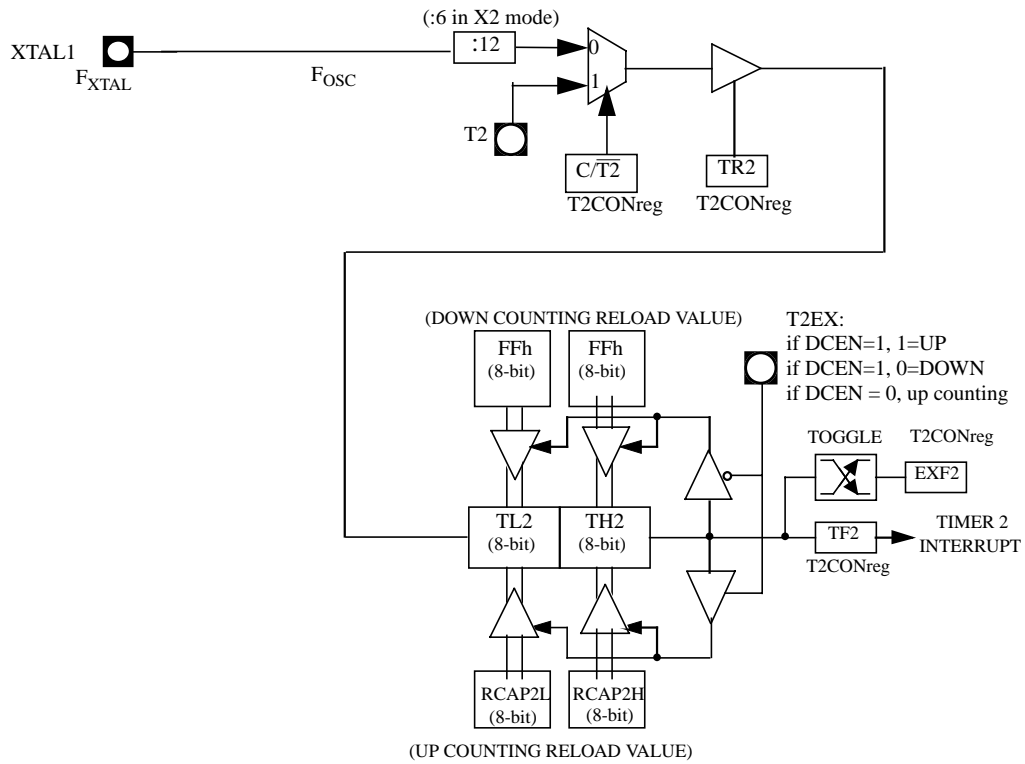


Figure 4. Auto-Reload Mode Up/Down Counter (DCEN = 1)

6.3.2 Programmable Clock-Output

In the clock-out mode, timer 2 operates as a 50%-duty-cycle, programmable clock generator (See Figure 5) . The input clock increments TL2 at frequency $F_{OSC}/2$. The timer repeatedly counts to overflow from a loaded value. At overflow, the contents of RCAP2H and RCAP2L registers are loaded into TH2 and TL2. In this mode, timer 2 overflows do not generate interrupts. The formula gives the clock-out frequency as a function of the system oscillator frequency and the value in the RCAP2H and RCAP2L registers :

$$Clock - OutFrequency = \frac{F_{osc}}{4 \times (65536 - RCAP2H/RCAP2L)}$$

For a 16 MHz system clock, timer 2 has a programmable frequency range of 61 Hz ($F_{OSC}/2^{16}$) to 4 MHz ($F_{OSC}/4$). The generated clock signal is brought out to T2 pin (P1.0).

Timer 2 is programmed for the clock-out mode as follows:

- Set T2OE bit in T2MOD register.
- Clear $C/\overline{T2}$ bit in T2CON register.
- Determine the 16-bit reload value from the formula and enter it in RCAP2H/RCAP2L registers.
- Enter a 16-bit initial value in timer registers TH2/TL2. It can be the same as the reload value or a different one depending on the application.

- To start the timer, set TR2 run control bit in T2CON register.

It is possible to use timer 2 as a baud rate generator and a clock generator simultaneously. For this configuration, the baud rates and clock frequencies are not independent since both functions use the values in the RCAP2H and RCAP2L registers.

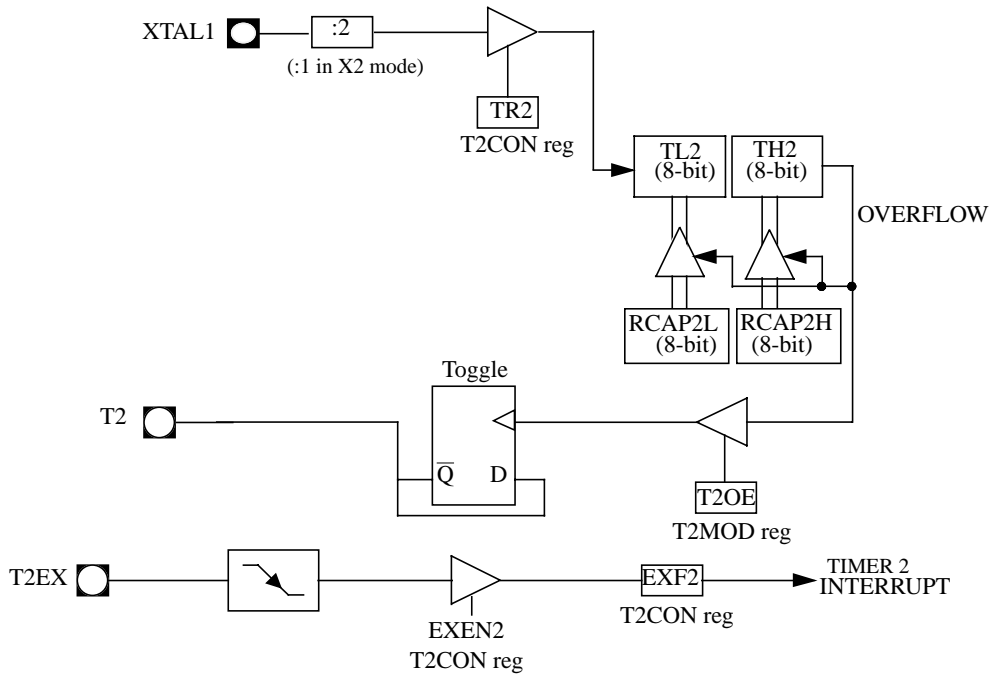


Figure 5. Clock-Out Mode $C/\overline{T2} = 0$

Table 6. T2CON Register

T2CON - Timer 2 Control Register (C8h)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|--------------|--|------|-------|-----|-------|---------|
| TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/T2# | CP/RL2# |
| Bit Number | Bit Mnemonic | Description | | | | | |
| 7 | TF2 | Timer 2 overflow Flag Must be cleared by software. Set by hardware on timer 2 overflow, if RCLK = 0 and TCLK = 0. | | | | | |
| 6 | EXF2 | Timer 2 External Flag Set when a capture or a reload is caused by a negative transition on T2EX pin if EXEN2=1. When set, causes the CPU to vector to timer 2 interrupt routine when timer 2 interrupt is enabled. Must be cleared by software. EXF2 doesn't cause an interrupt in Up/down counter mode (DCEN = 1) | | | | | |
| 5 | RCLK | Receive Clock bit Clear to use timer 1 overflow as receive clock for serial port in mode 1 or 3. Set to use timer 2 overflow as receive clock for serial port in mode 1 or 3. | | | | | |
| 4 | TCLK | Transmit Clock bit Clear to use timer 1 overflow as transmit clock for serial port in mode 1 or 3. Set to use timer 2 overflow as transmit clock for serial port in mode 1 or 3. | | | | | |
| 3 | EXEN2 | Timer 2 External Enable bit Clear to ignore events on T2EX pin for timer 2 operation. Set to cause a capture or reload when a negative transition on T2EX pin is detected, if timer 2 is not used to clock the serial port. | | | | | |
| 2 | TR2 | Timer 2 Run control bit Clear to turn off timer 2. Set to turn on timer 2. | | | | | |
| 1 | C/T2# | Timer/Counter 2 select bit Clear for timer operation (input from internal clock system: F _{OSC}). Set for counter operation (input from T2 input pin, falling edge trigger). Must be 0 for clock out mode. | | | | | |
| 0 | CP/RL2# | Timer 2 Capture/Reload bit If RCLK=1 or TCLK=1, CP/RL2# is ignored and timer is forced to auto-reload on timer 2 overflow. Clear to auto-reload on timer 2 overflows or negative transitions on T2EX pin if EXEN2=1. Set to capture on negative transitions on T2EX pin if EXEN2=1. | | | | | |

Reset Value = 0000 0000b

Bit addressable

Table 7. T2MOD Register

T2MOD - Timer 2 Mode Control Register (C9h)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|--------------|---|---|---|---|-------------|-------------|
| - | - | - | - | - | - | T2OE | DCEN |
| Bit Number | Bit Mnemonic | Description | | | | | |
| 7 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 6 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 5 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 4 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 3 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 2 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 1 | T2OE | Timer 2 Output Enable bit Clear to program P1.0/T2 as clock input or I/O port. Set to program P1.0/T2 as clock output. | | | | | |
| 0 | DCEN | Down Counter Enable bit Clear to disable timer 2 as up/down counter. Set to enable timer 2 as up/down counter. | | | | | |

Reset Value = XXXX XX00b

Not bit addressable

6.4 TS80C52X2 Serial I/O Port

The serial I/O port in the TS80C52X2 is compatible with the serial I/O port in the 80C52.

It provides both synchronous and asynchronous communication modes. It operates as an Universal Asynchronous Receiver and Transmitter (UART) in three full-duplex modes (Modes 1, 2 and 3). Asynchronous transmission and reception can occur simultaneously and at different baud rates

Serial I/O port includes the following enhancements:

- Framing error detection
- Automatic address recognition

6.4.1 Framing Error Detection

Framing bit error detection is provided for the three asynchronous modes (modes 1, 2 and 3). To enable the framing bit error detection feature, set SMOD0 bit in PCON register (See Figure 6).

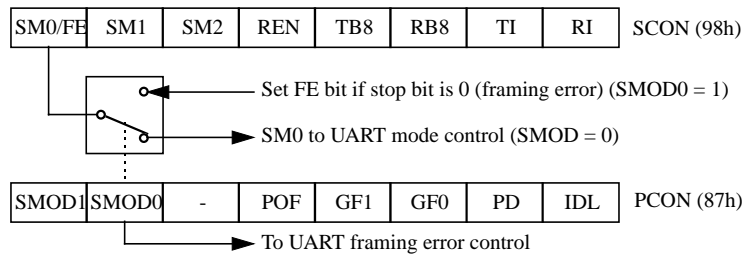


Figure 6. Framing Error Block Diagram

When this feature is enabled, the receiver checks each incoming data frame for a valid stop bit. An invalid stop bit may result from noise on the serial lines or from simultaneous transmission by two CPUs. If a valid stop bit is not found, the Framing Error bit (FE) in SCON register (See Table 8.) bit is set.

Software may examine FE bit after each reception to check for data errors. Once set, only software or a reset can clear FE bit. Subsequently received frames with valid stop bits cannot clear FE bit. When FE feature is enabled, RI rises on stop bit instead of the last data bit (See Figure 7. and Figure 8.).

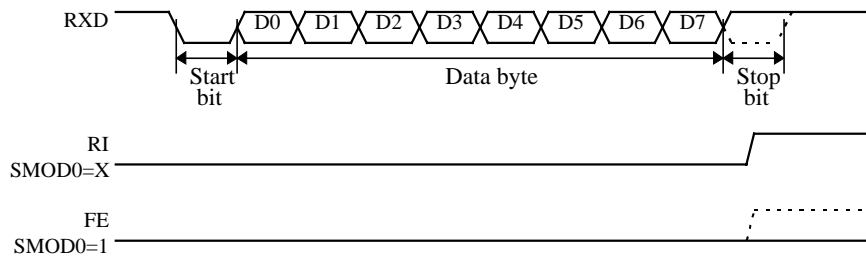


Figure 7. UART Timings in Mode 1

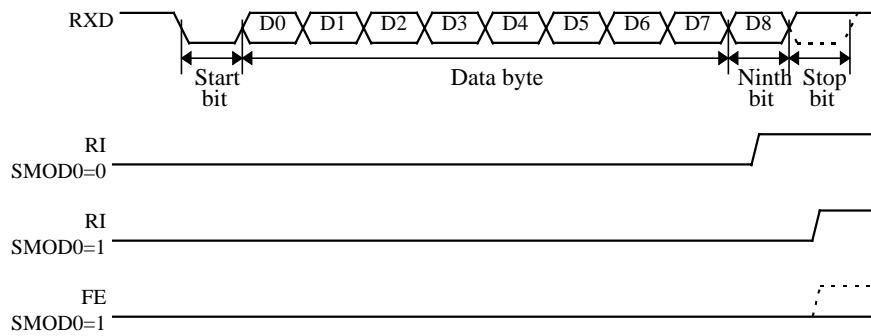


Figure 8. UART Timings in Modes 2 and 3

6.4.2 Automatic Address Recognition

The automatic address recognition feature is enabled when the multiprocessor communication feature is enabled (SM2 bit in SCON register is set).

Implemented in hardware, automatic address recognition enhances the multiprocessor communication feature by allowing the serial port to examine the address of each incoming command frame. Only when the serial port recognizes its own address, the receiver sets RI bit in SCON register to generate an interrupt. This ensures that the CPU is not interrupted by command frames addressed to other devices.

If desired, you may enable the automatic address recognition feature in mode 1. In this configuration, the stop bit takes the place of the ninth data bit. Bit RI is set only when the received command frame address matches the device's address and is terminated by a valid stop bit.

To support automatic address recognition, a device is identified by a given address and a broadcast address.

NOTE: The multiprocessor communication and automatic address recognition features cannot be enabled in mode 0 (i.e. setting SM2 bit in SCON register in mode 0 has no effect).

6.4.3 Given Address

Each device has an individual address that is specified in SADDR register; the SADEN register is a mask byte that contains don't-care bits (defined by zeros) to form the device's given address. The don't-care bits provide the flexibility to address one or more slaves at a time. The following example illustrates how a given address is formed. To address a device by its individual address, the SADEN mask byte must be 1111 1111b.

For example:

| | |
|-------|-------------------|
| SADDR | 0101 0110b |
| SADEN | <u>1111 1100b</u> |
| Given | 0101 01XXb |

The following is an example of how to use given addresses to address different slaves:

| | | |
|----------|-------|-------------------|
| Slave A: | SADDR | 1111 0001b |
| | SADEN | <u>1111 1010b</u> |
| | Given | 1111 0X0Xb |

| | | |
|----------|-------|-------------------|
| Slave B: | SADDR | 1111 0011b |
| | SADEN | <u>1111 1001b</u> |
| | Given | 1111 0XX1b |

| | | |
|----------|-------|-------------------|
| Slave C: | SADDR | 1111 0010b |
| | SADEN | <u>1111 1101b</u> |
| | Given | 1111 00X1b |

The SADEN byte is selected so that each slave may be addressed separately.

For slave A, bit 0 (the LSB) is a don't-care bit; for slaves B and C, bit 0 is a 1. To communicate with slave A only, the master must send an address where bit 0 is clear (e.g. 1111 0000b).

For slave A, bit 1 is a 1; for slaves B and C, bit 1 is a don't care bit. To communicate with slaves B and C, but not slave A, the master must send an address with bits 0 and 1 both set (e.g. 1111 0011b).

To communicate with slaves A, B and C, the master must send an address with bit 0 set, bit 1 clear, and bit 2 clear (e.g. 1111 0001b).

6.4.4 Broadcast Address

A broadcast address is formed from the logical OR of the SADDR and SADEN registers with zeros defined as don't-care bits, e.g.:

| | | |
|---------------------------|-------|------------|
| | SADDR | 0101 0110b |
| | SADEN | 1111 1100b |
| Broadcast =SADDR OR SADEN | | 1111 111Xb |

The use of don't-care bits provides flexibility in defining the broadcast address, however in most applications, a broadcast address is FFh. The following is an example of using broadcast addresses:

| | | |
|----------|-----------|-------------------|
| Slave A: | SADDR | 1111 0001b |
| | SADEN | <u>1111 1010b</u> |
| | Broadcast | 1111 1X11b, |

| | | |
|----------|-----------|-------------------|
| Slave B: | SADDR | 1111 0011b |
| | SADEN | <u>1111 1001b</u> |
| | Broadcast | 1111 1X11B, |

| | | |
|----------|-----------|-------------------|
| Slave C: | SADDR= | 1111 0010b |
| | SADEN | <u>1111 1101b</u> |
| | Broadcast | 1111 1111b |

For slaves A and B, bit 2 is a don't care bit; for slave C, bit 2 is set. To communicate with all of the slaves, the master must send an address FFh. To communicate with slaves A and B, but not slave C, the master can send and address FBh.

6.4.5 Reset Addresses

On reset, the SADDR and SADEN registers are initialized to 00h, i.e. the given and broadcast addresses are XXXX XXXXb (all don't-care bits). This ensures that the serial port will reply to any address, and so, that it is backwards compatible with the 80C51 microcontrollers that do not support automatic address recognition.

SADEN - Slave Address Mask Register (B9h)

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

Reset Value = 0000 0000b

Not bit addressable

SADDR - Slave Address Register (A9h)

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

Reset Value = 0000 0000b

Not bit addressable

Table 8. SCON Register

SCON - Serial Control Register (98h)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------|--------------|--|----------------|--|-----|----|----|-----|-----|------|-------------|-----------|---|---|---|----------------|-------------------------------|---|---|---|------------|----------|---|---|---|------------|--|---|---|---|------------|----------|
| FE/SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI | | | | | | | | | | | | | | | | | | | | | | | | | |
| Bit Number | Bit Mnemonic | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 7 | FE | Framing Error bit (SMOD0=1) Clear to reset the error state, not cleared by a valid stop bit. Set by hardware when an invalid stop bit is detected. SMOD0 must be set to enable access to the FE bit | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SM0 | Serial port Mode bit 0 Refer to SM1 for serial port mode selection. SMOD0 must be cleared to enable access to the SM0 bit | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6 | SM1 | Serial port Mode bit 1 <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">SM1</th> <th style="text-align: center;">SM0</th> <th style="text-align: center;">Mode</th> <th style="text-align: center;">Description</th> <th style="text-align: center;">Baud Rate</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>Shift Register</td> <td>$F_{XTAL}/12$ (/6 in X2 mode)</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td>8-bit UART</td> <td>Variable</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">2</td> <td>9-bit UART</td> <td>$F_{XTAL}/64$ or $F_{XTAL}/32$ (/32, /16 in X2 mode)</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">3</td> <td>9-bit UART</td> <td>Variable</td> </tr> </tbody> </table> | | | | | | SM1 | SM0 | Mode | Description | Baud Rate | 0 | 0 | 0 | Shift Register | $F_{XTAL}/12$ (/6 in X2 mode) | 0 | 1 | 1 | 8-bit UART | Variable | 1 | 0 | 2 | 9-bit UART | $F_{XTAL}/64$ or $F_{XTAL}/32$ (/32, /16 in X2 mode) | 1 | 1 | 3 | 9-bit UART | Variable |
| SM1 | SM0 | Mode | Description | Baud Rate | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | Shift Register | $F_{XTAL}/12$ (/6 in X2 mode) | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | 8-bit UART | Variable | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 2 | 9-bit UART | $F_{XTAL}/64$ or $F_{XTAL}/32$ (/32, /16 in X2 mode) | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 3 | 9-bit UART | Variable | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5 | SM2 | Serial port Mode 2 bit / Multiprocessor Communication Enable bit Clear to disable multiprocessor communication feature. Set to enable multiprocessor communication feature in mode 2 and 3, and eventually mode 1. This bit should be cleared in mode 0. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 | REN | Reception Enable bit Clear to disable serial reception. Set to enable serial reception. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 | TB8 | Transmitter Bit 8 / Ninth bit to transmit in modes 2 and 3. Clear to transmit a logic 0 in the 9th bit. Set to transmit a logic 1 in the 9th bit. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | RB8 | Receiver Bit 8 / Ninth bit received in modes 2 and 3 Cleared by hardware if 9th bit received is a logic 0. Set by hardware if 9th bit received is a logic 1. In mode 1, if SM2 = 0, RB8 is the received stop bit. In mode 0 RB8 is not used. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | TI | Transmit Interrupt flag Clear to acknowledge interrupt. 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. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | RI | Receive Interrupt flag Clear to acknowledge interrupt. Set by hardware at the end of the 8th bit time in mode 0, see Figure 7. and Figure 8. in the other modes. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Reset Value = 0000 0000b

Bit addressable

Table 9. PCON Register

PCON - Power Control Register (87h)

| | | | | | | | |
|--------------|--------------|----------|------------|------------|------------|-----------|------------|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SMOD1 | SMOD0 | - | POF | GF1 | GF0 | PD | IDL |

| Bit Number | Bit Mnemonic | Description |
|------------|--------------|--|
| 7 | SMOD1 | Serial port Mode bit 1 Set to select double baud rate in mode 1, 2 or 3. |
| 6 | SMOD0 | Serial port Mode bit 0 Clear to select SM0 bit in SCON register. Set to select FE bit in SCON register. |
| 5 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 4 | POF | Power-Off Flag Clear to recognize next reset type. Set by hardware when VCC rises from 0 to its nominal voltage. Can also be set by software. |
| 3 | GF1 | General purpose Flag Cleared by user for general purpose usage. Set by user for general purpose usage. |
| 2 | GF0 | General purpose Flag Cleared by user for general purpose usage. Set by user for general purpose usage. |
| 1 | PD | Power-Down mode bit Cleared by hardware when reset occurs. Set to enter power-down mode. |
| 0 | IDL | Idle mode bit Clear by hardware when interrupt or reset occurs. Set to enter idle mode. |

Reset Value = 00X1 0000b

Not bit addressable

Power-off flag reset value will be 1 only after a power on (cold reset). A warm reset doesn't affect the value of this bit.

6.5 Interrupt System

The TS80C52X2 has a total of 6 interrupt vectors: two external interrupts ($\overline{INT0}$ and $\overline{INT1}$), three timer interrupts (timers 0, 1 and 2) and the serial port interrupt. These interrupts are shown in Figure 9.

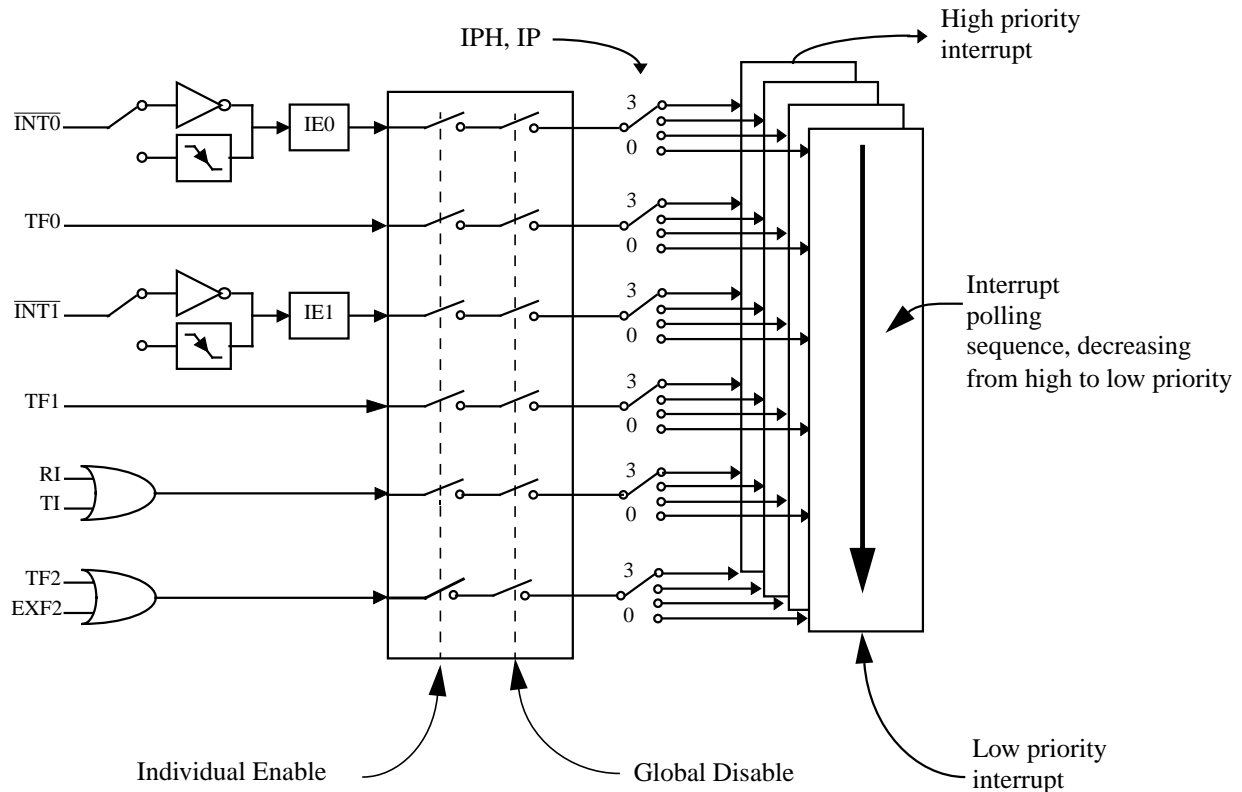


Figure 9. Interrupt Control System

Each of the interrupt sources can be individually enabled or disabled by setting or clearing a bit in the Interrupt Enable register (See Table 11.). This register also contains a global disable bit, which must be cleared to disable all interrupts at once.

Each interrupt source can also be individually programmed to one out of four priority levels by setting or clearing a bit in the Interrupt Priority register (See Table 12.) and in the Interrupt Priority High register (See Table 13.). shows the bit values and priority levels associated with each combination.

Table 10. Priority Level Bit Values

| IPH.x | IP.x | Interrupt Level Priority |
|-------|------|--------------------------|
| 0 | 0 | 0 (Lowest) |
| 0 | 1 | 1 |
| 1 | 0 | 2 |
| 1 | 1 | 3 (Highest) |

A low-priority interrupt can be interrupted by a high priority interrupt, but not by another low-priority interrupt. A high-priority interrupt can't be interrupted by any other interrupt source.

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

Table 11. IE Register

IE - Interrupt Enable Register (A8h)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|--------------|---|----|-----|-----|-----|-----|
| EA | - | ET2 | ES | ET1 | EX1 | ET0 | EX0 |
| Bit Number | Bit Mnemonic | Description | | | | | |
| 7 | EA | Enable All interrupt bit Clear to disable all interrupts. Set to enable all interrupts. If EA=1, each interrupt source is individually enabled or disabled by setting or clearing its own interrupt enable bit. | | | | | |
| 6 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 5 | ET2 | Timer 2 overflow interrupt Enable bit Clear to disable timer 2 overflow interrupt. Set to enable timer 2 overflow interrupt. | | | | | |
| 4 | ES | Serial port Enable bit Clear to disable serial port interrupt. Set to enable serial port interrupt. | | | | | |
| 3 | ET1 | Timer 1 overflow interrupt Enable bit Clear to disable timer 1 overflow interrupt. Set to enable timer 1 overflow interrupt. | | | | | |
| 2 | EX1 | External interrupt 1 Enable bit Clear to disable external interrupt 1. Set to enable external interrupt 1. | | | | | |
| 1 | ET0 | Timer 0 overflow interrupt Enable bit Clear to disable timer 0 overflow interrupt. Set to enable timer 0 overflow interrupt. | | | | | |
| 0 | EX0 | External interrupt 0 Enable bit Clear to disable external interrupt 0. Set to enable external interrupt 0. | | | | | |

Reset Value = 0X00 0000b

Bit addressable

Table 12. IP Register

IP - Interrupt Priority Register (B8h)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|--------------|--|-----------|------------|------------|------------|------------|
| - | - | PT2 | PS | PT1 | PX1 | PT0 | PX0 |
| Bit Number | Bit Mnemonic | Description | | | | | |
| 7 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 6 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 5 | PT2 | Timer 2 overflow interrupt Priority bit Refer to PT2H for priority level. | | | | | |
| 4 | PS | Serial port Priority bit Refer to PSH for priority level. | | | | | |
| 3 | PT1 | Timer 1 overflow interrupt Priority bit Refer to PT1H for priority level. | | | | | |
| 2 | PX1 | External interrupt 1 Priority bit Refer to PX1H for priority level. | | | | | |
| 1 | PT0 | Timer 0 overflow interrupt Priority bit Refer to PT0H for priority level. | | | | | |
| 0 | PX0 | External interrupt 0 Priority bit Refer to PX0H for priority level. | | | | | |

Reset Value = XX00 0000b

Bit addressable

Table 13. IPH Register

IPH - Interrupt Priority High Register (B7h)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | | | | | | | | | | | | |
|-------------|--------------|---|-----|------|------|------|------|-------------|------------|-----------------------|---|---|--------|---|---|--|---|---|--|---|---|---------|
| - | - | PT2H | PSH | PT1H | PX1H | PT0H | PX0H | | | | | | | | | | | | | | | |
| Bit Number | Bit Mnemonic | Description | | | | | | | | | | | | | | | | | | | | |
| 7 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | | | | | | | | | | | | | | | | |
| 6 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | | | | | | | | | | | | | | | | |
| 5 | PT2H | Timer 2 overflow interrupt Priority High bit <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;"><u>PT2H</u></td> <td style="text-align: center;"><u>PT2</u></td> <td style="text-align: center;"><u>Priority Level</u></td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">Lowest</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">Highest</td> </tr> </table> | | | | | | <u>PT2H</u> | <u>PT2</u> | <u>Priority Level</u> | 0 | 0 | Lowest | 0 | 1 | | 1 | 0 | | 1 | 1 | Highest |
| <u>PT2H</u> | <u>PT2</u> | <u>Priority Level</u> | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | Lowest | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | Highest | | | | | | | | | | | | | | | | | | | | |
| 4 | PSH | Serial port Priority High bit <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;"><u>PSH</u></td> <td style="text-align: center;"><u>PS</u></td> <td style="text-align: center;"><u>Priority Level</u></td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">Lowest</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">Highest</td> </tr> </table> | | | | | | <u>PSH</u> | <u>PS</u> | <u>Priority Level</u> | 0 | 0 | Lowest | 0 | 1 | | 1 | 0 | | 1 | 1 | Highest |
| <u>PSH</u> | <u>PS</u> | <u>Priority Level</u> | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | Lowest | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | Highest | | | | | | | | | | | | | | | | | | | | |
| 3 | PT1H | Timer 1 overflow interrupt Priority High bit <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;"><u>PT1H</u></td> <td style="text-align: center;"><u>PT1</u></td> <td style="text-align: center;"><u>Priority Level</u></td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">Lowest</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">Highest</td> </tr> </table> | | | | | | <u>PT1H</u> | <u>PT1</u> | <u>Priority Level</u> | 0 | 0 | Lowest | 0 | 1 | | 1 | 0 | | 1 | 1 | Highest |
| <u>PT1H</u> | <u>PT1</u> | <u>Priority Level</u> | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | Lowest | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | Highest | | | | | | | | | | | | | | | | | | | | |
| 2 | PX1H | External interrupt 1 Priority High bit <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;"><u>PX1H</u></td> <td style="text-align: center;"><u>PX1</u></td> <td style="text-align: center;"><u>Priority Level</u></td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">Lowest</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">Highest</td> </tr> </table> | | | | | | <u>PX1H</u> | <u>PX1</u> | <u>Priority Level</u> | 0 | 0 | Lowest | 0 | 1 | | 1 | 0 | | 1 | 1 | Highest |
| <u>PX1H</u> | <u>PX1</u> | <u>Priority Level</u> | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | Lowest | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | Highest | | | | | | | | | | | | | | | | | | | | |
| 1 | PT0H | Timer 0 overflow interrupt Priority High bit <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;"><u>PT0H</u></td> <td style="text-align: center;"><u>PT0</u></td> <td style="text-align: center;"><u>Priority Level</u></td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">Lowest</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">Highest</td> </tr> </table> | | | | | | <u>PT0H</u> | <u>PT0</u> | <u>Priority Level</u> | 0 | 0 | Lowest | 0 | 1 | | 1 | 0 | | 1 | 1 | Highest |
| <u>PT0H</u> | <u>PT0</u> | <u>Priority Level</u> | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | Lowest | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | Highest | | | | | | | | | | | | | | | | | | | | |
| 0 | PX0H | External interrupt 0 Priority High bit <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;"><u>PX0H</u></td> <td style="text-align: center;"><u>PX0</u></td> <td style="text-align: center;"><u>Priority Level</u></td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">Lowest</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">Highest</td> </tr> </table> | | | | | | <u>PX0H</u> | <u>PX0</u> | <u>Priority Level</u> | 0 | 0 | Lowest | 0 | 1 | | 1 | 0 | | 1 | 1 | Highest |
| <u>PX0H</u> | <u>PX0</u> | <u>Priority Level</u> | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | Lowest | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | Highest | | | | | | | | | | | | | | | | | | | | |

Reset Value = XX00 0000b

Not bit addressable

6.6 Idle mode

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

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

The flag bits GF0 and GF1 can be used to give an indication if an interrupt occurred during normal operation or during an Idle. For example, an instruction that activates Idle can also set one or both flag bits. When Idle is terminated by an interrupt, the interrupt service routine can examine the flag bits.

The other way of terminating the Idle mode is with a hardware reset. Since the clock oscillator is still running, the hardware reset needs to be held active for only two machine cycles (24 oscillator periods) to complete the reset.

6.7 Power-Down Mode

To save maximum power, a power-down mode can be invoked by software (Refer to Table 9., PCON register).

In power-down mode, the oscillator is stopped and the instruction that invoked power-down mode is the last instruction executed. The internal RAM and SFRs retain their value until the power-down mode is terminated. V_{CC} can be lowered to save further power. Either a hardware reset or an external interrupt can cause an exit from power-down. 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.

Only external interrupts $\overline{INT0}$ and $\overline{INT1}$ are useful to exit from power-down. For that, interrupt must be enabled and configured as level or edge sensitive interrupt input.

Holding the pin low restarts the oscillator but bringing the pin high completes the exit as detailed in Figure 10. When both interrupts are enabled, the oscillator restarts as soon as one of the two inputs is held low and power down exit will be completed when the first input will be released. In this case the higher priority interrupt service routine is executed.

Once the interrupt is serviced, the next instruction to be executed after RETI will be the one following the instruction that put TS80C52X2 into power-down mode.

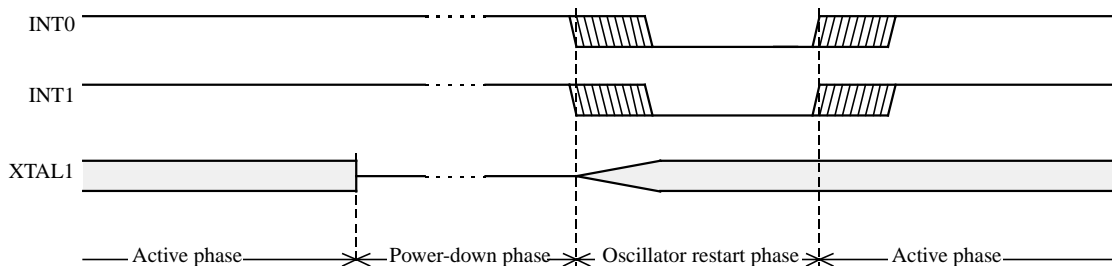


Figure 10. Power-Down Exit Waveform

Exit from power-down by reset redefines all the SFRs, exit from power-down by external interrupt does not affect the SFRs.

Exit from power-down by either reset or external interrupt does not affect the internal RAM content.

NOTE: If idle mode is activated with power-down mode (IDL and PD bits set), the exit sequence is unchanged, when execution is vectored to interrupt, PD and IDL bits are cleared and idle mode is not entered.

Table 14. The state of ports during idle and power-down modes

| Mode | Program Memory | ALE | $\overline{\text{PSEN}}$ | PORT0 | PORT1 | PORT2 | PORT3 |
|------------|----------------|-----|--------------------------|------------|-----------|-----------|-----------|
| Idle | Internal | 1 | 1 | Port Data* | Port Data | Port Data | Port Data |
| Idle | External | 1 | 1 | Floating | Port Data | Address | Port Data |
| Power Down | Internal | 0 | 0 | Port Data* | Port Data | Port Data | Port Data |
| Power Down | External | 0 | 0 | Floating | Port Data | Port Data | Port Data |

* Port 0 can force a "zero" level. A "one" will leave port floating.

6.8 ONCE™ Mode (ON Chip Emulation)

The ONCE mode facilitates testing and debugging of systems using TS80C52X2 without removing the circuit from the board. The ONCE mode is invoked by driving certain pins of the TS80C52X2; the following sequence must be exercised:

- Pull ALE low while the device is in reset (RST high) and $\overline{\text{PSEN}}$ is high.
- Hold ALE low as RST is deactivated.

While the TS80C52X2 is in ONCE mode, an emulator or test CPU can be used to drive the circuit Table 26 shows the status of the port pins during ONCE mode.

Normal operation is restored when normal reset is applied.

Table 15. External Pin Status during ONCE Mode

| ALE | $\overline{\text{PSEN}}$ | Port 0 | Port 1 | Port 2 | Port 3 | XTAL1/2 |
|--------------|--------------------------|--------|--------------|--------------|--------------|---------|
| Weak pull-up | Weak pull-up | Float | Weak pull-up | Weak pull-up | Weak pull-up | Active |

6.9 Power-Off Flag

The power-off flag allows the user to distinguish between a “cold start” reset and a “warm start” reset.

A cold start reset is the one induced by V_{CC} switch-on. A warm start reset occurs while V_{CC} is still applied to the device and could be generated for example by an exit from power-down.

The power-off flag (POF) is located in PCON register (See Table 16.). POF is set by hardware when V_{CC} rises from 0 to its nominal voltage. The POF can be set or cleared by software allowing the user to determine the type of reset.

The POF value is only relevant with a V_{cc} range from 4.5V to 5.5V. For lower V_{cc} value, reading POF bit will return indeterminate value.

Table 16. PCON Register

PCON - Power Control Register (87h)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|--------------|---|-----|-----|-----|----|-----|
| SMOD1 | SMOD0 | - | POF | GF1 | GF0 | PD | IDL |
| Bit Number | Bit Mnemonic | Description | | | | | |
| 7 | SMOD1 | Serial port Mode bit 1 Set to select double baud rate in mode 1, 2 or 3. | | | | | |
| 6 | SMOD0 | Serial port Mode bit 0 Clear to select SM0 bit in SCON register. Set to to select FE bit in SCON register. | | | | | |
| 5 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. | | | | | |
| 4 | POF | Power-Off Flag Clear to recognize next reset type. Set by hardware when V_{CC} rises from 0 to its nominal voltage. Can also be set by software. | | | | | |
| 3 | GF1 | General purpose Flag Cleared by user for general purpose usage. Set by user for general purpose usage. | | | | | |
| 2 | GF0 | General purpose Flag Cleared by user for general purpose usage. Set by user for general purpose usage. | | | | | |
| 1 | PD | Power-Down mode bit Cleared by hardware when reset occurs. Set to enter power-down mode. | | | | | |
| 0 | IDL | Idle mode bit Clear by hardware when interrupt or reset occurs. Set to enter idle mode. | | | | | |

Reset Value = 00X1 0000b

Not bit addressable

6.10 Reduced EMI Mode

The ALE signal is used to demultiplex address and data buses on port 0 when used with external program or data memory. Nevertheless, during internal code execution, ALE signal is still generated. In order to reduce EMI, ALE signal can be disabled by setting AO bit.

The AO bit is located in AUXR register at bit location 0. As soon as AO is set, ALE is no longer output but remains active during MOVX and MOVC instructions and external fetches. During ALE disabling, ALE pin is weakly pulled high.

Table 17. AUXR Register

AUXR - Auxiliary Register (8Eh)

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

| Bit Number | Bit Mnemonic | Description |
|------------|--------------|---|
| 7 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 6 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 5 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 4 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 3 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 2 | - | Reserved The value read from this bit is indeterminate. Do not set this bit. |
| 1 | EXTRAM | EXTRAM bit See Table 7. |
| 0 | AO | ALE Output bit Clear to restore ALE operation during internal fetches. Set to disable ALE operation during internal fetches. |

Reset Value = XXXX XX00b

Not bit addressable

7. TS80C52X2

7.1 ROM Structure

The TS80C52X2 ROM memory is divided in three different arrays:

- the code array: 8 Kbytes.
- the encryption array: 64 bytes.
- the signature array: 4 bytes.

7.2 ROM Lock System

The program Lock system, when programmed, protects the on-chip program against software piracy.

7.2.1 Encryption Array

Within the ROM array are 64 bytes of encryption array that are initially unprogrammed (all FF's). Every time a byte is addressed during program verify, 6 address lines are used to select a byte of the encryption array. This byte is then exclusive-NOR'ed (XNOR) with the code byte, creating an encrypted verify byte. The algorithm, with the encryption array in the unprogrammed state, will return the code in its original, unmodified form.

When using the encryption array, one important factor needs to be considered. If a byte has the value FFh, verifying the byte will produce the encryption byte value. If a large block (>64 bytes) of code is left unprogrammed, a verification routine will display the content of the encryption array. For this reason all the unused code bytes should be programmed with random values. This will ensure program protection.

7.2.2 Program Lock Bits

The lock bits when programmed according to Table 18. will provide different level of protection for the on-chip code and data.

Table 18. Program Lock bits

| Program Lock Bits | | | | Protection description |
|-------------------|-----|-----|-----|---|
| Security level | LB1 | LB2 | LB3 | |
| 1 | U | U | U | No program lock features enabled. Code verify will still be encrypted by the encryption array if programmed. MOVC instruction executed from external program memory returns non encrypted data. |
| 2 | P | U | U | MOVC instruction executed from external program memory are disabled from fetching code bytes from internal memory, EA is sampled and latched on reset. |

U: unprogrammed

P: programmed

7.2.3 Signature bytes

The TS80C52X2 contains 4 factory programmed signatures bytes. To read these bytes, perform the process described in section 9.

7.2.4 Verify Algorithm

Refer to 8.3.4

8. TS87C52X2

8.1 EPROM Structure

The TS87C52X2 is divided in two different arrays:

- the code array: 8 Kbytes.
- the encryption array: 64 bytes.

In addition a third non programmable array is implemented:

- the signature array: 4 bytes.

8.2 EPROM Lock System

The program Lock system, when programmed, protects the on-chip program against software piracy.

8.2.1 Encryption Array

Within the EPROM array are 64 bytes of encryption array that are initially unprogrammed (all FF's). Every time a byte is addressed during program verify, 6 address lines are used to select a byte of the encryption array. This byte is then exclusive-NOR'ed (XNOR) with the code byte, creating an encrypted verify byte. The algorithm, with the encryption array in the unprogrammed state, will return the code in its original, unmodified form.

When using the encryption array, one important factor needs to be considered. If a byte has the value FFh, verifying the byte will produce the encryption byte value. If a large block (>64 bytes) of code is left unprogrammed, a verification routine will display the content of the encryption array. For this reason all the unused code bytes should be programmed with random values. This will ensure program protection.

8.2.2 Program Lock Bits

The three lock bits, when programmed according to Table 19., will provide different level of protection for the on-chip code and data.

Table 19. Program Lock bits

| Program Lock Bits | | | | Protection description |
|-------------------|-----|-----|-----|---|
| Security level | LB1 | LB2 | LB3 | |
| 1 | U | U | U | No program lock features enabled. Code verify will still be encrypted by the encryption array if programmed. MOVC instruction executed from external program memory returns non encrypted data. |
| 2 | P | U | U | MOVC instruction executed from external program memory are disabled from fetching code bytes from internal memory, \overline{EA} is sampled and latched on reset, and further programming of the EPROM is disabled. |
| 3 | U | P | U | Same as 2, also verify is disabled. |
| 4 | U | U | P | Same as 3, also external execution is disabled. |

U: unprogrammed,
P: programmed

WARNING: Security level 2 and 3 should only be programmed after EPROM and Core verification.

8.2.3 Signature bytes

The TS80/87C52X2 contains 4 factory programmed signatures bytes. To read these bytes, perform the process described in section 9.

8.3 EPROM Programming

8.3.1 Set-up modes

In order to program and verify the EPROM or to read the signature bytes, the TS87C52X2 is placed in specific set-up modes (See Figure 11.).

Control and program signals must be held at the levels indicated in Table 35.

8.3.2 Definition of terms

Address Lines:P1.0-P1.7, P2.0-P2.4 respectively for A0-A12

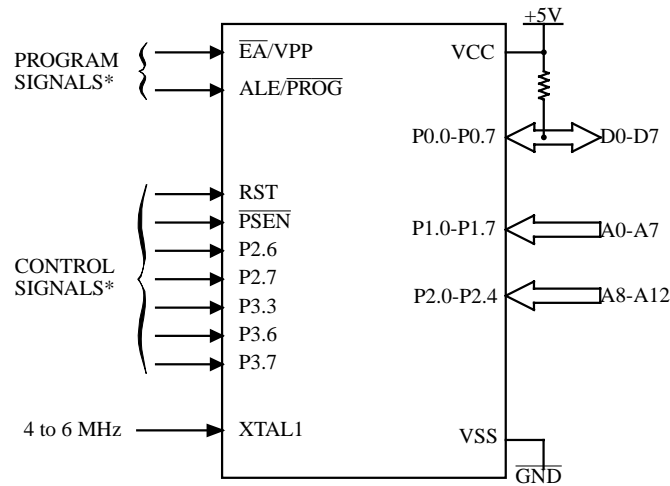
Data Lines:P0.0-P0.7 for D0-D7

Control Signals:RST, $\overline{\text{PSEN}}$, P2.6, P2.7, P3.3, P3.6, P3.7.

Program Signals:ALE/ $\overline{\text{PROG}}$, $\overline{\text{EA}}$ /VPP.

Table 20. EPROM Set-Up Modes

| Mode | RST | $\overline{\text{PSEN}}$ | ALE/ $\overline{\text{PROG}}$ | $\overline{\text{EA}}$ / VPP | P2.6 | P2.7 | P3.3 | P3.6 | P3.7 |
|---|-----|--------------------------|----------------------------------|---------------------------------|------|------|------|------|------|
| Program Code data | 1 | 0 | | 12.75V | 0 | 1 | 1 | 1 | 1 |
| Verify Code data | 1 | 0 | 1 | 1 | 0 | | 0 | 1 | 1 |
| Program Encryption Array Address 0-3Fh | 1 | 0 | | 12.75V | 0 | 1 | 1 | 0 | 1 |
| Read Signature Bytes | 1 | 0 | 1 | 1 | 0 | | 0 | 0 | 0 |
| Program Lock bit 1 | 1 | 0 | | 12.75V | 1 | 1 | 1 | 1 | 1 |
| Program Lock bit 2 | 1 | 0 | | 12.75V | 1 | 1 | 1 | 0 | 0 |
| Program Lock bit 3 | 1 | 0 | | 12.75V | 1 | 0 | 1 | 1 | 0 |



* See Table 31. for proper value on these inputs

Figure 11. Set-Up Modes Configuration

8.3.3 Programming Algorithm

The Improved Quick Pulse algorithm is based on the Quick Pulse algorithm and decreases the number of pulses applied during byte programming from 25 to 1.

To program the TS87C52X2 the following sequence must be exercised:

- Step 1: Activate the combination of control signals.
- Step 2: Input the valid address on the address lines.
- Step 3: Input the appropriate data on the data lines.
- Step 4: Raise \overline{EA}/VPP from VCC to VPP (typical 12.75V).
- Step 5: Pulse ALE/\overline{PROG} once.
- Step 6: Lower \overline{EA}/VPP from VPP to VCC

Repeat step 2 through 6 changing the address and data for the entire array or until the end of the object file is reached (See Figure 12.).

8.3.4 Verify algorithm

Code array verify must be done after each byte or block of bytes is programmed. In either case, a complete verify of the programmed array will ensure reliable programming of the TS87C52X2.

P 2.7 is used to enable data output.

To verify the TS87C52X2 code the following sequence must be exercised:

- Step 1: Activate the combination of program and control signals.
- Step 2: Input the valid address on the address lines.
- Step 3: Read data on the data lines.

Repeat step 2 through 3 changing the address for the entire array verification (See Figure 12.)

The encryption array cannot be directly verified. Verification of the encryption array is done by observing that the code array is well encrypted.

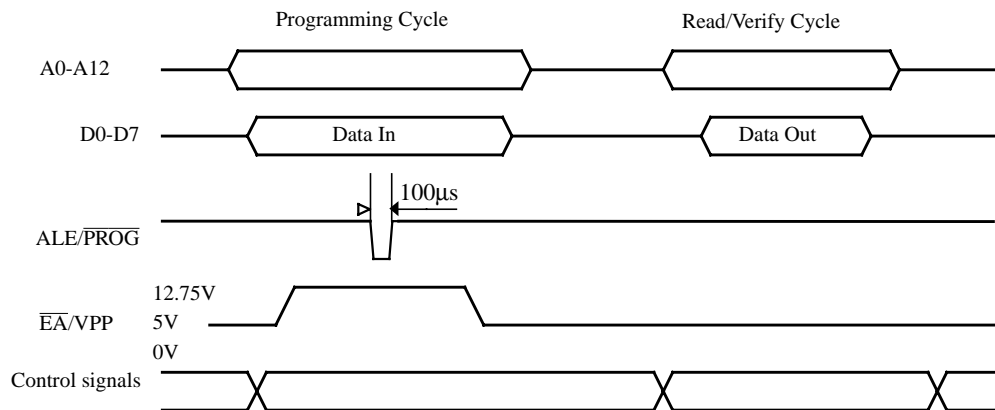


Figure 12. Programming and Verification Signal's Waveform

8.4 EPROM Erasure (Windowed Packages Only)

Erasing the EPROM erases the code array, the encryption array and the lock bits returning the parts to full functionality.

Erasure leaves all the EPROM cells in a 1's state (FF).

8.4.1 Erasure Characteristics

The recommended erasure procedure is exposure to ultraviolet light (at 2537 Å) to an integrated dose at least 15 W-sec/cm². Exposing the EPROM to an ultraviolet lamp of 12,000 µW/cm² rating for 30 minutes, at a distance of about 25 mm, should be sufficient. An exposure of 1 hour is recommended with most of standard erasers.

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelength shorter than approximately 4,000 Å. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room-level fluorescent lighting) could cause inadvertent erasure. If an application subjects the device to this type of exposure, it is suggested that an opaque label be placed over the window.

9. Signature Bytes

The TS80/87C52X2 has four signature bytes in location 30h, 31h, 60h and 61h. To read these bytes follow the procedure for EPROM verify but activate the control lines provided in Table 31. for Read Signature Bytes. Table 35. shows the content of the signature byte for the TS80/87C52X2.

Table 21. Signature Bytes Content

| Location | Contents | Comment |
|----------|----------|--------------------------|
| 30h | 58h | Manufacturer Code: TEMIC |
| 31h | 57h | Family Code: C51 X2 |
| 60h | 2Dh | Product name: TS80C52X2 |
| 60h | ADh | Product name: TS87C52X2 |
| 60h | 20h | Product name: TS80C32X2 |
| 61h | FFh | Product revision number |

10. Electrical Characteristics

10.1 Absolute Maximum Ratings ⁽¹⁾

Ambiant Temperature Under Bias:

| | |
|---|-----------------------------------|
| C = commercial | 0°C to 70°C |
| I = industrial | -40°C to 85°C |
| Storage Temperature | -65°C to + 150°C |
| Voltage on V _{CC} to V _{SS} | -0.5 V to + 7 V |
| Voltage on V _{PP} to V _{SS} | -0.5 V to + 13 V |
| Voltage on Any Pin to V _{SS} | -0.5 V to V _{CC} + 0.5 V |
| Power Dissipation | 1 W ⁽²⁾ |

NOTES

1. Stresses at or 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 other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions may affect device reliability.
2. This value is based on the maximum allowable die temperature and the thermal resistance of the package.

10.2 Power consumption measurement

Since the introduction of the first C51 devices, every manufacturer made operating I_{cc} measurements under reset, which made sense for the designs where the CPU was running under reset. In TEMIC new devices, the CPU is no more active during reset, so the power consumption is very low but is not really representative of what will happen in the customer system. That's why, while keeping measurements under Reset, TEMIC presents a new way to measure the operating I_{cc}:

Using an internal test ROM, the following code is executed:

Label: SJMP Label (80 FE)

Ports 1, 2, 3 are disconnected, Port 0 is tied to FFh, EA = V_{cc}, RST = V_{ss}, XTAL2 is not connected and XTAL1 is driven by the clock.

This is much more representative of the real operating I_{cc}.

10.3 DC Parameters for Standard Voltage

$T_A = 0^\circ\text{C}$ to $+70^\circ\text{C}$; $V_{SS} = 0\text{ V}$; $V_{CC} = 5\text{ V} \pm 10\%$; $F = 0$ to 40 MHz .

$T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$; $V_{SS} = 0\text{ V}$; $V_{CC} = 5\text{ V} \pm 10\%$; $F = 0$ to 40 MHz .

Table 22. DC Parameters in Standard Voltage

| Symbol | Parameter | Min | Typ | Max | Unit | Test Conditions |
|-----------|--|--|-------------------|--------------------|---------------|---|
| V_{IL} | Input Low Voltage | -0.5 | | $0.2 V_{CC} - 0.1$ | V | |
| V_{IH} | Input High Voltage except XTAL1, RST | $0.2 V_{CC} + 0.9$ | | $V_{CC} + 0.5$ | V | |
| V_{IH1} | Input High Voltage, XTAL1, RST | $0.7 V_{CC}$ | | $V_{CC} + 0.5$ | V | |
| V_{OL} | Output Low Voltage, ports 1, 2, 3 ⁽⁶⁾ | | | 0.3 0.45 1.0 | V V V | $I_{OL} = 100\ \mu\text{A}^{(4)}$ $I_{OL} = 1.6\ \text{mA}^{(4)}$ $I_{OL} = 3.5\ \text{mA}^{(4)}$ |
| V_{OL1} | Output Low Voltage, port 0 ⁽⁶⁾ | | | 0.3 0.45 1.0 | V V V | $I_{OL} = 200\ \mu\text{A}^{(4)}$ $I_{OL} = 3.2\ \text{mA}^{(4)}$ $I_{OL} = 7.0\ \text{mA}^{(4)}$ |
| V_{OL2} | Output Low Voltage, ALE, $\overline{\text{PSEN}}$ | | | 0.3 0.45 1.0 | V V V | $I_{OL} = 100\ \mu\text{A}^{(4)}$ $I_{OL} = 1.6\ \text{mA}^{(4)}$ $I_{OL} = 3.5\ \text{mA}^{(4)}$ |
| V_{OH} | Output High Voltage, ports 1, 2, 3 | $V_{CC} - 0.3$ $V_{CC} - 0.7$ $V_{CC} - 1.5$ | | | V V V | $I_{OH} = -10\ \mu\text{A}$ $I_{OH} = -30\ \mu\text{A}$ $I_{OH} = -60\ \mu\text{A}$ $V_{CC} = 5\text{ V} \pm 10\%$ |
| V_{OH1} | Output High Voltage, port 0 | $V_{CC} - 0.3$ $V_{CC} - 0.7$ $V_{CC} - 1.5$ | | | V V V | $I_{OH} = -200\ \mu\text{A}$ $I_{OH} = -3.2\ \text{mA}$ $I_{OH} = -7.0\ \text{mA}$ $V_{CC} = 5\text{ V} \pm 10\%$ |
| V_{OH2} | Output High Voltage, ALE, $\overline{\text{PSEN}}$ | $V_{CC} - 0.3$ $V_{CC} - 0.7$ $V_{CC} - 1.5$ | | | V V V | $I_{OH} = -100\ \mu\text{A}$ $I_{OH} = -1.6\ \text{mA}$ $I_{OH} = -3.5\ \text{mA}$ $V_{CC} = 5\text{ V} \pm 10\%$ |
| R_{RST} | RST Pulldown Resistor | 50 | 90 ⁽⁵⁾ | 200 | k Ω | |
| I_{IL} | Logical 0 Input Current ports 1, 2 and 3 | | | -50 | μA | $V_{in} = 0.45\text{ V}$ |
| I_{LI} | Input Leakage Current | | | ± 10 | μA | $0.45\text{ V} < V_{in} < V_{CC}$ |
| I_{TL} | Logical 1 to 0 Transition Current, ports 1, 2, 3 | | | -650 | μA | $V_{in} = 2.0\text{ V}$ |
| C_{IO} | Capacitance of I/O Buffer | | | 10 | pF | $F_c = 1\text{ MHz}$ $T_A = 25^\circ\text{C}$ |
| I_{PD} | Power Down Current | | 20 ⁽⁵⁾ | 50 | μA | $2.0\text{ V} < V_{CC} < 5.5\text{ V}^{(3)}$ |

| Symbol | Parameter | Min | Typ | Max | Unit | Test Conditions |
|----------------------------|--|-----|-----|---|------|--------------------------------|
| I_{CC} under RESET | Power Supply Current Maximum values, X1 mode: ⁽⁷⁾ | | | 1 + 0.4 Freq (MHz) @12MHz 5.8 @16MHz 7.4 | mA | $V_{CC} = 5.5 \text{ V}^{(1)}$ |
| I_{CC} operating | Power Supply Current Maximum values, X1 mode: ⁽⁷⁾ | | | 3 + 0.6 Freq (MHz) @12MHz 10.2 @16MHz 12.6 | mA | $V_{CC} = 5.5 \text{ V}^{(8)}$ |
| I_{CC} idle | Power Supply Current Maximum values, X1 mode: ⁽⁷⁾ | | | 0.25+0.3 Freq (MHz) @12MHz 3.9 @16MHz 5.1 | mA | $V_{CC} = 5.5 \text{ V}^{(2)}$ |

10.4 DC Parameters for Low Voltage

$T_A = 0^\circ\text{C}$ to $+70^\circ\text{C}$; $V_{SS} = 0 \text{ V}$; $V_{CC} = 2.7 \text{ V}$ to $5.5 \text{ V} \pm 10\%$; $F = 0$ to 30 MHz .

$T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$; $V_{SS} = 0 \text{ V}$; $V_{CC} = 2.7 \text{ V}$ to $5.5 \text{ V} \pm 10\%$; $F = 0$ to 30 MHz .

Table 23. DC Parameters for Low Voltage

| Symbol | Parameter | Min | Typ | Max | Unit | Test Conditions |
|----------------------------|--|--------------------|--------------------------|---|---------------|--|
| V_{IL} | Input Low Voltage | -0.5 | | $0.2 V_{CC} - 0.1$ | V | |
| V_{IH} | Input High Voltage except XTAL1, RST | $0.2 V_{CC} + 0.9$ | | $V_{CC} + 0.5$ | V | |
| V_{IHI} | Input High Voltage, XTAL1, RST | $0.7 V_{CC}$ | | $V_{CC} + 0.5$ | V | |
| V_{OL} | Output Low Voltage, ports 1, 2, 3 ⁽⁶⁾ | | | 0.45 | V | $I_{OL} = 0.8 \text{ mA}^{(4)}$ |
| V_{OL1} | Output Low Voltage, port 0, ALE, $\overline{\text{PSEN}}$ ⁽⁶⁾ | | | 0.45 | V | $I_{OL} = 1.6 \text{ mA}^{(4)}$ |
| V_{OH} | Output High Voltage, ports 1, 2, 3 | $0.9 V_{CC}$ | | | V | $I_{OH} = -10 \mu\text{A}$ |
| V_{OH1} | Output High Voltage, port 0, ALE, $\overline{\text{PSEN}}$ | $0.9 V_{CC}$ | | | V | $I_{OH} = -40 \mu\text{A}$ |
| I_{IL} | Logical 0 Input Current ports 1, 2 and 3 | | | -50 | μA | $V_{in} = 0.45 \text{ V}$ |
| I_{LI} | Input Leakage Current | | | ± 10 | μA | $0.45 \text{ V} < V_{in} < V_{CC}$ |
| I_{TL} | Logical 1 to 0 Transition Current, ports 1, 2, 3 | | | -650 | μA | $V_{in} = 2.0 \text{ V}$ |
| R_{RST} | RST Pulldown Resistor | 50 | $90^{(5)}$ | 200 | k Ω | |
| CIO | Capacitance of I/O Buffer | | | 10 | pF | $F_c = 1 \text{ MHz}$ $T_A = 25^\circ\text{C}$ |
| I_{PD} | Power Down Current | | $20^{(5)}$ $10^{(5)}$ | 50 30 | μA | $V_{CC} = 2.0 \text{ V}$ to $5.5 \text{ V}^{(3)}$ $V_{CC} = 2.0 \text{ V}$ to $3.3 \text{ V}^{(3)}$ |
| I_{CC} under RESET | Power Supply Current Maximum values, X1 mode: ⁽⁷⁾ | | | 1 + 0.2 Freq (MHz) @12MHz 3.4 @16MHz 4.2 | mA | $V_{CC} = 3.3 \text{ V}^{(1)}$ |

| Symbol | Parameter | Min | Typ | Max | Unit | Test Conditions |
|--------------------|--|-----|-----|---|------|--------------------------------|
| I_{CC} operating | Power Supply Current Maximum values, X1 mode: ⁽⁷⁾ | | | 1 + 0.3 Freq (MHz) @ 12MHz 4.6 @ 16MHz 5.8 | mA | $V_{CC} = 3.3 \text{ V}^{(8)}$ |
| I_{CC} idle | Power Supply Current Maximum values, X1 mode: ⁽⁷⁾ | | | 0.15 Freq (MHz) + 0.2 @ 12MHz 2 @ 16MHz 2.6 | mA | $V_{CC} = 3.3 \text{ V}^{(2)}$ |

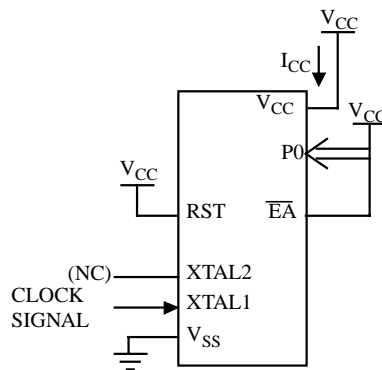
NOTES

- I_{CC} under reset is measured with all output pins disconnected; XTAL1 driven with T_{CLCH} , $T_{CHCL} = 5 \text{ ns}$ (see Figure 17.), $V_{IL} = V_{SS} + 0.5 \text{ V}$, $V_{IH} = V_{CC} - 0.5\text{V}$; XTAL2 N.C.; $\overline{EA} = RST = \text{Port } 0 = V_{CC}$; I_{CC} would be slightly higher if a crystal oscillator used.
- Idle I_{CC} is measured with all output pins disconnected; XTAL1 driven with T_{CLCH} , $T_{CHCL} = 5 \text{ ns}$, $V_{IL} = V_{SS} + 0.5 \text{ V}$, $V_{IH} = V_{CC} - 0.5 \text{ V}$; XTAL2 N.C; Port 0 = V_{CC} ; $\overline{EA} = RST = V_{SS}$ (see Figure 15.).
- Power Down I_{CC} is measured with all output pins disconnected; $\overline{EA} = V_{SS}$, PORT 0 = V_{CC} ; XTAL2 N.C.; RST = V_{SS} (see Figure 16.).
- Capacitance loading on Ports 0 and 2 may cause spurious noise pulses to be superimposed on the V_{OL} s of ALE and Ports 1 and 3. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make 1 to 0 transitions during bus operation. In the worst cases (capacitive loading 100pF), the noise pulse on the ALE line may exceed 0.45V with maxi V_{OL} peak 0.6V. A Schmitt Trigger use is not necessary.
- Typicals are based on a limited number of samples and are not guaranteed. The values listed are at room temperature and 5V.
- Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:

- Maximum I_{OL} per port pin: 10 mA
- Maximum I_{OL} per 8-bit port:
 - Port 0: 26 mA
 - Ports 1, 2 and 3: 15 mA
- Maximum total I_{OL} for all output pins: 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.

- For other values, please contact your sales office.
- Operating I_{CC} is measured with all output pins disconnected; XTAL1 driven with T_{CLCH} , $T_{CHCL} = 5 \text{ ns}$ (see Figure 17.), $V_{IL} = V_{SS} + 0.5 \text{ V}$, $V_{IH} = V_{CC} - 0.5\text{V}$; XTAL2 N.C.; $\overline{EA} = \text{Port } 0 = V_{CC}$; RST = V_{SS} . The internal ROM runs the code 80 FE (label: SJMP label). I_{CC} would be slightly higher if a crystal oscillator is used. Measurements are made with OTP products when possible, which is the worst case.



All other pins are disconnected.

Figure 13. I_{CC} Test Condition, under reset

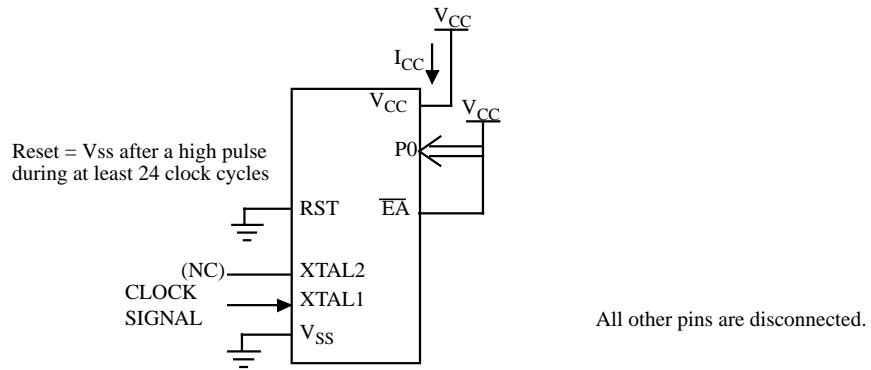


Figure 14. Operating I_{CC} Test Condition

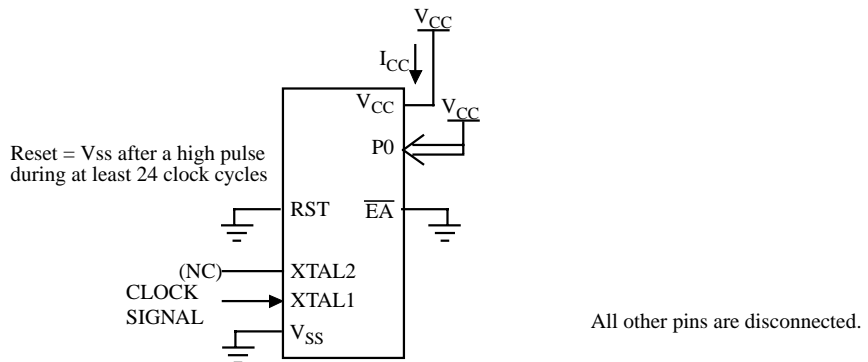


Figure 15. I_{CC} Test Condition, Idle Mode

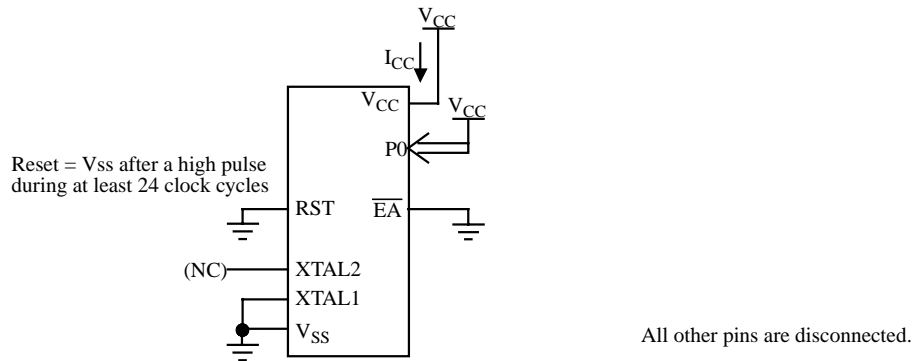


Figure 16. I_{CC} Test Condition, Power-Down Mode

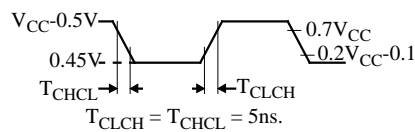


Figure 17. Clock Signal Waveform for I_{CC} Tests in Active and Idle Modes

10.5 AC Parameters

10.5.1 Explanation of the AC Symbols

Each timing symbol has 5 characters. The first character is always a “T” (stands for time). The other characters, depending on their positions, stand for the name of a signal or the logical status of that signal. The following is a list of all the characters and what they stand for.

Example: T_{AVLL} = Time for Address Valid to ALE Low.

T_{LLPL} = Time for ALE Low to \overline{PSEN} Low.

$T_A = 0$ to $+70^\circ\text{C}$ (commercial temperature range); $V_{SS} = 0$ V; $V_{CC} = 5$ V \pm 10%; -M and -V ranges.

$T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$ (industrial temperature range); $V_{SS} = 0$ V; $V_{CC} = 5$ V \pm 10%; -M and -V ranges.

$T_A = 0$ to $+70^\circ\text{C}$ (commercial temperature range); $V_{SS} = 0$ V; 2.7 V $< V_{CC} < 5.5$ V; -L range.

$T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$ (industrial temperature range); $V_{SS} = 0$ V; 2.7 V $< V_{CC} < 5.5$ V; -L range.

Table 24. gives the maximum applicable load capacitance for Port 0, Port 1, 2 and 3, and ALE and \overline{PSEN} signals. Timings will be guaranteed if these capacitances are respected. Higher capacitance values can be used, but timings will then be degraded.

Table 24. Load Capacitance versus speed range, in pF

| | -M | -V | -L |
|-------------------------|-----|----|-----|
| Port 0 | 100 | 50 | 100 |
| Port 1, 2, 3 | 80 | 50 | 80 |
| ALE / \overline{PSEN} | 100 | 30 | 100 |

Table 26., Table 29. and Table 32. give the description of each AC symbols.

Table 27., Table 30. and Table 33. give for each range the AC parameter.

Table 28., Table 31. and Table 34. give the frequency derating formula of the AC parameter. To calculate each AC symbols, take the x value corresponding to the speed grade you need (-M, -V or -L) and replace this value in the formula. Values of the frequency must be limited to the corresponding speed grade:

Table 25. Max frequency for derating formula regarding the speed grade

| | -M X1 mode | -M X2 mode | -V X1 mode | -V X2 mode | -L X1 mode | -L X2 mode |
|------------|------------|------------|------------|------------|------------|------------|
| Freq (MHz) | 40 | 20 | 40 | 30 | 30 | 20 |
| T (ns) | 25 | 50 | 25 | 33.3 | 33.3 | 50 |

Example:

T_{LLIV} in X2 mode for a -V part at 20 MHz ($T = 1/20^{E6} = 50$ ns):

x= 25 (Table 28.)

T= 50ns

$T_{LLIV} = 2T - x = 2 \times 50 - 25 = 75\text{ns}$

10.5.2 External Program Memory Characteristics

Table 26. Symbol Description

| Symbol | Parameter |
|-------------------|--|
| T | Oscillator clock period |
| T _{LHLL} | ALE pulse width |
| T _{AVLL} | Address Valid to ALE |
| T _{LLAX} | Address Hold After ALE |
| T _{LLIV} | ALE to Valid Instruction In |
| T _{LLPL} | ALE to $\overline{\text{PSEN}}$ |
| T _{PLPH} | $\overline{\text{PSEN}}$ Pulse Width |
| T _{PLIV} | $\overline{\text{PSEN}}$ to Valid Instruction In |
| T _{PXIX} | Input Instruction Hold After $\overline{\text{PSEN}}$ |
| T _{PXIZ} | Input Instruction Float After $\overline{\text{PSEN}}$ |
| T _{PXAV} | $\overline{\text{PSEN}}$ to Address Valid |
| T _{AVIV} | Address to Valid Instruction In |
| T _{PLAZ} | $\overline{\text{PSEN}}$ Low to Address Float |

Table 27. AC Parameters for Fix Clock

| Speed | -M 40 MHz | | -V X2 mode 30 MHz 60 MHz equiv. | | -V standard mode 40 MHz | | -L X2 mode 20 MHz 40 MHz equiv. | | -L standard mode 30 MHz | | Units |
|-------------------|--------------|-----|--|-----|-------------------------------|-----|--|-----|-------------------------------|-----|-------|
| | Min | Max | Min | Max | Min | Max | Min | Max | Min | Max | |
| T | 25 | | 33 | | 25 | | 50 | | 33 | | ns |
| T _{LHLL} | 40 | | 25 | | 42 | | 35 | | 52 | | ns |
| T _{AVLL} | 10 | | 4 | | 12 | | 5 | | 13 | | ns |
| T _{LLAX} | 10 | | 4 | | 12 | | 5 | | 13 | | ns |
| T _{LLIV} | | 70 | | 45 | | 78 | | 65 | | 98 | ns |
| T _{LLPL} | 15 | | 9 | | 17 | | 10 | | 18 | | ns |
| T _{PLPH} | 55 | | 35 | | 60 | | 50 | | 75 | | ns |
| T _{PLIV} | | 35 | | 25 | | 50 | | 30 | | 55 | ns |
| T _{PXIX} | 0 | | 0 | | 0 | | 0 | | 0 | | ns |
| T _{PXIZ} | | 18 | | 12 | | 20 | | 10 | | 18 | ns |
| T _{AVIV} | | 85 | | 53 | | 95 | | 80 | | 122 | ns |
| T _{PLAZ} | | 10 | | 10 | | 10 | | 10 | | 10 | ns |

Table 28. AC Parameters for a Variable Clock: derating formula

| Symbol | Type | Standard Clock | X2 Clock | -M | -V | -L | Units |
|------------|------|----------------|-------------|----|----|----|-------|
| T_{LHLL} | Min | $2 T - x$ | $T - x$ | 10 | 8 | 15 | ns |
| T_{AVLL} | Min | $T - x$ | $0.5 T - x$ | 15 | 13 | 20 | ns |
| T_{LLAX} | Min | $T - x$ | $0.5 T - x$ | 15 | 13 | 20 | ns |
| T_{LLIV} | Max | $4 T - x$ | $2 T - x$ | 30 | 22 | 35 | ns |
| T_{LLPL} | Min | $T - x$ | $0.5 T - x$ | 10 | 8 | 15 | ns |
| T_{PLPH} | Min | $3 T - x$ | $1.5 T - x$ | 20 | 15 | 25 | ns |
| T_{PLIV} | Max | $3 T - x$ | $1.5 T - x$ | 40 | 25 | 45 | ns |
| T_{PXIX} | Min | x | x | 0 | 0 | 0 | ns |
| T_{PXIZ} | Max | $T - x$ | $0.5 T - x$ | 7 | 5 | 15 | ns |
| T_{AVIV} | Max | $5 T - x$ | $2.5 T - x$ | 40 | 30 | 45 | ns |
| T_{PLAZ} | Max | x | x | 10 | 10 | 10 | ns |

10.5.3 External Program Memory Read Cycle

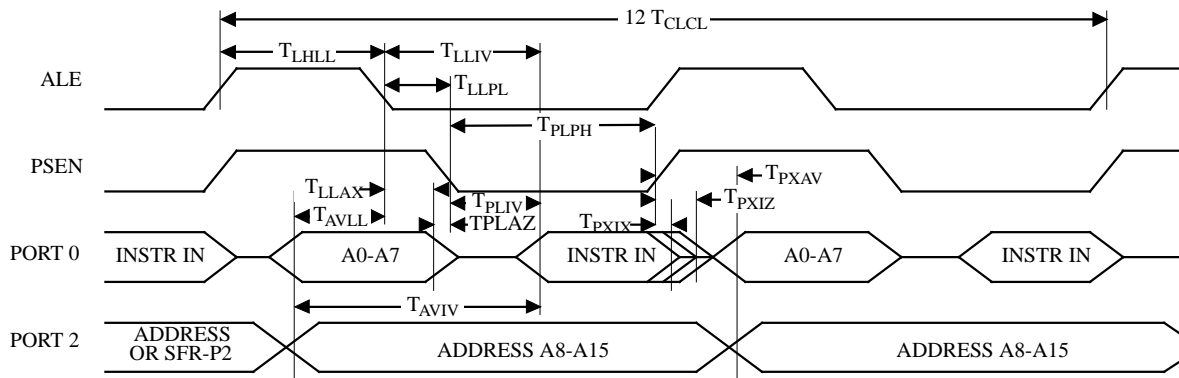


Figure 18. External Program Memory Read Cycle

10.5.4 External Data Memory Characteristics

Table 29. Symbol Description

| Symbol | Parameter |
|------------|---|
| T_{RLRH} | \overline{RD} Pulse Width |
| T_{WLWH} | \overline{WR} Pulse Width |
| T_{RLDV} | \overline{RD} to Valid Data In |
| T_{RHDX} | Data Hold After \overline{RD} |
| T_{RHDZ} | Data Float After \overline{RD} |
| T_{LLDV} | ALE to Valid Data In |
| T_{AVDV} | Address to Valid Data In |
| T_{LLWL} | ALE to \overline{WR} or \overline{RD} |
| T_{AVWL} | Address to \overline{WR} or \overline{RD} |
| T_{QVWX} | Data Valid to \overline{WR} Transition |
| T_{QVWH} | Data set-up to \overline{WR} High |
| T_{WHQX} | Data Hold After \overline{WR} |
| T_{RLAZ} | \overline{RD} Low to Address Float |
| T_{WHLH} | \overline{RD} or \overline{WR} High to ALE high |

Table 30. AC Parameters for a Fix Clock

| Speed | -M 40 MHz | | -V X2 mode 30 MHz 60 MHz equiv. | | -V standard mode 40 MHz | | -L X2 mode 20 MHz 40 MHz equiv. | | -L standard mode 30 MHz | | Units |
|-------------------|--------------|-----|--|-----|-------------------------------|-----|--|-----|-------------------------------|-----|-------|
| | Min | Max | Min | Max | Min | Max | Min | Max | Min | Max | |
| T _{RLRH} | 130 | | 85 | | 135 | | 125 | | 175 | | ns |
| T _{WLWH} | 130 | | 85 | | 135 | | 125 | | 175 | | ns |
| T _{RLDV} | | 100 | | 60 | | 102 | | 95 | | 137 | ns |
| T _{RHDX} | 0 | | 0 | | 0 | | 0 | | 0 | | ns |
| T _{RHDZ} | | 30 | | 18 | | 35 | | 25 | | 42 | ns |
| T _{LLDV} | | 160 | | 98 | | 165 | | 155 | | 222 | ns |
| T _{AVDV} | | 165 | | 100 | | 175 | | 160 | | 235 | ns |
| T _{LLWL} | 50 | 100 | 30 | 70 | 55 | 95 | 45 | 105 | 70 | 130 | ns |
| T _{AVWL} | 75 | | 47 | | 80 | | 70 | | 103 | | ns |
| T _{QVWX} | 10 | | 7 | | 15 | | 5 | | 13 | | ns |
| T _{QVWH} | 160 | | 107 | | 165 | | 155 | | 213 | | ns |
| T _{WHQX} | 15 | | 9 | | 17 | | 10 | | 18 | | ns |
| T _{RLAZ} | | 0 | | 0 | | 0 | | 0 | | 0 | ns |
| T _{WHLH} | 10 | 40 | 7 | 27 | 15 | 35 | 5 | 45 | 13 | 53 | ns |

Table 31. AC Parameters for a Variable Clock: derating formula

| Symbol | Type | Standard Clock | X2 Clock | -M | -V | -L | Units |
|------------|------|----------------|-----------|----|----|----|-------|
| T_{RLRH} | Min | 6 T - x | 3 T - x | 20 | 15 | 25 | ns |
| T_{WLWH} | Min | 6 T - x | 3 T - x | 20 | 15 | 25 | ns |
| T_{RLDV} | Max | 5 T - x | 2.5 T - x | 25 | 23 | 30 | ns |
| T_{RHDx} | Min | x | x | 0 | 0 | 0 | ns |
| T_{RHDz} | Max | 2 T - x | T - x | 20 | 15 | 25 | ns |
| T_{LLDV} | Max | 8 T - x | 4T - x | 40 | 35 | 45 | ns |
| T_{AVDV} | Max | 9 T - x | 4.5 T - x | 60 | 50 | 65 | ns |
| T_{LLWL} | Min | 3 T - x | 1.5 T - x | 25 | 20 | 30 | ns |
| T_{LLWL} | Max | 3 T + x | 1.5 T + x | 25 | 20 | 30 | ns |
| T_{AVWL} | Min | 4 T - x | 2 T - x | 25 | 20 | 30 | ns |
| T_{QVWX} | Min | T - x | 0.5 T - x | 15 | 10 | 20 | ns |
| T_{QVWH} | Min | 7 T - x | 3.5 T - x | 15 | 10 | 20 | ns |
| T_{WHQx} | Min | T - x | 0.5 T - x | 10 | 8 | 15 | ns |
| T_{RLAZ} | Max | x | x | 0 | 0 | 0 | ns |
| T_{WHLH} | Min | T - x | 0.5 T - x | 15 | 10 | 20 | ns |
| T_{WHLH} | Max | T + x | 0.5 T + x | 15 | 10 | 20 | ns |

10.5.5 External Data Memory Write Cycle

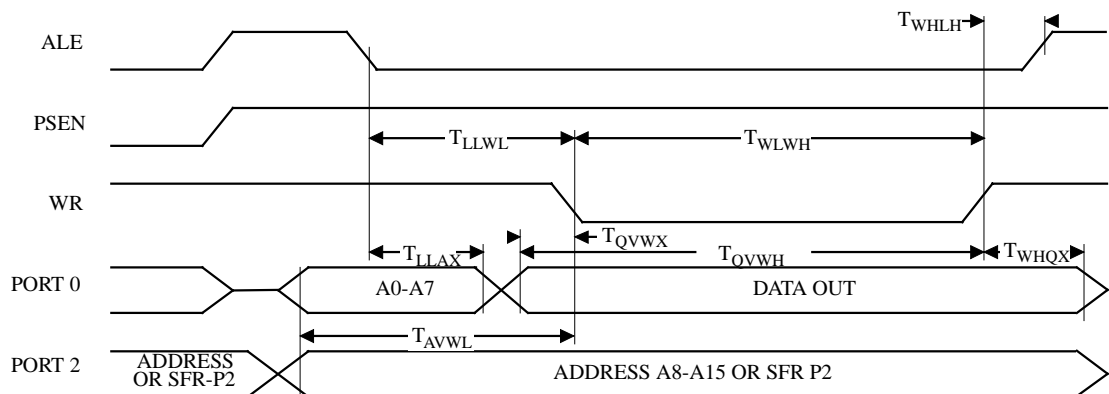


Figure 19. External Data Memory Write Cycle

10.5.6 External Data Memory Read Cycle

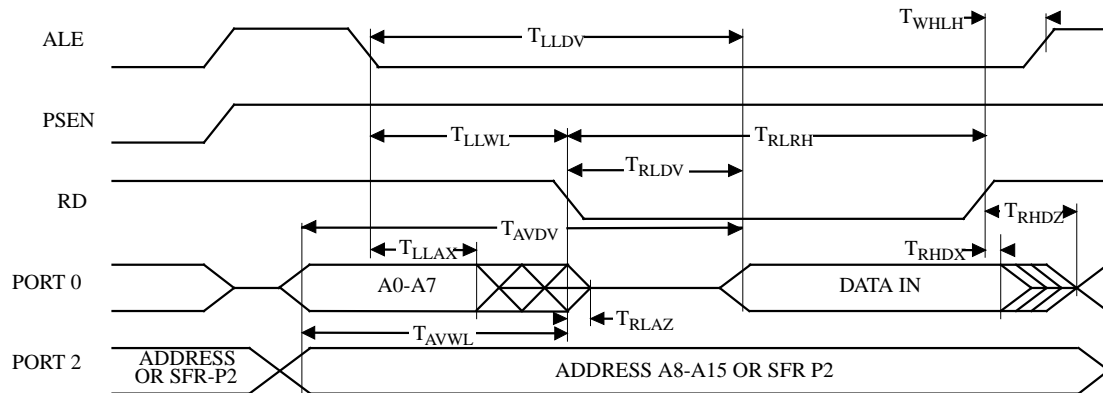


Figure 20. External Data Memory Read Cycle

10.5.7 Serial Port Timing - Shift Register Mode

Table 32. Symbol Description

| Symbol | Parameter |
|-------------|--|
| T_{XLXL} | Serial port clock cycle time |
| T_{QVHX} | Output data set-up to clock rising edge |
| T_{XHQX} | Output data hold after clock rising edge |
| T_{XHDX} | Input data hold after clock rising edge |
| T_{XHDTV} | Clock rising edge to input data valid |

Table 33. AC Parameters for a Fix Clock

| Speed | -M 40 MHz | | -V X2 mode 30 MHz 60 MHz equiv. | | -V standard mode 40 MHz | | -L X2 mode 20 MHz 40 MHz equiv. | | -L standard mode 30 MHz | | Units |
|-------------|--------------|-----|--|-----|-------------------------------|-----|--|-----|-------------------------------|-----|-------|
| | Min | Max | Min | Max | Min | Max | Min | Max | Min | Max | |
| T_{XLXL} | 300 | | 200 | | 300 | | 300 | | 400 | | ns |
| T_{QVHX} | 200 | | 117 | | 200 | | 200 | | 283 | | ns |
| T_{XHQX} | 30 | | 13 | | 30 | | 30 | | 47 | | ns |
| T_{XHDX} | 0 | | 0 | | 0 | | 0 | | 0 | | ns |
| T_{XHDTV} | | 117 | | 34 | | 117 | | 117 | | 200 | ns |

Table 34. AC Parameters for a Variable Clock: derating formula

| Symbol | Type | Standard Clock | X2 Clock | -M | -V | -L | Units |
|------------|------|----------------|----------|-----|-----|-----|-------|
| T_{XLXL} | Min | 12 T | 6 T | | | | ns |
| T_{QVXH} | Min | 10 T - x | 5 T - x | 50 | 50 | 50 | ns |
| T_{XHQX} | Min | 2 T - x | T - x | 20 | 20 | 20 | ns |
| T_{XHDX} | Min | x | x | 0 | 0 | 0 | ns |
| T_{XHDV} | Max | 10 T - x | 5 T - x | 133 | 133 | 133 | ns |

10.5.8 Shift Register Timing Waveforms

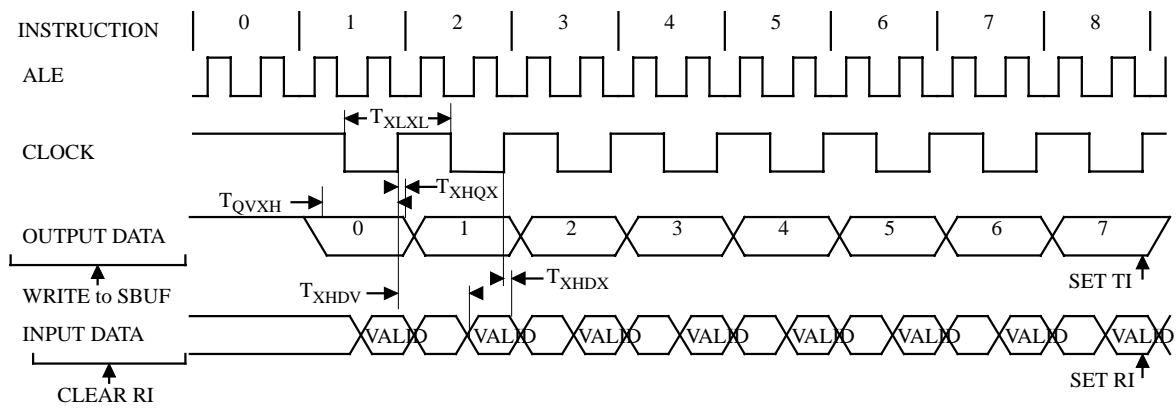


Figure 21. Shift Register Timing Waveforms

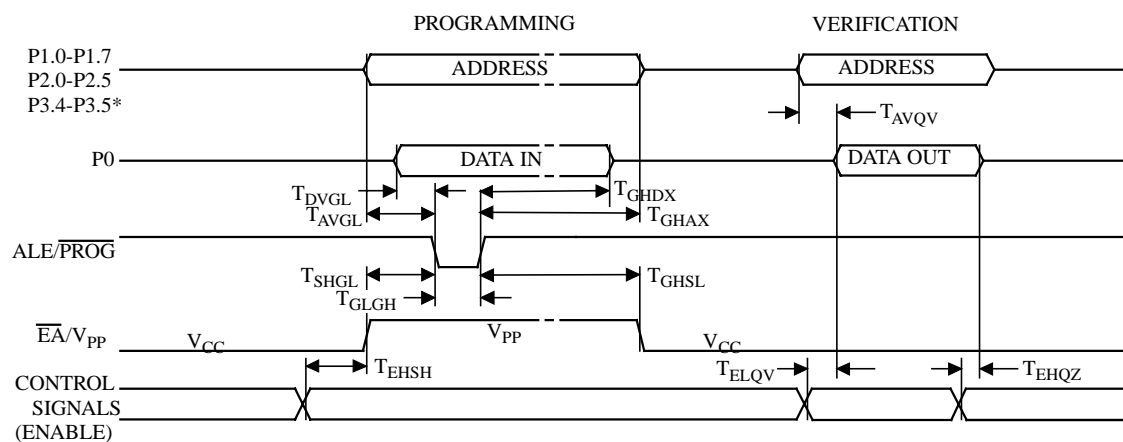
10.5.9 EPROM Programming and Verification Characteristics

$T_A = 21^\circ\text{C}$ to 27°C ; $V_{SS} = 0\text{V}$; $V_{CC} = 5\text{V} \pm 10\%$ while programming. V_{CC} = operating range while verifying.

Table 35. EPROM Programming Parameters

| Symbol | Parameter | Min | Max | Units |
|--------------|--|---------------|---------------|---------------|
| V_{PP} | Programming Supply Voltage | 12.5 | 13 | V |
| I_{PP} | Programming Supply Current | | 75 | mA |
| $1/T_{CLCL}$ | Oscillator Frquency | 4 | 6 | MHz |
| T_{AVGL} | Address Setup to $\overline{\text{PROG}}$ Low | $48 T_{CLCL}$ | | |
| T_{GHAX} | Adress Hold after $\overline{\text{PROG}}$ | $48 T_{CLCL}$ | | |
| T_{DVGL} | Data Setup to $\overline{\text{PROG}}$ Low | $48 T_{CLCL}$ | | |
| T_{GHDX} | Data Hold after $\overline{\text{PROG}}$ | $48 T_{CLCL}$ | | |
| T_{EHS} | (Enable) High to V_{PP} | $48 T_{CLCL}$ | | |
| T_{SHGL} | V_{PP} Setup to $\overline{\text{PROG}}$ Low | 10 | | μs |
| T_{GHSL} | V_{PP} Hold after $\overline{\text{PROG}}$ | 10 | | μs |
| T_{GLGH} | $\overline{\text{PROG}}$ Width | 90 | 110 | μs |
| T_{AVQV} | Address to Valid Data | | $48 T_{CLCL}$ | |
| T_{ELQV} | ENABLE Low to Data Valid | | $48 T_{CLCL}$ | |
| T_{EHQZ} | Data Float after ENABLE | 0 | $48 T_{CLCL}$ | |

10.5.10 EPROM Programming and Verification Waveforms



* 8KB: up to P2.4, 16KB: up to P2.5, 32KB: up to P3.4, 64KB: up to P3.5

Figure 22. EPROM Programming and Verification Waveforms

10.5.11 External Clock Drive Characteristics (XTAL1)

Table 36. AC Parameters

| Symbol | Parameter | Min | Max | Units |
|---------------------|-------------------------|-----|-----|-------|
| T_{CLCL} | Oscillator Period | 25 | | ns |
| T_{CHCX} | High Time | 5 | | ns |
| T_{CLCX} | Low Time | 5 | | ns |
| T_{CLCH} | Rise Time | | 5 | ns |
| T_{CHCL} | Fall Time | | 5 | ns |
| T_{CHCX}/T_{CLCX} | Cyclic ratio in X2 mode | 40 | 60 | % |

10.5.12 External Clock Drive Waveforms

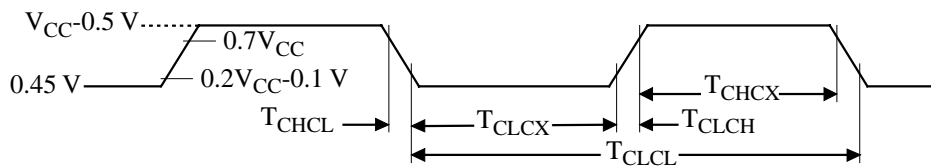


Figure 23. External Clock Drive Waveforms

10.5.13 AC Testing Input/Output Waveforms

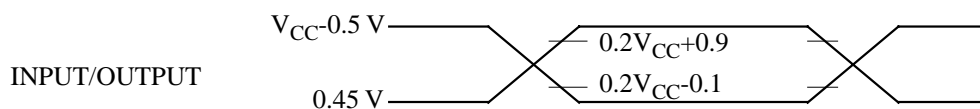


Figure 24. AC Testing Input/Output Waveforms

AC inputs during testing are driven at $V_{CC} - 0.5$ for a logic “1” and 0.45V for a logic “0”. Timing measurement are made at V_{IH} min for a logic “1” and V_{IL} max for a logic “0”.

10.5.14 Float Waveforms

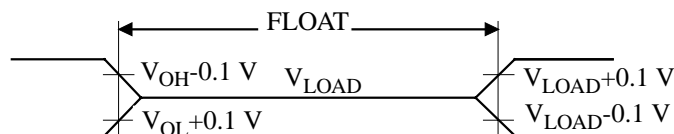


Figure 25. Float Waveforms

For timing purposes a port pin is no longer floating when a 100 mV change from load voltage occurs and begins to float when a 100 mV change from the loaded V_{OH}/V_{OL} level occurs. $I_{OL}/I_{OH} \geq \pm 20\text{mA}$.

10.5.15 Clock Waveforms

Valid in normal clock mode. In X2 mode XTAL2 signal must be changed to XTAL2 divided by two.

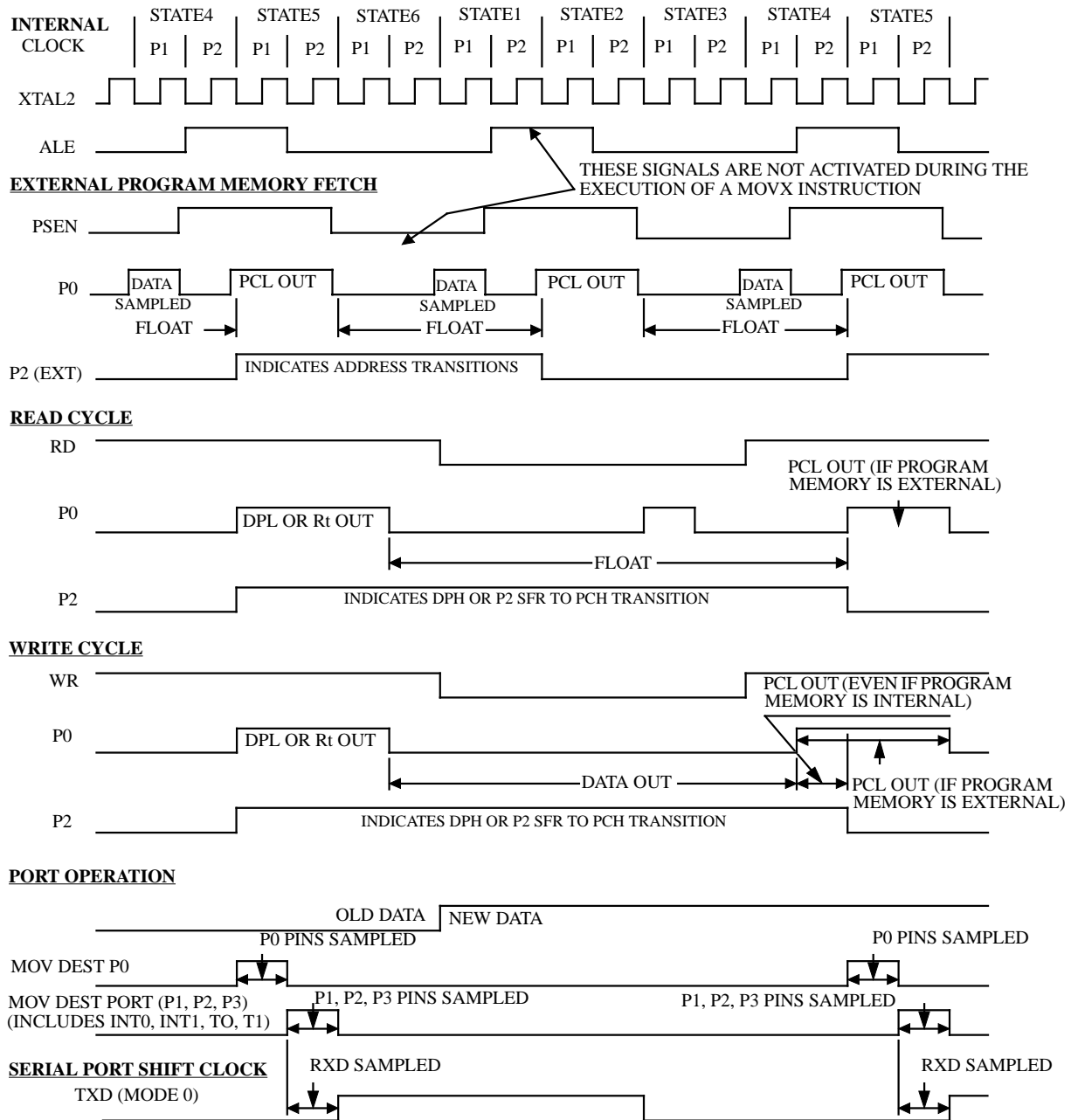
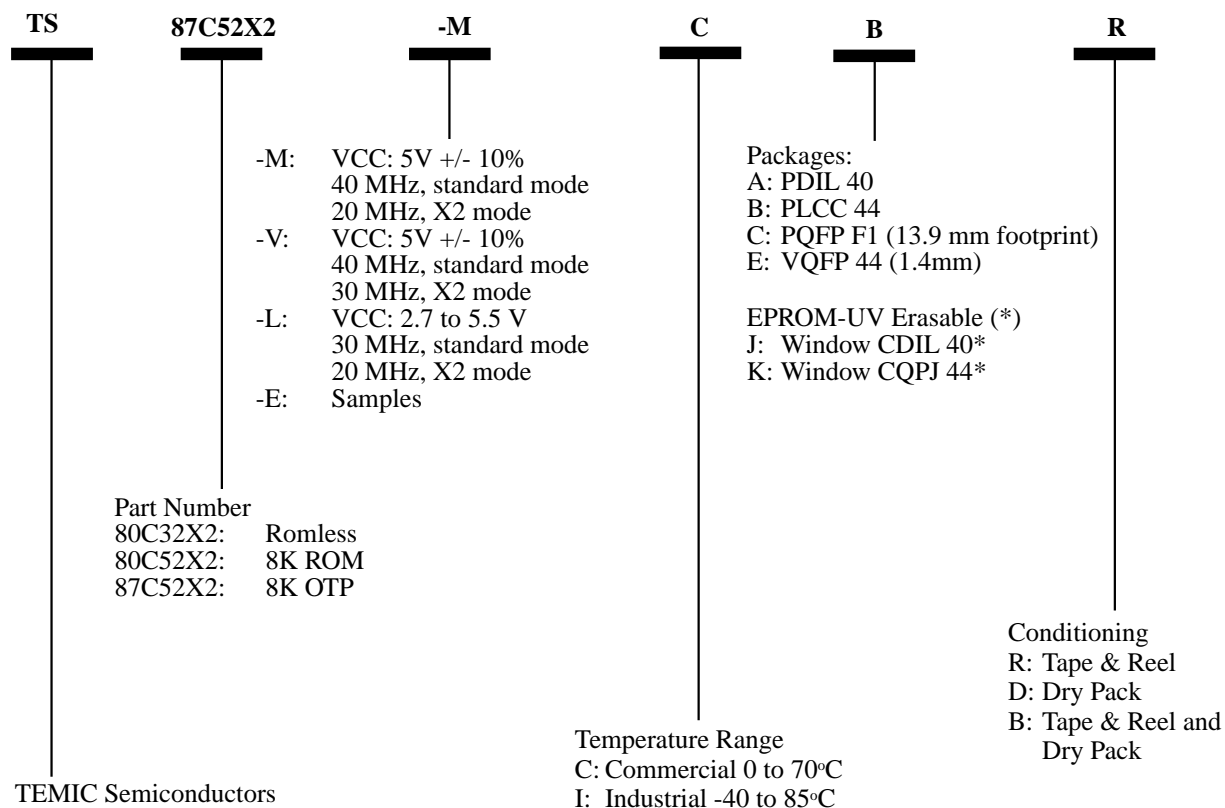


Figure 26. Clock Waveforms

This diagram indicates when signals are clocked internally. The time it takes the signals to propagate to the pins, however, ranges from 25 to 125 ns. This propagation delay is dependent on variables such as temperature and pin loading. Propagation also varies from output to output and component. Typically though ($T_A=25^\circ\text{C}$ fully loaded) \overline{RD} and \overline{WR} propagation delays are approximately 50ns. The other signals are typically 85 ns. Propagation delays are incorporated in the AC specifications.

11. Ordering Information



(*) Check with TEMIC Sales Office for availability. Ceramic packages (J, K) are available for prototyping, not for volume production. Ceramic packages are available for OTP only.

Table 37. Maximum Clock Frequency

| Code | -M | -V | -L | Unit |
|--|----|-----------|-----------|------|
| Standard Mode, oscillator frequency | 40 | 40 | 30 | MHz |
| Standard Mode, internal frequency | 40 | 40 | 30 | |
| X2 Mode, oscillator frequency | 20 | 30 | 20 | MHz |
| X2 Mode, internal equivalent frequency | 40 | 60 | 40 | |

Table 38. Possible Ordering Entries

| | TS80C32X2 | TS80C52zzz ROM | TS87C52 OTP |
|------|-----------|----------------|-------------|
| -MCA | X | X | X |
| -MCB | X | X | X |
| -MCC | X | X | X |
| -MCE | X | X | X |
| -VCA | X | X | X |
| -VCB | X | X | X |
| -VCC | X | X | X |
| -VCE | X | X | X |
| -LCA | X | X | X |
| -LCB | X | X | X |
| -LCC | X | X | X |
| -LCE | X | X | X |
| -MIA | X | X | X |
| -MIB | X | X | X |
| -MIC | X | X | X |
| -MIE | X | X | X |
| -VIA | X | X | X |
| -VIB | X | X | X |
| -VIC | X | X | X |
| -VIE | X | X | X |
| -LIA | X | X | X |
| -LIB | X | X | X |
| -LIC | X | X | X |
| -LIE | X | X | X |
| -EA | X | | X |
| -EB | X | | X |
| -EC | X | | X |
| -EE | X | | X |
| -EJ | | | X |
| -EK | | | X |

- -Ex for samples
- Tape and Reel available for B, C and E packages
- Dry pack mandatory for E packages