

## Features

- **80C52 Compatible**
  - Four 8-bit I/O Ports
  - Three 16-bit Timer/Counters
  - 256 Bytes Scratch Pad RAM
  - 8 Interrupt Sources with 4 Priority Levels
  - Dual Data Pointer
- Variable Length MOVX for Slow RAM/Peripherals
- High-speed Architecture
  - 10 to 40 MHz in Standard Mode
- 16K/32K Bytes On-Chip ROM Program
- T80C51RD2 ROMless Versions
- On-Chip 1024 bytes Expanded RAM (XRAM)
  - Software Selectable Size (0, 256, 512, 768, 1024 bytes)
  - 256 Bytes Selected at Reset for AT87C51RB2/RC2 Compatibility
- Keyboard Interrupt Interface on Port P1
- 8-bit Clock Prescaler
- 64K Program and Data Memory Spaces
- Improved X2 Mode with Independent Selection for CPU and Each Peripheral
- Programmable Counter Array 5 Channels with:
  - High-speed Output
  - Compare/Capture
  - Pulse Width Modulator
  - Watchdog Timer Capabilities
- Asynchronous Port Reset
- Full Duplex Enhanced UART
- Dedicated Baud Rate Generator for UART
- Low EMI (Inhibit ALE)
- Hardware Watchdog Timer (One-time Enabled with Reset-out)
- Power Control Modes
  - Idle Mode
  - Power-down Mode
  - Power-off Flag
- Power Supply: 2.7V to 5.5V or 2.7V to 3.6V
- Temperature Ranges: Commercial (0 to +70°C) and Industrial (-40°C to +85°C)
- Packages: PDIL40, PLCC44, VQFP44

## Description

AT8xC51Rx2 microcontrollers are high performance ROM versions of the 80C51 8-bit microcontrollers. They contain a 0K, 16K or 32K bytes ROM memory block for program.

The microcontrollers retain all features of the Atmel 80C52 with 256 bytes of internal RAM, a 7-source 4-level interrupt controller and three timer/counters.

In addition, the microcontrollers have a Programmable Counter Array, an XRAM of 1024 byte, a Hardware Watchdog Timer, a Keyboard Interface, a more versatile serial channel that facilitates multiprocessor communication (EUART) and a speed improvement mechanism (X2 mode).

The microcontrollers have 2 software-selectable modes of reduced activity and 8 bit clock prescaler for further reduction in power consumption. In Idle mode, the CPU is frozen while the peripherals and the interrupt system are still operating. In the Power-down mode, the RAM is saved and all other functions are inoperative.



**80C51 High  
Performance  
ROM 8-bit  
Microcontroller**

**AT80C51RD2  
AT83C51RB2  
AT83C51RC2**

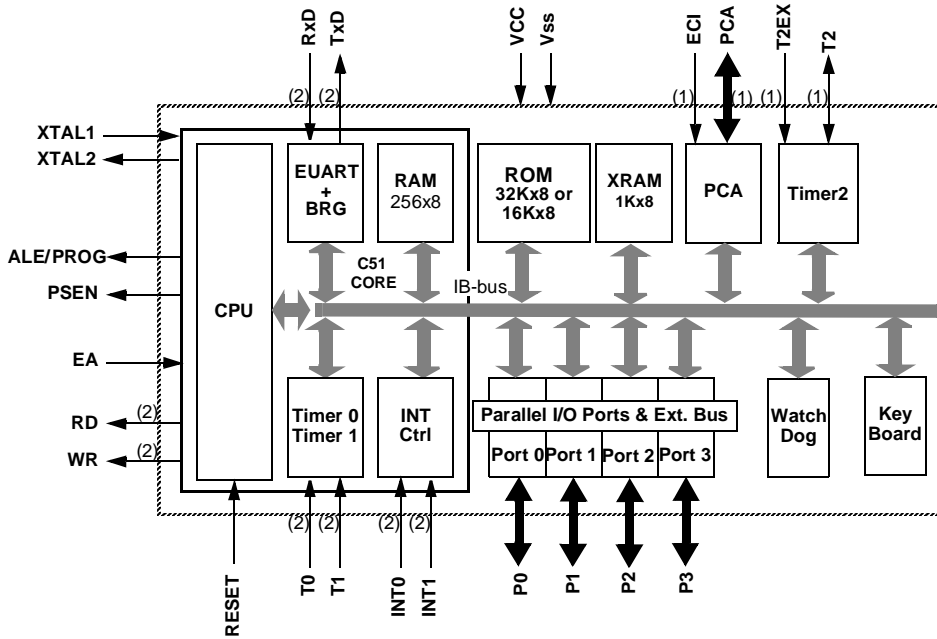
Rev. 4113A-8051-09/02



**Table 1. Memory Size**

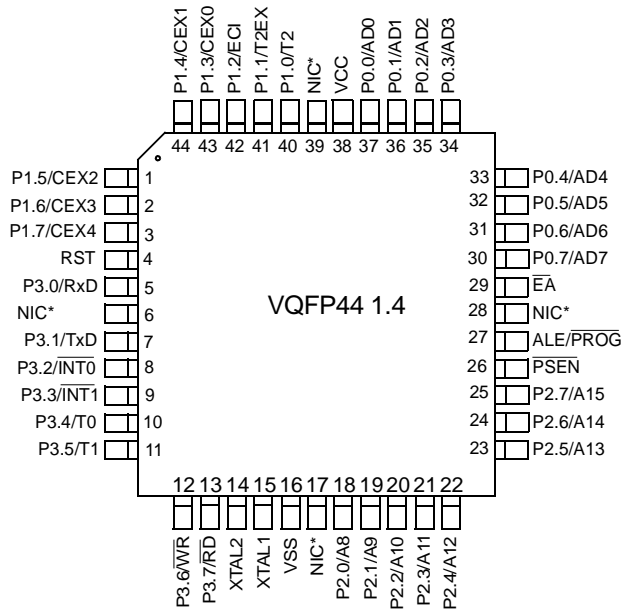
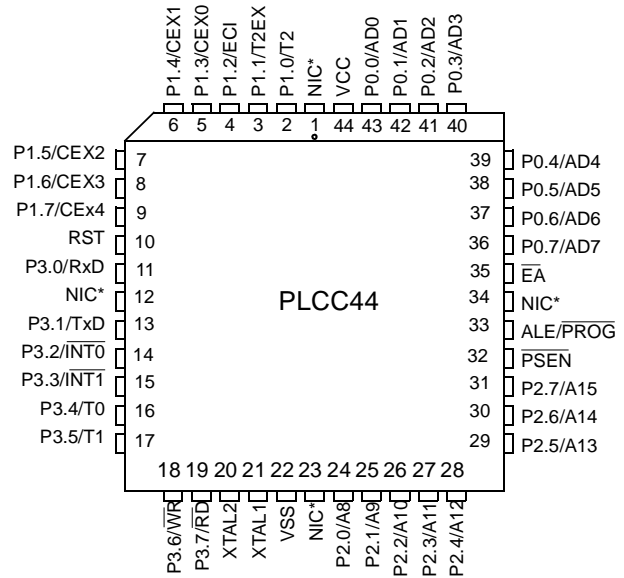
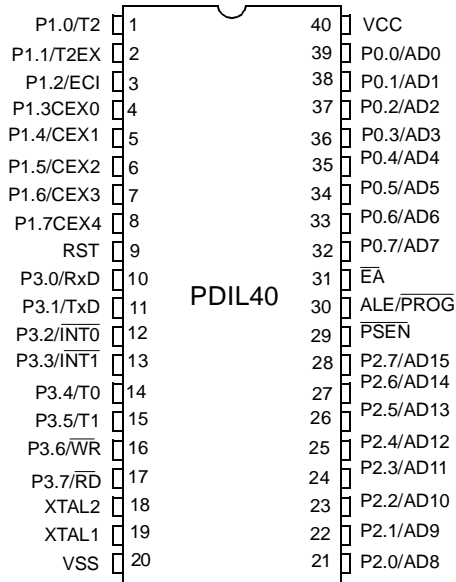
	ROM (Bytes)	XRAM (Bytes)	TOTAL RAM (Bytes)	I/O
AT83C51RB2	16K	1024	1280	32
AT83C51RC2	32K	1024	1280	32
AT80C51RD2	ROMless	1024	1280	32

## Block Diagram



- Notes: 1. Alternate function of Port 1  
2. Alternate function of Port 3

## Pin Configurations



\*NIC: No Internal Connection

**Table 2. Pin Description**

Mnemonic	Pin Number			Type	Name and Function
	DIL	PLCC44	VQFP44 1.4		
V <sub>SS</sub>	20	22	16	I	<b>Ground:</b> 0V reference
V <sub>CC</sub>	40	44	38	I	<b>Power Supply:</b> 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	<b>Port 0:</b> Port 0 is an open-drain, bi-directional I/O port. Port 0 pins that have 1s written to them float and can be used as high impedance inputs. Port 0 must be polarized to V <sub>CC</sub> or V <sub>SS</sub> 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	<b>Port 1:</b> Port 1 is an 8-bit bi-directional 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 T89C51RB2/RC2 Port 1 include:
	1	2	40	I/O	<b>P1.0:</b> Input/Output
				I/O	<b>T2 (P1.0):</b> Timer/Counter 2 external count input/Clockout
	2	3	41	I/O	<b>P1.1:</b> Input/Output
				I	<b>T2EX:</b> Timer/Counter 2 Reload/Capture/Direction Control
	3	4	42	I/O	<b>P1.2:</b> Input/Output
				I	<b>ECl:</b> External Clock for the PCA
	4	5	43	I/O	<b>P1.3:</b> Input/Output
				I/O	<b>CEx0:</b> Capture/Compare External I/O for PCA module 0
	5	6	44	I/O	<b>P1.4:</b> Input/Output
				I/O	<b>CEx1:</b> Capture/Compare External I/O for PCA module 1
	6	7	1	I/O	<b>P1.5:</b> Input/Output
				I/O	<b>CEx2:</b> Capture/Compare External I/O for PCA module 2
	7	8	2	I/O	<b>P1.6:</b> Input/Output
				I/O	<b>CEx3:</b> Capture/Compare External I/O for PCA module 3
	8	9	3	I/O	<b>P1.7:</b> Input/Output:
				I/O	<b>CEx4:</b> Capture/Compare External I/O for PCA module 4
XTAL1	19	21	15	I	<b>Crystal 1:</b> Input to the inverting oscillator amplifier and input to the internal clock generator circuits.
XTAL2	18	20	14	O	<b>Crystal 2:</b> Output from the inverting oscillator amplifier

**Table 2.** Pin Description (Continued)

Mnemonic	Pin Number			Type	Name and Function
	DIL	PLCC44	VQFP44 1.4		
P2.0 - P2.7	21 - 28	24 - 31	18 - 25	I/O	<p><b>Port 2:</b> Port 2 is an 8-bit bi-directional 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 ROM reading and verification:</p> <p>P2.0 to P2.5 for 16 KB devices P2.0 to P2.6 for 32 KB devices</p>
P3.0 - P3.7	10 - 17	11, 13 - 19	5, 7 - 13	I/O	<p><b>Port 3:</b> Port 3 is an 8-bit bi-directional 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.</p>
	10	11	5	I	<b>RXD (P3.0):</b> Serial input port
	11	13	7	O	<b>TXD (P3.1):</b> Serial output port
	12	14	8	I	<b>INT0 (P3.2):</b> External interrupt 0
	13	15	9	I	<b>INT1 (P3.3):</b> External interrupt 1
	14	16	10	I	<b>T0 (P3.4):</b> Timer 0 external input
	15	17	11	I	<b>T1 (P3.5):</b> Timer 1 external input
	16	18	12	O	<b>WR (P3.6):</b> External data memory write strobe
	17	19	13	O	<b>RD (P3.7):</b> External data memory read strobe
RST	9	10	4	I/O	<b>Reset:</b> A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to V <sub>SS</sub> permits a power-on reset using only an external capacitor to V <sub>CC</sub> . This pin is an output when the hardware watchdog forces a system reset.
ALE/ $\overline{\text{PROG}}$	30	33	27	O (I)	<b>Address Latch Enable/Program Pulse:</b> 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 (PROG) during Flash programming. ALE can be disabled by setting SFR's AUXR.0 bit. With this bit set, ALE will be inactive during internal fetches.
PSEN	29	32	26	O	<b>Program Strobe Enable:</b> 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.
EA	31	35	29	I	<b>External Access Enable:</b> $\overline{\text{EA}}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H to 3FFFH (16K), 7FFFH (32K). If security level 1 is programmed, $\overline{\text{EA}}$ will be internally latched on Reset.

## SFR Mapping

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

- C51 core registers: ACC, B, DPH, DPL, PSW, SP
- 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
- PCA (Programmable Counter Array) registers: CCON, CCAPMx, CL, CH, CCAPxH, CCAPxL (x: 0 to 4)
- Power and clock control registers: PCON
- Hardware Watchdog Timer registers: WDTRST, WDTPRG
- Interrupt system registers: IE0, IPL0, IPH0, IE1, IPL1, IPH1
- Keyboard Interface registers: KBE, KBF, KBLS
- BRG (Baud Rate Generator) registers: BRL, BDRCON
- Clock Prescaler register: CKRL
- Others: AUXR, AUXR1, CKCON0, CKCON1



Table 3 shows all SFRs with their address and their reset value.

**Table 3.** SFR Mapping

	Bit Addressable	Non-bit Addressable							
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
F8h		CH 0000 0000	CCAP0H XXXX XXXX	CCAP1H XXXX XXXX	CCAPL2H XXXX XXXX	CCAPL3H XXXX XXXX	CCAPL4H XXXX XXXX		FFh
F0h	B 0000 0000								F7h
E8h		CL 0000 0000	CCAP0L XXXX XXXX	CCAP1L XXXX XXXX	CCAPL2L XXXX XXXX	CCAPL3L XXXX XXXX	CCAPL4L XXXX XXXX		EFh
E0h	ACC 0000 0000								E7h
D8h	CCON 00X0 0000	CMOD 00XX X000	CCAPM0 X000 0000	CCAPM1 X000 0000	CCAPM2 X000 0000	CCAPM3 X000 0000	CCAPM4 X000 0000		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	IPL0 X000 000	SADEN 0000 0000							BFh
B0h	P3 1111 1111	IE1 XXXX XXX0b	IPL1 XXXX XXX0b	IPH1 XXXX XXX0b				IPH0 X000 0000	B7h
A8h	IE0 0000 0000	SADDR 0000 0000							AFh
A0h	P2 1111 1111		AUXR1 XXXX XXX0				WDTRST XXXX XXXX	WDTPRG XXXX X000	A7h
98h	SCON 0000 0000	SBUF XXXX XXXX	BRL 0000 0000	BDRCON XXX0 0000	KBLS 0000 0000	KBE 0000 0000	KBF 0000 0000		9Fh
90h	P1 1111 1111							CKRL 1111 1111	97h
88h	TCON 0000 0000	TMOD 0000 0000	TL0 0000 0000	TL1 0000 0000	TH0 0000 0000	TH1 0000 0000	AUXR XX0X 0000	CKCON0 0000 0000	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

## Oscillators

### Overview

One oscillator is available for CPU:

- OSC used for high frequency (3 MHz to 40 MHz)

In order to optimize the power consumption and the execution time needed for a specific task, an internal prescaler feature has been implemented between the selected oscillator and the CPU.

### Registers

**Table 4.** Clock Reload Register

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-
Bit Number	Bit Mnemonic	Description					
7:0	CKRL	Clock Reload Register: Prescaler value					

Reset Value = 1111 1111b

Not bit addressable

### Prescaler Divider

A hardware RESET puts the prescaler divider in the following state:

- CKRL = FFh:  $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSC}/2$  (Standard C51 feature)  
KS signal selects OSC:  $F_{CLK\ OUT} = F_{OSC}$
- Any value between FFh down to 00h can be written by software into CKRL register in order to divide frequency of the selected oscillator:
  - CKRL = 00h: minimum frequency  
 $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSC}/1020$  (Standard Mode)  
 $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSC}/510$  (X2 Mode)
  - CKRL = FFh: maximum frequency  
 $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSC}/2$  (Standard Mode)  
 $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSC}$  (X2 Mode)
  - $F_{CLK\ CPU}$  and  $F_{CLK\ PERIPH}$   
 In X2 mode:

$$F_{CPU} = F_{CLKPERIPH} = \frac{F_{OSC}}{2 \times (255 - CKRL)}$$

$$\text{In X1 mode: } F_{CPU} = F_{CLKPERIPH} = \frac{F_{OSCA}}{4 \times (255 - CKRL)}$$



## Enhanced Features

In comparison to the original 80C52, the microcontrollers implement the following new features:

- X2 option
- Dual Data Pointer
- Extended RAM
- Programmable Counter Array (PCA)
- Hardware Watchdog
- 4-level Interrupt Priority System
- Power-off Flag
- Power On Reset
- ONCE mode
- ALE disabling
- Some enhanced features are also located in the UART and the Timer 2

## X2 Feature and OSC Clock Generation

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

- Divides frequency crystals by 2 (cheaper crystals) while keeping same CPU power.
- Saves power consumption while keeping same CPU power (oscillator power saving).
- Saves power consumption by dividing dynamically the operating frequency by 2 in operating and idle modes.
- Increases 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.

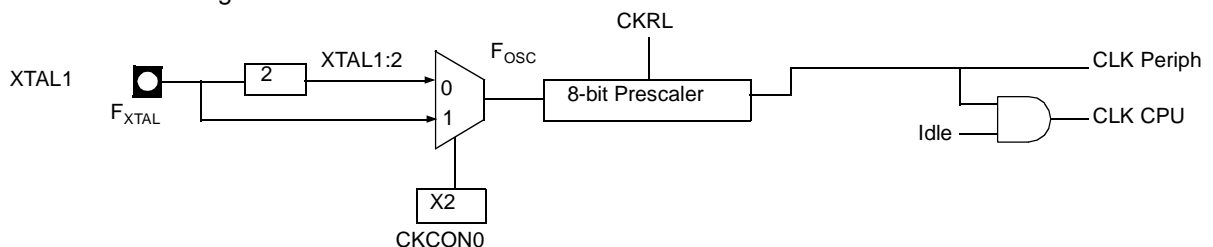
### Description

The clock for the whole circuit and peripherals is first divided by two before being used by the CPU core and the peripherals.

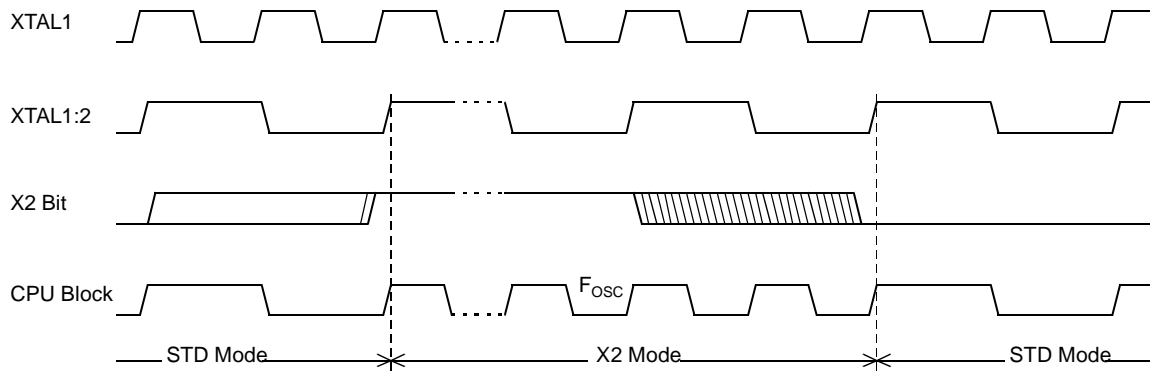
This allows any cyclic ratio to be accepted on XTAL1 input. In X2 mode, as this divider is bypassed, the signals on XTAL1 must have a cyclic ratio between 40 to 60%.

Figure 2 shows the clock generation block diagram. X2 bit is validated on the rising edge of the  $XTAL1 \div 2$  to avoid glitches when switching from X2 to standard mode. Figure 3 shows the switching mode waveforms.

**Figure 2.** Clock Generation Diagram



**Figure 3. Mode Switching Waveforms**



The X2 bit in the CKCON0 register (see Table 5) allows to switch from 12 clock periods per instruction to 6 clock periods and vice versa. At reset, the speed is set according to X2 bit of Hardware Config Byte (HCB). By default, Standard mode is activated. Setting the X2 bit activates the X2 feature (X2 mode).

The T0X2, T1X2, T2X2, UARTX2, PCAX2 and WDX2 bits in the CKCON0 register (Table 5) allow to switch from standard peripheral speed (12 clock periods per peripheral clock cycle) to fast peripheral speed (6 clock periods per peripheral clock cycle). These bits are active only in X2 mode.

**Table 5. CKCON0 Register**

CKCON0 - Clock Control Register (8Fh)

7	6	5	4	3	2	1	0
-	WDX2	PCAX2	SIX2	T2X2	T1X2	T0X2	X2
Bit Number	Bit Mnemonic	Description					
7	-	<b>Reserved</b> Do not set this bit.					
6	WDX2	<b>Watchdog clock</b> (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect). Cleared to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.					
5	PCAX2	<b>Programmable Counter Array clock</b> (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect). Cleared to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.					
4	SIX2	<b>Enhanced UART clock (Mode 0 and 2)</b> (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect). Cleared to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.					
3	T2X2	<b>Timer 2 clock</b> (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect). Cleared to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.					
2	T1X2	<b>Timer 1 clock</b> (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect). Cleared to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.					
1	T0X2	<b>Timer 0 clock</b> (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect). Cleared to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.					
0	X2	<b>CPU clock</b> Cleared to select 12 clock periods per machine cycle (STD mode) for CPU and all the peripherals. Set to select 6 clock periods per machine cycle (X2 mode) and to enable the individual peripherals "X2" bits. Programmed by hardware after Power-up regarding Hardware Config Byte (HCB).					

Reset Value = 0000 000'HCB.X2'b (see Hardware Config Byte)

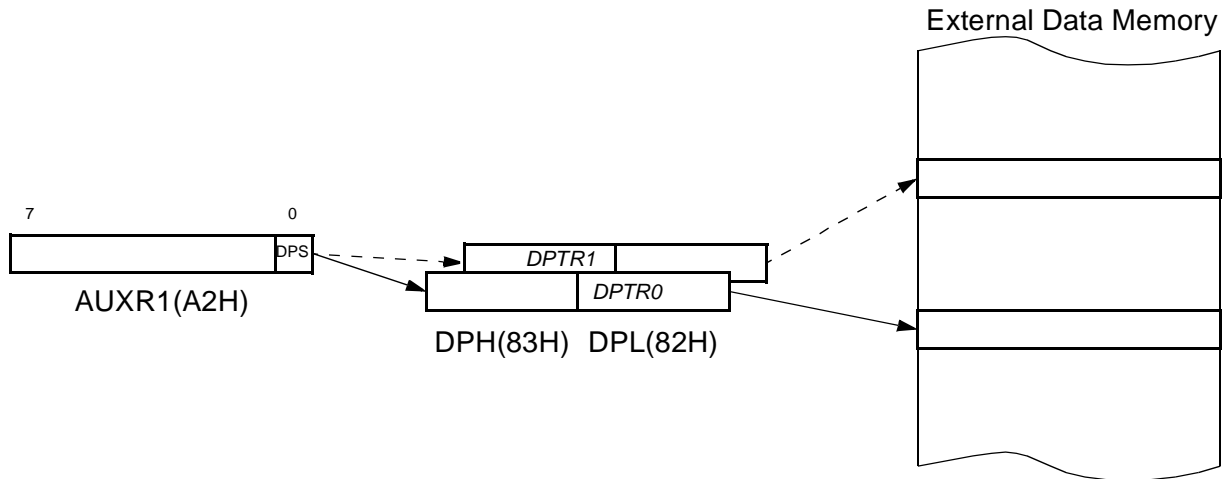
Not bit addressable

## Dual Data Pointer Register

The additional data pointer can be used to speed up code execution and reduce code size.

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 = AUXR1.0 (see Table 6) that allows the program code to switch between them (Refer to Figure 4).

**Figure 4.** Use of Dual Pointer



**Table 6.** AUXR1 Register

AUXR1- Auxiliary Register 1(0A2h)

7	6	5	4	3	2	1	0
-	-	-	-	GF3	0	-	DPS
Bit Number	Bit Mnemonic	Description					
7	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
6	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
5	-	Reserved					
4	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
3	GF3	This bit is a general purpose user flag.					
2	0	Always cleared <sup>(1)</sup> .					
1	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
0	DPS	<b>Data Pointer Selection</b> Cleared to select DPTR0. Set to select DPTR1.					

Reset Value: XXXX XXXX0b

Not bit addressable

Note: 1. Bit 2 stuck at 0; this allows to use INC AUXR1 to toggle DPS without changing GF3.

## Assembly Language

```

; Block move using dual data pointers
; Modifies DPTR0, DPTR1, A and PSW
; note: DPS exits opposite of entry state
; unless an extra INC AUXR1 is added
;
00A2  AUXR1 EQU 0A2H
;
0000 909000MOV 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 70F6JNZ 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.

## Expanded RAM (XRAM)

The T8xc51Rx2 devices provide additional Bytes of Random Access Memory (RAM) space for increased data parameter handling and high level language usage.

The devices have expanded RAM in external data space; maximum size and location are described in Table 7.

**Table 7.** Expanded RAM

	XRAM size	Address	
		Start	End
T83C51RB2/RC2 T80C51RD2	1024	00h	3FFh

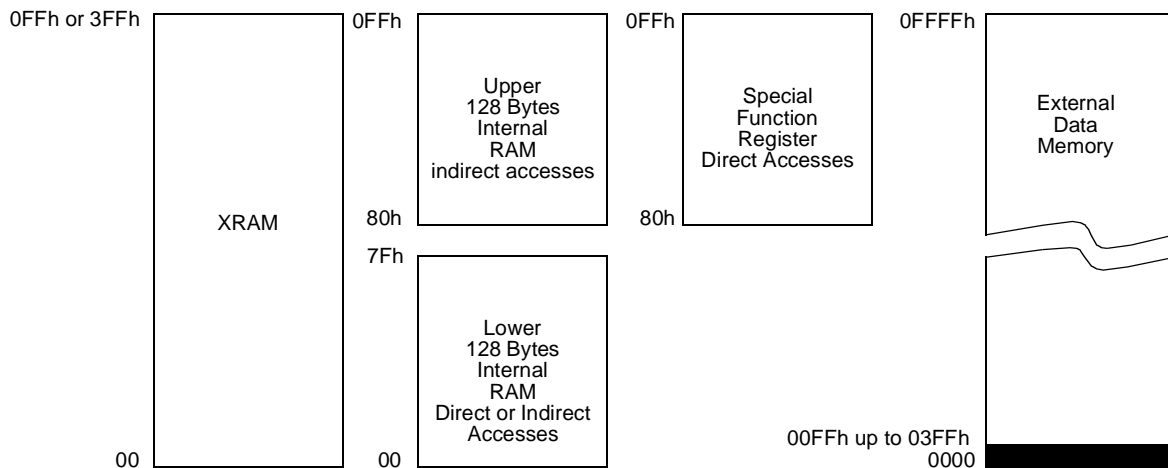
The T8xc51Rx2 has internal data memory that is mapped into four separate segments.

The four segments are:

1. The Lower 128 bytes of RAM (addresses 00h to 7Fh) are directly and indirectly addressable.
2. The Upper 128 bytes of RAM (addresses 80h to FFh) are indirectly addressable only.
3. The Special Function Registers (SFRs) (addresses 80h to FFh) are directly addressable only.
4. The expanded RAM bytes are indirectly accessed by MOVX instructions, and with the EXTRAM bit cleared in the AUXR register (see Table 7).

The lower 128 bytes can be accessed by either direct or indirect addressing. The Upper 128 bytes can be accessed by indirect addressing only. The Upper 128 bytes occupy the same address space as the SFR. That means they have the same address, but are physically separate from SFR space.

**Figure 5.** Internal and External Data Memory Address



When an instruction accesses an internal location above address 7Fh, the CPU knows whether the access is to the upper 128 bytes of data RAM or to SFR space by the addressing mode used in the instruction.

- Instructions that use direct addressing access SFR space. For example: MOV 0A0H, # data, accesses the SFR at location 0A0h (which is P2).
- Instructions that use indirect addressing access the Upper 128 bytes of data RAM. For example: MOV @R0, # data where R0 contains 0A0h, accesses the data byte at address 0A0h, rather than P2 (whose address is 0A0h).
- The XRAM bytes can be accessed by indirect addressing, with EXTRAM bit cleared and MOVX instructions. This part of memory which is physically located on-chip, logically occupies the first bytes of external data memory. The bits XRS0 and XRS1 are used to hide a part of the available XRAM as explained in Table 7. This can be useful if external peripherals are mapped at addresses already used by the internal XRAM.
- With EXTRAM = 0, the XRAM is indirectly addressed, using the MOVX instruction in combination with any of the registers R0, R1 of the selected bank or DPTR. An access to XRAM will not affect ports P0, P2, P3.6 (WR) and P3.7 (RD). For example, with EXTRAM = 0, MOVX @R0, # data where R0 contains 0A0H, accesses the XRAM at address 0A0H rather than external memory. An access to external data memory locations higher than the accessible size of the XRAM will be performed with the MOVX DPTR instructions in the same way as in the standard 80C51, with P0 and P2 as data/address busses, and P3.6 and P3.7 as write and read timing signals. Accesses to XRAM above 0FFH can only be done by the use of DPTR.
- With EXTRAM = 1, MOVX @Ri and MOVX @DPTR will be similar to the standard 80C51. MOVX @ Ri will provide an eight-bit address multiplexed with data on Port 0 and any output port pins can be used to output higher order address bits. This is to provide the external paging capability. MOVX @DPTR will generate a sixteen-bit address. Port2 outputs the high-order eight address bits (the contents of DPH) while Port0 multiplexes the low-order eight address bits (DPL) with data. MOVX @ Ri and MOVX @DPTR will generate either read or write signals on P3.6 ( $\overline{WR}$ ) and P3.7 ( $\overline{RD}$ ).

The stack pointer (SP) may be located anywhere in the 256 bytes RAM (lower and upper RAM) internal data memory. The stack may not be located in the XRAM.

The M0 bit allows to stretch the XRAM timings; if M0 is set, the read and write pulses are extended from 6 to 30 clock periods. This is useful to access external slow peripherals.

**Table 8. AUXR Register**

AUXR - Auxiliary Register (8Eh)

	7	6	5	4	3	2	1	0
	-	-	M0	-	XRS1	XRS0	EXTRAM	AO

Bit Number	Bit Mnemonic	Description															
7	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit															
6	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit															
5	M0	<b>Pulse length</b> Cleared to stretch MOVX control: the $\overline{RD}$ and the $\overline{WR}$ pulse length is 6 clock periods (default). Set to stretch MOVX control: the $\overline{RD}$ and the $\overline{WR}$ pulse length is 30 clock periods.															
4	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit															
3	XRS1	<b>XRAM Size</b> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;"><u>XRS1</u></th> <th style="text-align: left;"><u>XRS0</u></th> <th style="text-align: left;"><u>XRAM Size</u></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>256 bytes (default)</td> </tr> <tr> <td>0</td> <td>1</td> <td>512 bytes</td> </tr> <tr> <td>1</td> <td>0</td> <td>768 bytes</td> </tr> <tr> <td>1</td> <td>1</td> <td>1024 bytes</td> </tr> </tbody> </table>	<u>XRS1</u>	<u>XRS0</u>	<u>XRAM Size</u>	0	0	256 bytes (default)	0	1	512 bytes	1	0	768 bytes	1	1	1024 bytes
<u>XRS1</u>	<u>XRS0</u>		<u>XRAM Size</u>														
0	0		256 bytes (default)														
0	1		512 bytes														
1	0	768 bytes															
1	1	1024 bytes															
2	XRS0																
1	EXTRAM	<b>EXTRAM bit</b> Cleared to access internal XRAM using MOVX @ Ri/ @ DPTR. Set to access external memory. Programmed by hardware after Power-up regarding Hardware Security Byte (HSB), default setting, XRAM selected.															
0	AO	<b>ALE Output bit</b> Cleared, ALE is emitted at a constant rate of 1/6 the oscillator frequency (or 1/3 if X2 mode is used) (default). Set, ALE is active only if a MOVX or MOVC instruction is used.															

Reset Value = XX0X 00'HSB.XRAM'0b (see Table 7)

Not bit addressable



## Timer 2

The Timer 2 in the T8xc51Rx2 is the standard C52 Timer 2.

It is a 16-bit timer/counter: the count is maintained by two eight-bit timer registers, TH2 and TL2 are cascaded. It is controlled by T2CON (Table 9) and T2MOD (Table 10) registers. 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, auto-reload and Baud Rate Generator. These modes are selected by the combination of RCLK, TCLK and CP/RL2 (T2CON).

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

Timer 2 includes the following enhancements:

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

## Auto-reload Mode

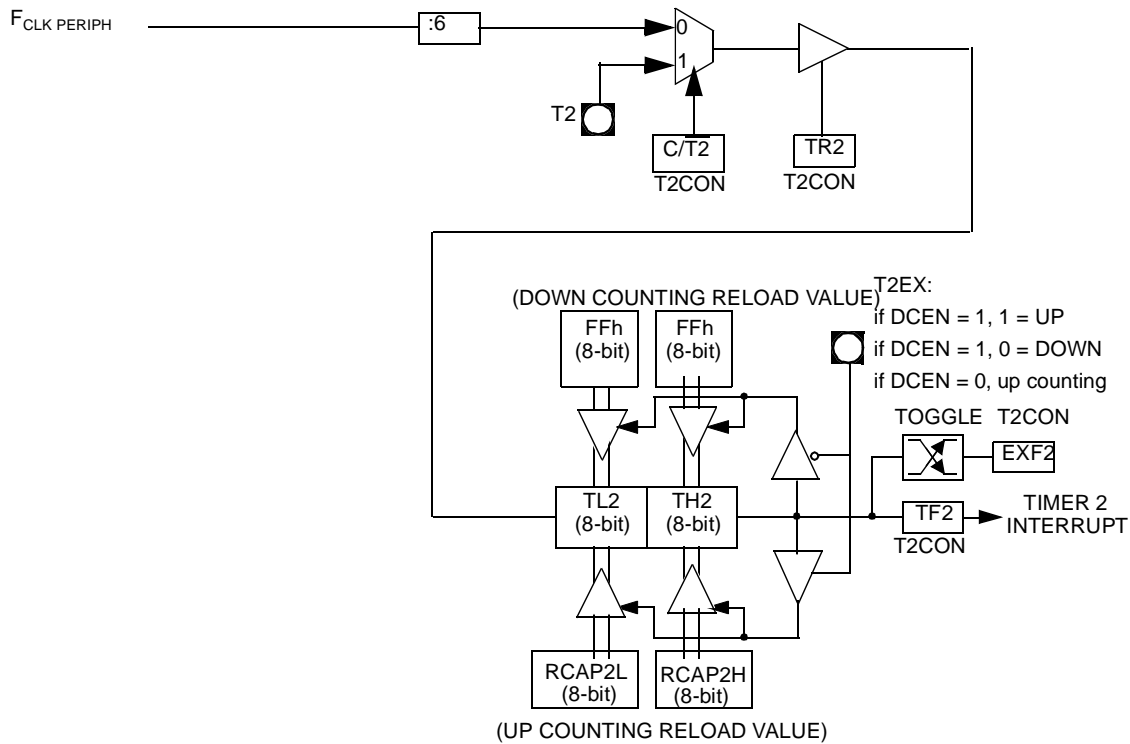
The auto-reload mode configures Timer 2 as a 16-bit timer or event counter with automatic reload. If DCEN bit in T2MOD is cleared, Timer 2 behaves as in 80C52 (refer to the Atmel 8-bit Microcontroller Hardware description). If DCEN bit is set, Timer 2 acts as an Up/down timer/counter as shown in Figure 6. 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 direction of the count. EXF2 does not generate any interrupt. This bit can be used to provide 17-bit resolution.

**Figure 6.** Auto-Reload Mode Up/Down Counter (DCEN = 1)



## Programmable Clock-Output

In the clock-out mode, Timer 2 operates as a 50% duty-cycle, programmable clock generator (see Figure 7). The input clock increments  $TL2$  at frequency  $F_{CLK PERIPH}/2$ . 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:

$$Clock - Out Frequency = \frac{F_{CLK PERIPH}}{4 \times (65536 - RCAP2H / RCAP2L)}$$

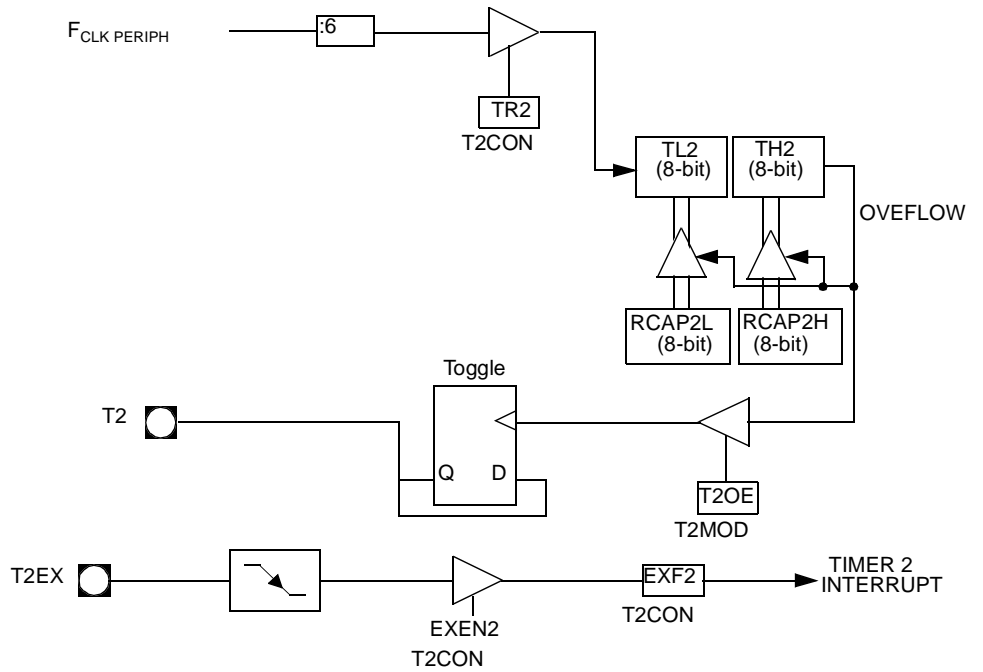
For a 16 MHz system clock, Timer 2 has a programmable frequency range of 61 Hz ( $F_{CLK PERIPH}/2^{16}$ ) to 4 MHz ( $F_{CLK PERIPH}/4$ ). The generated clock signal is brought out to T2 pin (P1.0).

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 7. Clock-Out Mode  $C/\overline{T2} = 07$



**Table 9.** 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	<b>Timer 2 overflow Flag</b> Must be cleared by software. Set by hardware on Timer 2 overflow, if RCLK = 0 and TCLK = 0.					
6	EXF2	<b>Timer 2 External Flag</b> 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	<b>Receive Clock bit</b> Cleared 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	<b>Transmit Clock bit</b> Cleared 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	<b>Timer 2 External Enable bit</b> Cleared 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	<b>Timer 2 Run control bit</b> Cleared to turn off Timer 2. Set to turn on Timer 2.					
1	C/T2#	<b>Timer/Counter 2 select bit</b> Cleared for timer operation (input from internal clock system: $F_{CLK\ PERIPH}$ ). Set for counter operation (input from T2 input pin, falling edge trigger). Must be 0 for clock out mode.					
0	CP/RL2#	<b>Timer 2 Capture/Reload bit</b> If RCLK = 1 or TCLK = 1, CP/RL2# is ignored and timer is forced to auto-reload on Timer 2 overflow. Cleared 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 10.** 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	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
6	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
5	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
4	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
3	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
2	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
1	T2OE	<b>Timer 2 Output Enable bit</b> Cleared to program P1.0/T2 as clock input or I/O port. Set to program P1.0/T2 as clock output.					
0	DCEN	<b>Down Counter Enable bit</b> Cleared to disable Timer 2 as up/down counter. Set to enable Timer 2 as up/down counter.					

Reset Value = XXXX XX00b

Not bit addressable

## Programmable Counter Array (PCA)

The PCA provides more timing capabilities with less CPU intervention than the standard timer/counters. Its advantages include reduced software overhead and improved accuracy. The PCA consists of a dedicated timer/counter which serves as the time base for an array of five compare/capture modules. Its clock input can be programmed to count any one of the following signals:

- Peripheral clock frequency ( $F_{CLK\ PERIPH} \div 6$ )
- Peripheral clock frequency ( $F_{CLK\ PERIPH} \div 2$ )
- Timer 0 overflow
- External input on ECI (P1.2)

Each compare/capture modules can be programmed in any one of the following modes:

- Rising and/or falling edge capture
- Software timer
- High-speed output
- Pulse width modulator

Module 4 can also be programmed as a Watchdog Timer (see Section "PCA Watchdog Timer", page 34).

When the compare/capture modules are programmed in the capture mode, software timer, or high-speed output mode, an interrupt can be generated when the module executes its function. All five modules plus the PCA timer overflow share one interrupt vector.

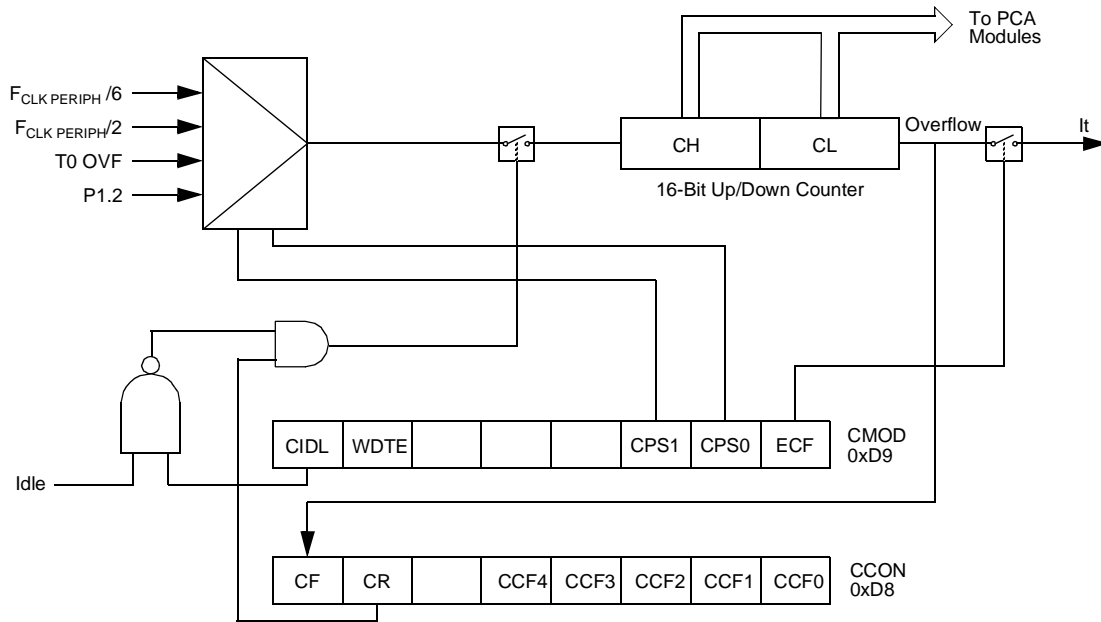
The PCA timer/counter and compare/capture modules share Port 1 for external I/O. These pins are listed below. If the port is not used for the PCA, it can still be used for standard I/O.

PCA Component	External I/O Pin
16-bit Counter	P1.2/ECI
16-bit Module 0	P1.3/CEX0
16-bit Module 1	P1.4/CEX1
16-bit Module 2	P1.5/CEX2
16-bit Module 3	P1.6/CEX3

The PCA timer is a common time base for all five modules (see Figure 8). The timer count source is determined from the CPS1 and CPS0 bits in the CMOD register (Table 11) and can be programmed to run at:

- 1/6 the peripheral clock frequency ( $F_{CLK\ PERIPH}$ )
- 1/2 the peripheral clock frequency ( $F_{CLK\ PERIPH}$ )
- The Timer 0 overflow
- The input on the ECI pin (P1.2)

**Figure 8. PCA Timer/Counter**



**Table 11.** CMOD Register

CMOD - PCA Counter Mode Register (D9h)

7	6	5	4	3	2	1	0
CIDL	WDTE	-	-	-	CPS1	CPS0	ECF
Bit Number	Bit Mnemonic	Description					
7	CIDL	<b>Counter Idle Control</b> Cleared to program the PCA Counter to continue functioning during idle Mode. Set to program PCA to be gated off during idle.					
6	WDTE	<b>Watchdog Timer Enable</b> Cleared to disable Watchdog Timer function on PCA Module 4. Set to enable Watchdog Timer function on PCA Module 4.					
5	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
4	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
3	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
2	CPS1	<b>PCA Count Pulse Select</b>					
1	CPS0	CPS1	CPS0	Selected PCA input			
		0	0	Internal clock $f_{CLK PERIPH/6}$			
		0	1	Internal clock $f_{CLK PERIPH/2}$			
		1	0	Timer 0 Overflow			
1	1	External clock at ECI/P1.2 pin (max rate = $f_{CLK PERIPH/4}$ )					
0	ECF	<b>PCA Enable Counter Overflow Interrupt</b> Cleared to disable CF bit in CCON to inhibit an interrupt. Set to enable CF bit in CCON to generate an interrupt.					

Reset Value = 00XX X000b

Not bit addressable

The CMOD register includes three additional bits associated with the PCA (see Figure 11 and Table 11).

- The CIDL bit which allows the PCA to stop during idle mode.
- The WDTE bit which enables or disables the watchdog function on module 4.
- The ECF bit which when set causes an interrupt and the PCA overflow flag CF (in the CCON SFR) to be set when the PCA timer overflows.

The CCON register contains the run control bit for the PCA and the flags for the PCA timer (CF) and each module (see Table 12).

- Bit CR (CCON.6) must be set by software to run the PCA. The PCA is shut off by clearing this bit.
- Bit CF: The CF bit (CCON.7) is set when the PCA counter overflows and an interrupt will be generated if the ECF bit in the CMOD register is set. The CF bit can only be cleared by software.
- Bits 0 through 4 are the flags for the modules (bit 0 for module 0, bit 1 for module 1, etc.) and are set by hardware when either a match or a capture occurs. These flags can only be cleared by software.





**Table 12.** CCON Register

CCON - PCA Counter Control Register (D8h)

7	6	5	4	3	2	1	0
CF	CR	-	CCF4	CCF3	CCF2	CCF1	CCF0

Bit Number	Bit Mnemonic	Description
7	CF	<b>PCA Counter Overflow flag</b> Set by hardware when the counter rolls over. CF flags an interrupt if bit ECF in CMOD is set. CF may be set by either hardware or software but can only be cleared by software.
6	CR	<b>PCA Counter Run control bit</b> Must be cleared by software to turn the PCA counter off. Set by software to turn the PCA counter on.
5	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.
4	CCF4	<b>PCA Module 4 interrupt flag</b> Must be cleared by software. Set by hardware when a match or capture occurs.
3	CCF3	<b>PCA Module 3 interrupt flag</b> Must be cleared by software. Set by hardware when a match or capture occurs.
2	CCF2	<b>PCA Module 2 interrupt flag</b> Must be cleared by software. Set by hardware when a match or capture occurs.
1	CCF1	<b>PCA Module 1 interrupt flag</b> Must be cleared by software. Set by hardware when a match or capture occurs.
0	CCF0	<b>PCA Module 0 interrupt flag</b> Must be cleared by software. Set by hardware when a match or capture occurs.

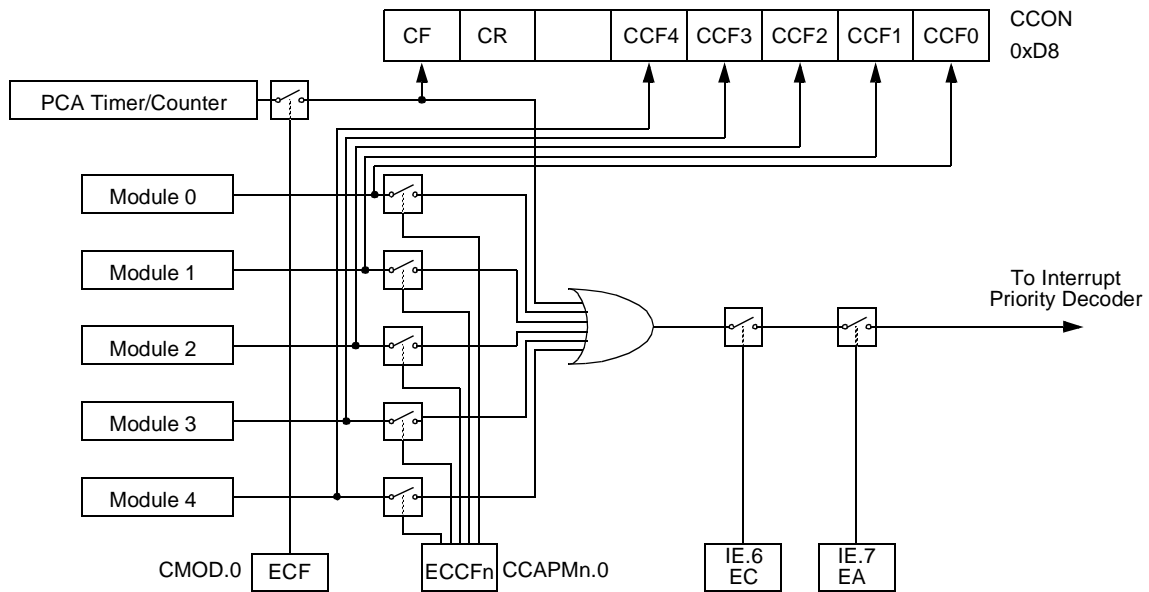
Reset Value = 000X 0000b

Not bit addressable

The watchdog timer function is implemented in module 4 (see Figure 11).

The PCA interrupt system is shown in Figure 9.

**Figure 9.** PCA Interrupt System



**PCA Modules:** each one of the five compare/capture modules has six possible functions. It can perform:

- 16-bit Capture, positive-edge triggered
- 16-bit Capture, negative-edge triggered
- 16-bit Capture, both positive and negative-edge triggered
- 16-bit Software Timer
- 16-bit High-speed Output
- 8-bit Pulse Width Modulator

In addition, module 4 can be used as a Watchdog Timer.

Each module in the PCA has a special function register associated with it. These registers are: CCAPM0 for module 0, CCAPM1 for module 1, etc. (see Table 13). The registers contain the bits that control the mode that each module will operate in.

- The ECCF bit (CCAPMn.0 where n = 0, 1, 2, 3, or 4 depending on the module) enables the CCF flag in the CCON SFR to generate an interrupt when a match or compare occurs in the associated module.
- PWM (CCAPMn.1) enables the pulse width modulation mode.
- The TOG bit (CCAPMn.2) when set causes the CEX output associated with the module to toggle when there is a match between the PCA counter and the module's capture/compare register.
- The match bit MAT (CCAPMn.3) when set will cause the CCFn bit in the CCON register to be set when there is a match between the PCA counter and the module's capture/compare register.
- The next two bits CAPN (CCAPMn.4) and CAPP (CCAPMn.5) determine the edge that a capture input will be active on. The CAPN bit enables the negative edge, and the CAPP bit enables the positive edge. If both bits are set both edges will be enabled and a capture will occur for either transition.
- The last bit in the register ECOM (CCAPMn.6) when set enables the comparator function.

Table 13 shows the CCAPMn settings for the various PCA functions.

**Table 13.** CCAPMn Registers (n = 0-4)

CCAPM0 - PCA Module 0 Compare/Capture Control Register (0DAh)

CCAPM1 - PCA Module 1 Compare/Capture Control Register (0DBh)

CCAPM2 - PCA Module 2 Compare/Capture Control Register (0DCh)

CCAPM3 - PCA Module 3 Compare/Capture Control Register (0DDh)

CCAPM4 - PCA Module 4 Compare/Capture Control Register (0DEh)

7	6	5	4	3	2	1	0
-	ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCFn
Bit Number	Bit Mnemonic	Description					
7	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
6	ECOMn	<b>Enable Comparator</b> Cleared to disable the comparator function. Set to enable the comparator function.					
5	CAPPn	<b>Capture Positive</b> Cleared to disable positive edge capture. Set to enable positive edge capture.					
4	CAPNn	<b>Capture Negative</b> Cleared to disable negative edge capture. Set to enable negative edge capture.					
3	MATn	<b>Match</b> When MATn = 1, a match of the PCA counter with this module's compare/capture register causes the CCFn bit in CCON to be set, flagging an interrupt.					
2	TOGn	<b>Toggle</b> When TOGn = 1, a match of the PCA counter with this module's compare/capture register causes the CEXn pin to toggle.					
1	PWMn	<b>Pulse Width Modulation Mode</b> Cleared to disable the CEXn pin to be used as a pulse width modulated output. Set to enable the CEXn pin to be used as a pulse width modulated output.					
0	CCF0	<b>Enable CCF interrupt</b> Cleared to disable compare/capture flag CCFn in the CCON register to generate an interrupt. Set to enable compare/capture flag CCFn in the CCON register to generate an interrupt.					

Reset Value = X000 0000b

Not bit addressable

**Table 14.** PCA Module Modes (CCAPMn Registers)

ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMm	ECCFn	Module Function
0	0	0	0	0	0	0	No Operation
X	1	0	0	0	0	X	16-bit capture by a positive-edge trigger on CEXn
X	0	1	0	0	0	X	16-bit capture by a negative trigger on CEXn
X	1	1	0	0	0	X	16-bit capture by a transition on CEXn
1	0	0	1	0	0	X	16-bit Software Timer/Compare mode.
1	0	0	1	1	0	X	16-bit High-speed Output
1	0	0	0	0	1	0	8-bit PWM
1	0	0	1	X	0	X	Watchdog Timer (module 4 only)

There are two additional registers associated with each of the PCA modules. They are CCAPnH and CCAPnL and these are the registers that store the 16-bit count when a capture occurs or a compare should occur. When a module is used in the PWM mode these registers are used to control the duty cycle of the output (see Table 15 and Table 16).

**Table 15.** CCAPnH Registers (n = 0-4)

CCAP0H - PCA Module 0 Compare/Capture Control Register High (0FAh)

CCAP1H - PCA Module 1 Compare/Capture Control Register High (0FBh)

CCAP2H - PCA Module 2 Compare/Capture Control Register High (0FCh)

CCAP3H - PCA Module 3 Compare/Capture Control Register High (0FDh)

CCAP4H - PCA Module 4 Compare/Capture Control Register High (0FEh)

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-
Bit Number	Bit Mnemonic	Description					
7-0	-	PCA Module n Compare/Capture Control CCAPnH Value					

Reset Value = 0000 0000b

Not bit addressable

**Table 16.** CCAPnL Registers (n = 0-4)

CCAP0L - PCA Module 0 Compare/Capture Control Register Low (0EAh)

CCAP1L - PCA Module 1 Compare/Capture Control Register Low (0EBh)

CCAP2L - PCA Module 2 Compare/Capture Control Register Low (0ECh)

CCAP3L - PCA Module 3 Compare/Capture Control Register Low (0EDh)

CCAP4L - PCA Module 4 Compare/Capture Control Register Low (0EEh)

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

Bit Number	Bit Mnemonic	Description
7-0	-	<b>PCA Module n Compare/Capture Control</b> CCAPnL Value

Reset Value = 0000 0000b

Not bit addressable

**Table 17.** CH Register

CH - PCA Counter Register High (0F9h)

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

Bit Number	Bit Mnemonic	Description
7-0	-	<b>PCA counter</b> CH Value

Reset Value = 0000 0000b

Not bit addressable

**Table 18.** CL Register

CL - PCA Counter Register Low (0E9h)

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

Bit Number	Bit Mnemonic	Description
7-0	-	<b>PCA Counter</b> CL Value

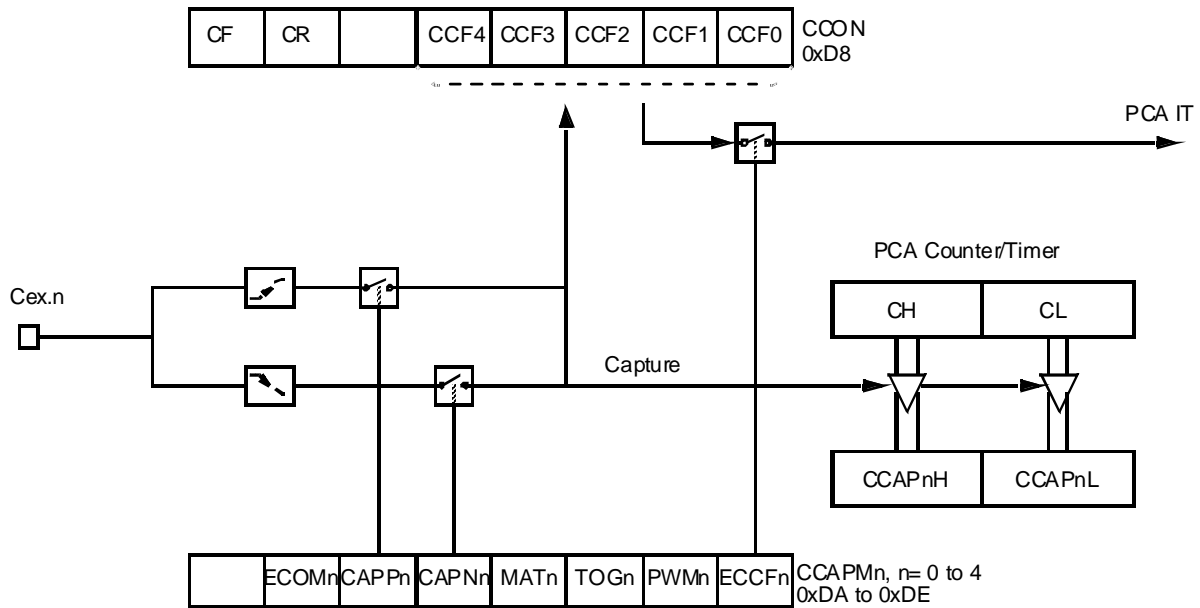
Reset Value = 0000 0000b

Not bit addressable

## PCA Capture Mode

To use one of the PCA modules in the capture mode either one or both of the CCAPM bits CAPN and CAPP for that module must be set. The external CEX input for the module (on port 1) is sampled for a transition. When a valid transition occurs the PCA hardware loads the value of the PCA counter registers (CH and CL) into the module's capture registers (CCAPnL and CCAPnH). If the CCFn bit for the module in the CCON SFR and the ECCFn bit in the CCAPMn SFR are set then an interrupt will be generated (see Figure 10).

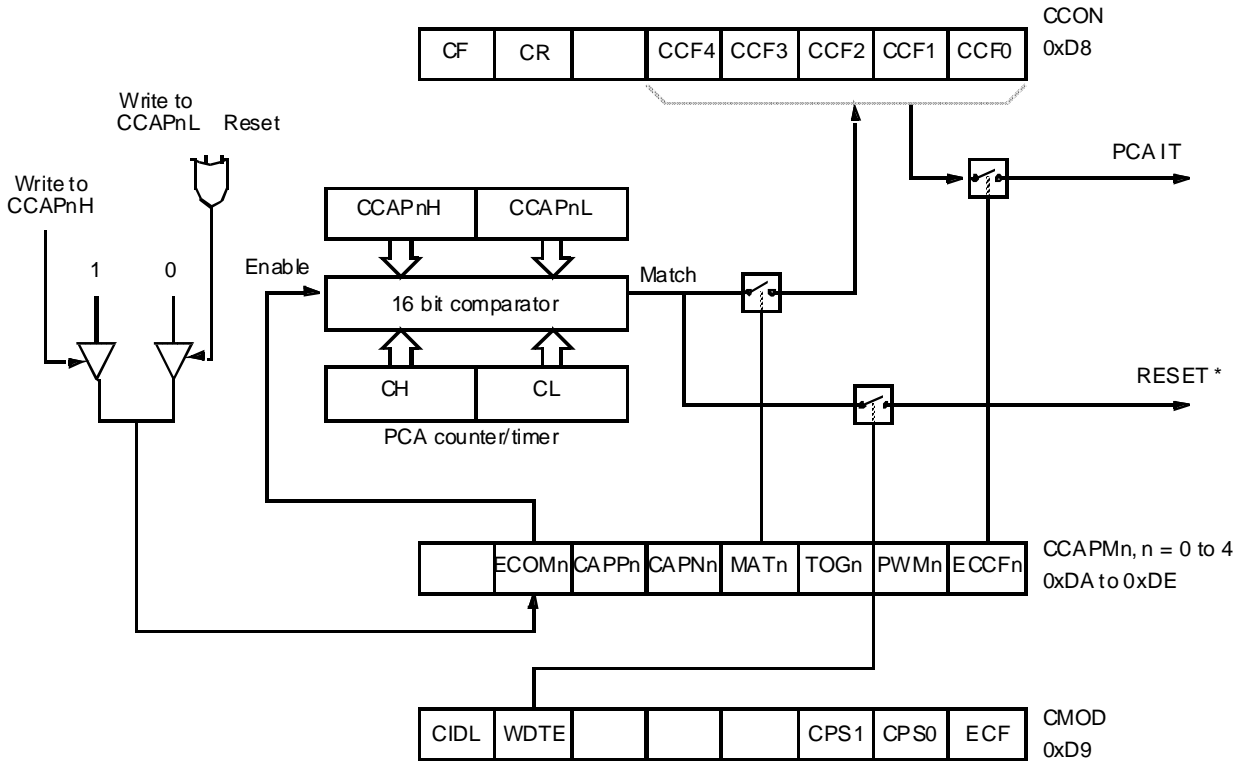
Figure 10. PCA Capture Mode



## 16-bit Software Timer/ Compare Mode

The PCA modules can be used as software timers by setting both the ECOM and MAT bits in the modules CCAPMn register. The PCA timer will be compared to the module's capture registers and when a match occurs an interrupt will occur if the CCFn (CCON SFR) and the ECCFn (CCAPMn SFR) bits for the module are both set (see Figure 11).

**Figure 11. PCA Compare Mode and PCA Watchdog Timer**



Before enabling ECOM bit, CCAPnL and CCAPnH should be set with a non zero value, otherwise an unwanted match could happen. Writing to CCAPnH will set the ECOM bit.

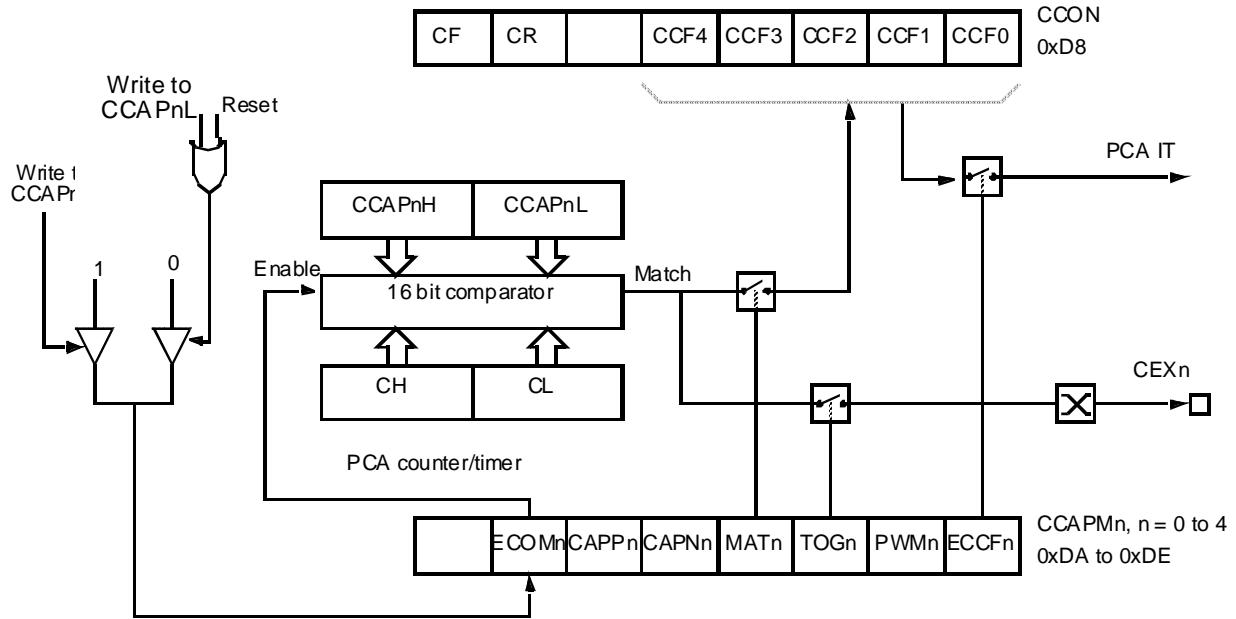
Once ECOM set, writing CCAPnL will clear ECOM so that an unwanted match doesn't occur while modifying the compare value. Writing to CCAPnH will set ECOM. For this reason, user software should write CCAPnL first, and then CCAPnH. Of course, the ECOM bit can still be controlled by accessing to CCAPMn register.

### High-speed Output Mode

In this mode, the CEX output (on port 1) associated with the PCA module will toggle each time a match occurs between the PCA counter and the module's capture registers. To activate this mode the TOG, MAT, and ECOM bits in the module's CCAPMn SFR must be set (see Figure 12).

A prior write must be done to CCAPnL and CCAPnH before writing the ECOMn bit.

**Figure 12. PCA High-speed Output Mode**



Before enabling ECOM bit, CCAPnL and CCAPnH should be set with a non zero value, otherwise an unwanted match could occur.

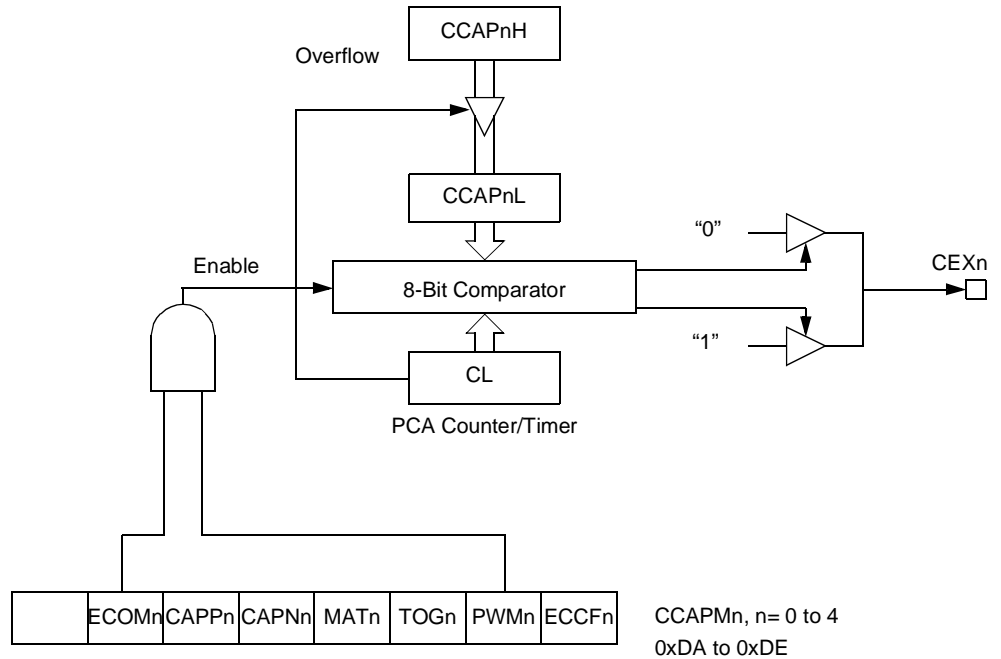
Once ECOM is set, writing CCAPnL will clear ECOM so that an unwanted match doesn't occur while modifying the compare value. Writing to CCAPnH will set ECOM. For this reason, user software should write CCAPnL first, and then CCAPnH. Of course, the ECOM bit can still be controlled by accessing the CCAPMn register.

## Pulse Width Modulator Mode

All of the PCA modules can be used as PWM outputs. Figure 13 shows the PWM function. The frequency of the output depends on the source for the PCA timer. All of the modules will have the same frequency of output because they all share the PCA timer. The duty cycle of each module is independently variable using the module's capture register CCAPL<sub>n</sub>. When the value of the PCA CL SFR is less than the value in the module's CCAPL<sub>n</sub> SFR the output will be low, when it is equal to or greater than the output will be high. When CL overflows from FF to 00, CCAPL<sub>n</sub> is reloaded with the value in CCAPH<sub>n</sub>. This allows updating the PWM without glitches. The PWM and ECOM bits in the module's CCAPM<sub>n</sub> register must be set to enable the PWM mode.



**Figure 13. PCA PWM Mode**



## PCA Watchdog Timer

An on-board watchdog timer is available with the PCA to improve the reliability of the system without increasing chip count. Watchdog timers are useful for systems that are susceptible to noise, power glitches, or electrostatic discharge. Module 4 is the only PCA module that can be programmed as a watchdog. However, this module can still be used for other modes if the watchdog is not needed. Figure 11 shows a diagram of how the watchdog works. The user pre-loads a 16-bit value in the compare registers. Just like the other compare modes, this 16-bit value is compared to the PCA timer value. If a match is allowed to occur, an internal reset will be generated. This will not cause the RST pin to be driven high.

In order to hold off the reset, the user has three options:

1. Periodically change the compare value so it will never match the PCA timer.
2. Periodically change the PCA timer value so it will never match the compare values.
3. Disable the watchdog by clearing the WDTE bit before a match occurs and then re-enable it.

The first two options are more reliable because the watchdog timer is never disabled as in option #3. If the program counter ever goes astray, a match will eventually occur and cause an internal reset. The second option is also not recommended if other PCA modules are being used. Remember, the PCA timer is the time base for all modules; changing the time base for other modules would not be a good idea. Thus, in most applications the first solution is the best option.

This watchdog timer won't generate a reset out on the reset pin.

## Serial I/O Port

The serial I/O port in the T8xc51Rx2 is compatible with the serial I/O port in the 80C52. It provides both synchronous and asynchronous communication modes. It operates as a 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

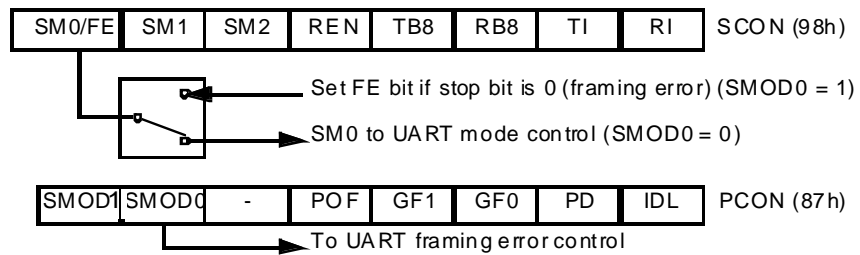
Serial I/O port includes the following enhancements:

- Framing error detection
- Automatic address recognition

## 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 14).

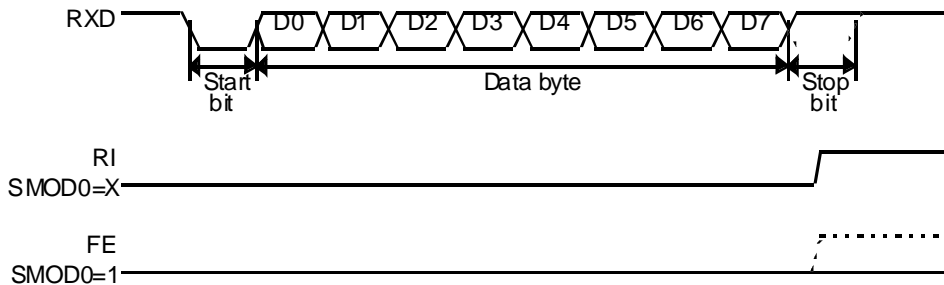
**Figure 14.** Framing Error Block diagram



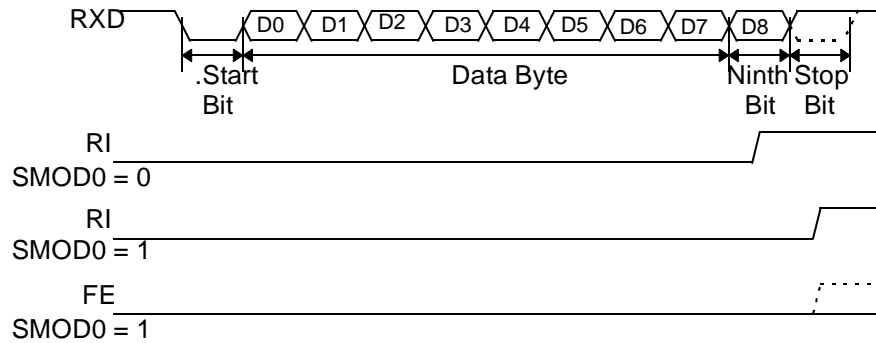
When this feature is enabled, the receiver checks each incoming data frame for a valid stop bit. An invalid stop bit may result from noise on the serial lines or from simultaneous transmission by two CPUs. If a valid stop bit is not found, the Framing Error bit (FE) in SCON register (see Table 22) bit 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 15 and Figure 16).

**Figure 15.** UART Timings in Mode 1



**Figure 16.** UART Timings in Modes 2 and 3



## 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).

## 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:

```
SADDR0101 0110b
SADEN1111 1100b
Given0101 01XXb
```

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

```
Slave A:SADDR1111 0001b
SADEN1111 1010b
Given1111 0X0Xb
```

```
Slave B:SADDR1111 0011b
SADEN1111 1001b
Given1111 0XX1b
```

```
Slave C:SADDR1111 0010b
SADEN1111 1101b
Given1111 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).

## 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 SADEN1111 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:SADDR1111 0001b
      SADEN1111 1010b
Broadcast1111 1X11b,
```

```
Slave B:SADDR1111 0011b
      SADEN1111 1001b
Broadcast1111 1X11b,
```

```
Slave C:SADDR = 1111 0010b
      SADEN1111 1101b
Broadcast1111 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 an address FBh.

## 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.

**Table 19.** SADEN Register  
SADEN - Slave Address Mask Register (B9h)

7	6	5	4	3	2	1	0

Reset Value = 0000 0000b  
Not bit addressable

**Table 20.** SADDR Register

SADDR - Slave Address Register (A9h)

7	6	5	4	3	2	1	0

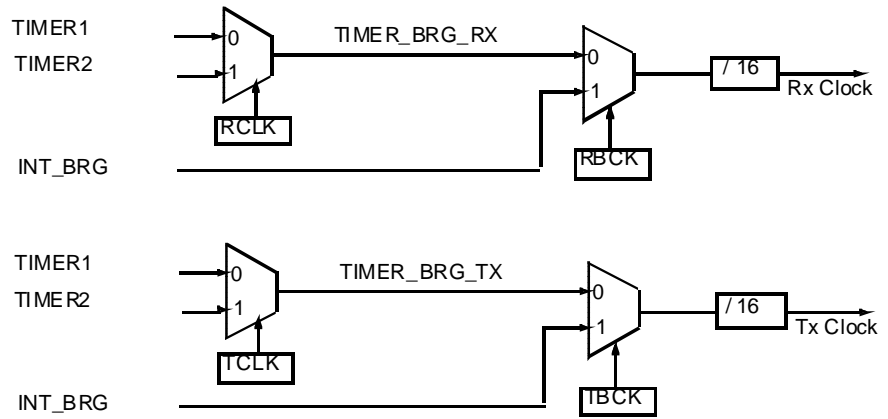
Reset Value = 0000 0000b

Not bit addressable

### Baud Rate Selection for UART for Mode 1 and 3

The Baud Rate Generator for transmit and receive clocks can be selected separately via the T2CON and BDRCON registers.

**Figure 17.** Baud Rate selection



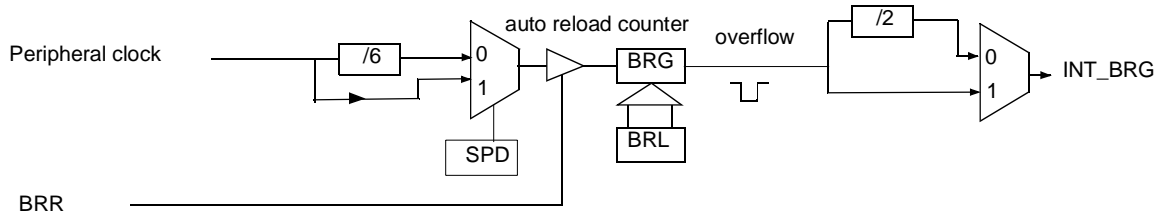
**Table 21.** Baud Rate Selection Table UART

TCLK (T2CON)	RCLK (T2CON)	TBCK (BDRCON)	RBCK (BDRCON)	Clock Source UART Tx	Clock Source UART Rx
0	0	0	0	Timer 1	Timer 1
1	0	0	0	Timer 2	Timer 1
0	1	0	0	Timer 1	Timer 2
1	1	0	0	Timer 2	Timer 2
X	0	1	0	INT_BRG	Timer 1
X	1	1	0	INT_BRG	Timer 2
0	X	0	1	Timer 1	INT_BRG
1	X	0	1	Timer 2	INT_BRG
X	X	1	1	INT_BRG	INT_BRG

## Internal Baud Rate Generator (BRG)

When the internal Baud Rate Generator is used, the Baud Rates are determined by the BRG overflow depending on the BRL reload value, the value of SPD bit (Speed Mode) in BDRCON register and the value of the SMOD1 bit in PCON register.

**Figure 18.** Internal Baud Rate



- The baud rate for UART is token by formula:

$$BaudRate = \frac{2^{SMOD} \times F_{CLKPERIPH}}{2 \times 2 \times 6 \times (1 - SPD) \times 16 \times [256 - (BRL)]}$$

$$(BRL) = 256 - \frac{2^{SMOD1} \times F_{CLKPERIPH}}{2 \times 2 \times 6 \times (1 - SPD) \times 16 \times BaudRate}$$

**Table 22.** 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	<b>Framing Error bit (SMOD0 = 1)</b> 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	<b>Serial port Mode bit 0</b> Refer to SM1 for serial port mode selection. SMOD0 must be cleared to enable access to the SM0 bit																									
6	SM1	<b>Serial port Mode bit 1</b> <table border="1"> <thead> <tr> <th>SM1</th> <th>Mode</th> <th>Description</th> <th>Baud Rate</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Shift Register</td> <td><math>f_{CPU\ PERIPH/6}</math></td> </tr> <tr> <td>1</td> <td>1</td> <td>8-bit UART</td> <td>Variable</td> </tr> <tr> <td>0</td> <td>2</td> <td>9-bit UART</td> <td><math>f_{CPU\ PERIPH/32\ or\ /16}</math></td> </tr> <tr> <td>1</td> <td>3</td> <td>9-bit UART</td> <td>Variable</td> </tr> </tbody> </table>						SM1	Mode	Description	Baud Rate	0	0	Shift Register	$f_{CPU\ PERIPH/6}$	1	1	8-bit UART	Variable	0	2	9-bit UART	$f_{CPU\ PERIPH/32\ or\ /16}$	1	3	9-bit UART	Variable
SM1	Mode	Description	Baud Rate																								
0	0	Shift Register	$f_{CPU\ PERIPH/6}$																								
1	1	8-bit UART	Variable																								
0	2	9-bit UART	$f_{CPU\ PERIPH/32\ or\ /16}$																								
1	3	9-bit UART	Variable																								
5	SM2	<b>Serial port Mode 2 bit/Multiprocessor Communication Enable bit</b> 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	<b>Reception Enable bit</b> Clear to disable serial reception. Set to enable serial reception.																									
3	TB8	<b>Transmitter Bit 8/Ninth bit to transmit in modes 2 and 3</b> 0 transmit a logic 0 in the 9th bit. Set to transmit a logic 1 in the 9th bit.																									
2	RB8	<b>Receiver Bit 8/Ninth bit received in modes 2 and 3</b> 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	<b>Transmit Interrupt flag</b> 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	<b>Receive Interrupt flag</b> Clear to acknowledge interrupt. Set by hardware at the end of the 8th bit time in mode 0, see Figure 15. and Figure 16. in the other modes.																									

Reset Value = 0000 0000b

Bit addressable

**Table 23.** Example of Computed Value when X2 = 1, SMOD1 = 1, SPD = 1

Baud Rates	F <sub>OSC</sub> =16.384 MHz		F <sub>OSC</sub> =24 MHz	
	BRL	Error (%)	BRL	Error (%)
115200	247	1.23	243	0.16
57600	238	1.23	230	0.16
38400	229	1.23	217	0.16
28800	220	1.23	204	0.16
19200	203	0.63	178	0.16
9600	149	0.31	100	0.16
4800	43	1.23	-	-

**Table 24.** Example of Computed Value when X2 = 0, SMOD1 = 0, SPD = 0

Baud Rates	F <sub>OSC</sub> =16.384 MHz		F <sub>OSC</sub> =24 MHz	
	BRL	Error (%)	BRL	Error (%)
4800	247	1.23	243	0.16
2400	238	1.23	230	0.16
1200	220	1.23	202	3.55
600	185	0.16	152	0.16

The baud rate generator can be used for mode 1 or 3 (see Figure 17.), but also for mode 0 for UART, thanks to the bit SRC located in BDRCON register (Table 31.)



## UART Registers

**Table 25.** SADEN Register

SADEN - Slave Address Mask Register for UART (B9h)

7	6	5	4	3	2	1	0

Reset Value = 0000 0000b

**Table 26.** SADDR Register

SADDR - Slave Address Register for UART (A9h)

7	6	5	4	3	2	1	0

Reset Value = 0000 0000b

**Table 27.** SBUF Register

SBUF - Serial Buffer Register for UART (99h)

7	6	5	4	3	2	1	0

Reset Value = XXXX XXXXb

**Table 28.** BRL Register

BRL - Baud Rate Reload Register for the internal baud rate generator, UART (9Ah)

7	6	5	4	3	2	1	0

Reset Value = 0000 0000b

**Table 29.** 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	<b>Timer 2 overflow Flag</b> Must be cleared by software. Set by hardware on Timer 2 overflow, if RCLK=0 and TCLK=0.					
6	EXF2	<b>Timer 2 External Flag</b> 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	<b>Receive Clock bit for UART</b> Cleared 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	<b>Transmit Clock bit for UART</b> Cleared 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	<b>Timer 2 External Enable bit</b> Cleared 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	<b>Timer 2 Run control bit</b> Cleared to turn off Timer 2. Set to turn on Timer 2.					
1	C/T2#	<b>Timer/Counter 2 select bit</b> Cleared for timer operation (input from internal clock system: F <sub>CLK PERIPH</sub> ). Set for counter operation (input from T2 input pin, falling edge trigger). Must be 0 for clock out mode.					
0	CP/RL2#	<b>Timer 2 Capture/Reload bit</b> If RCLK = 1 or TCLK = 1, CP/RL2# is ignored and timer is forced to auto-reload on Timer 2 overflow. Cleared 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 30.** 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	<b>Serial Port Mode bit 1 for UART</b> Set to select double baud rate in mode 1, 2 or 3.
6	SMOD0	<b>Serial Port Mode bit 0 for UART</b> Cleared to select SM0 bit in SCON register. Set to select FE bit in SCON register.
5	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.
4	POF	<b>Power-off Flag</b> Cleared to recognize next reset type. Set by hardware when V <sub>CC</sub> rises from 0 to its nominal voltage. Can also be set by software.
3	GF1	<b>General purpose Flag</b> Cleared by user for general purpose usage. Set by user for general purpose usage.
2	GF0	<b>General purpose Flag</b> Cleared by user for general purpose usage. Set by user for general purpose usage.
1	PD	<b>Power-down mode bit</b> Cleared by hardware when reset occurs. Set to enter power-down mode.
0	IDL	<b>Idle mode bit</b> Cleared 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.

**Table 31.** BDRCON Register

BDRCON - Baud Rate Control Register (9Bh)

7	6	5	4	3	2	1	0
-	-	-	BRR	TBCK	RBCK	SPD	SRC
Bit Number	Bit Mnemonic	Description					
7	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit					
6	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit					
5	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
4	BRR	<b>Baud Rate Run Control bit</b> Cleared to stop the internal Baud Rate Generator. Set to start the internal Baud Rate Generator.					
3	TBCK	<b>Transmission Baud rate Generator Selection bit for UART</b> Cleared to select Timer 1 or Timer 2 for the Baud Rate Generator. Set to select internal Baud Rate Generator.					
2	RBCK	<b>Reception Baud Rate Generator Selection bit for UART</b> Cleared to select Timer 1 or Timer 2 for the Baud Rate Generator. Set to select internal Baud Rate Generator.					
1	SPD	<b>Baud Rate Speed Control bit for UART</b> Cleared to select the SLOW Baud Rate Generator. Set to select the FAST Baud Rate Generator.					
0	SRC	<b>Baud Rate Source select bit in Mode 0 for UART</b> Cleared to select $F_{OSC}/12$ as the Baud Rate Generator ( $F_{CLK\_PERIPH}/6$ in X2 mode). Set to select the internal Baud Rate Generator for UARTs in mode 0.					

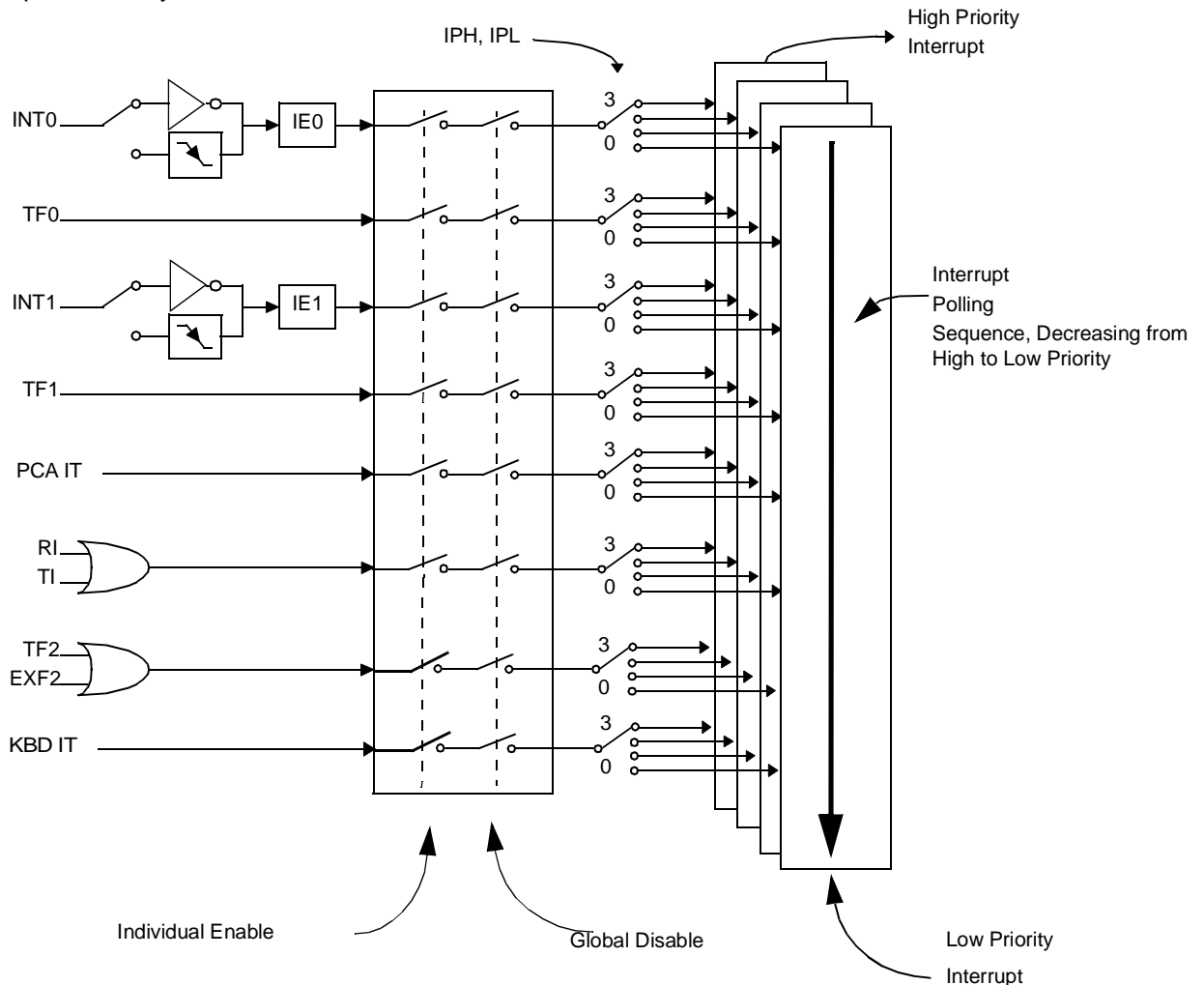
Reset Value = XXX0 0000b

Not bit addressable

## Interrupt System

The T8xc51Rx2 have a total of 8 interrupt vectors: two external interrupts ( $\overline{INT0}$  and  $\overline{INT1}$ ), three timer interrupts (timers 0, 1 and 2), the serial port interrupt, Keyboard interrupt and the PCA global interrupt. These interrupts are shown in Figure 19.

**Figure 19.** 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 (Table 36 and Table 34). This register also contains a global disable bit, which must be cleared to disable all interrupts at once.

Each interrupt source also can be individually programmed to one out of four priority levels by setting or clearing a bit in the Interrupt Priority register (Table 37) and in the Interrupt Priority High register (Table 35 and Table 36) shows the bit values and priority levels associated with each combination.

## Registers

The PCA interrupt vector is located at address 0033H, the Keyboard interrupt vector is located at address 004BH. All other vectors addresses are the same as standard C52 devices.

**Table 32.** Priority Level Bit Values

IPH.x	IPL.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 33.** IEO Register

IE0 - Interrupt Enable Register (A8h)

7	6	5	4	3	2	1	0
EA	EC	ET2	ES	ET1	EX1	ET0	EX0
Bit Number	Bit Mnemonic	Description					
7	EA	<b>Enable All interrupt bit</b> Cleared to disable all interrupts. Set to enable all interrupts.					
6	EC	<b>PCA interrupt enable bit</b> Cleared to disable. Set to enable.					
5	ET2	<b>Timer 2 overflow interrupt enable bit</b> Cleared to disable Timer 2 overflow interrupt. Set to enable Timer 2 overflow interrupt.					
4	ES	<b>Serial port enable bit</b> Cleared to disable serial port interrupt. Set to enable serial port interrupt.					
3	ET1	<b>Timer 1 overflow interrupt enable bit</b> Cleared to disable timer 1 overflow interrupt. Set to enable timer 1 overflow interrupt.					
2	EX1	<b>External interrupt 1 enable bit</b> Cleared to disable external interrupt 1. Set to enable external interrupt 1.					
1	ET0	<b>Timer 0 overflow interrupt enable bit</b> Cleared to disable timer 0 overflow interrupt. Set to enable timer 0 overflow interrupt.					
0	EX0	<b>External interrupt 0 enable bit</b> Cleared to disable external interrupt 0. Set to enable external interrupt 0.					

Reset Value = 0000 0000b

Bit addressable

**Table 34.** IPL0 Register

IPL0 - Interrupt Priority Register (B8h)

7	6	5	4	3	2	1	0
-	PPCL	PT2L	PSL	PT1L	PX1L	PT0L	PX0L
Bit Number	Bit Mnemonic	Description					
7	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
6	PPCL	<b>PCA interrupt priority bit</b> Refer to PPCH for priority level.					
5	PT2L	<b>Timer 2 overflow interrupt priority bit</b> Refer to PT2H for priority level.					
4	PSL	<b>Serial port priority bit</b> Refer to PSH for priority level.					
3	PT1L	<b>Timer 1 overflow interrupt priority bit</b> Refer to PT1H for priority level.					
2	PX1L	<b>External interrupt 1 priority bit</b> Refer to PX1H for priority level.					
1	PT0L	<b>Timer 0 overflow interrupt priority bit</b> Refer to PT0H for priority level.					
0	PX0L	<b>External interrupt 0 priority bit</b> Refer to PX0H for priority level.					

Reset Value = X000 0000b

Bit addressable



**Table 35.** IPH0 Register

IPH0 - Interrupt Priority High Register (B7h)

	<b>7</b>	<b>6</b>	<b>5</b>	<b>4</b>	<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>
	-	PPCH	PT2H	PSH	PT1H	PX1H	PT0H	PX0H

Bit Number	Bit Mnemonic	Description
7	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.
6	PPCH	<b>PCA interrupt priority high bit.</b> <u>PPCH</u> <u>PPCL</u> <u>Priority Level</u> 0 0 Lowest 0 1 1 0 1 1 Highest
5	PT2H	<b>Timer 2 overflow interrupt priority high bit</b> <u>PT2H</u> <u>PT2L</u> <u>Priority Level</u> 0 0 Lowest 0 1 1 0 1 1 Highest
4	PSH	<b>Serial port priority high bit</b> <u>PSH</u> <u>PSL</u> <u>Priority Level</u> 0 0 Lowest 0 1 1 0 1 1 Highest
3	PT1H	<b>Timer 1 overflow interrupt priority high bit</b> <u>PT1H</u> <u>PT1L</u> <u>Priority Level</u> 0 0 Lowest 0 1 1 0 1 1 Highest
2	PX1H	<b>External interrupt 1 priority high bit</b> <u>PX1H</u> <u>PX1L</u> <u>Priority Level</u> 0 0 Lowest 0 1 1 0 1 1 Highest
1	PT0H	<b>Timer 0 overflow interrupt priority high bit</b> <u>PT0H</u> <u>PT0L</u> <u>Priority Level</u> 0 0 Lowest 0 1 1 0 1 1 Highest
0	PX0H	<b>External interrupt 0 priority high bit</b> <u>PX0H</u> <u>PX0L</u> <u>Priority Level</u> 0 0 Lowest 0 1 1 0 1 1 Highest

Reset Value = X000 0000b  
Not bit addressable

**Table 36.** IE1 Register

IE1 - Interrupt Enable Register (B1h)

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

Bit Number	Bit Mnemonic	Description
7	-	Reserved
6	-	Reserved
5	-	Reserved
4	-	Reserved
3	-	Reserved
2	-	Reserved
1	-	Reserved
0	KBD	<b>Keyboard interrupt Enable bit</b> Cleared to disable keyboard interrupt. Set to enable keyboard interrupt.

Reset Value = XXXX XXX0b

Bit addressable

**Table 37. IPL1 Register**

IPL1 - Interrupt Priority Register (B2h)

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

Bit Number	Bit Mnemonic	Description
7	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.
6	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.
5	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.
4	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.
3	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.
2	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.
1	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.
0	KBDL	<b>Keyboard Interrupt Priority bit</b> Refer to KBDH for priority level.

Reset Value = XXXX XXX0b

Bit addressable

**Table 38.** IPH1 Register

IPH1 - Interrupt Priority High Register (B3h)

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

Bit Number	Bit Mnemonic	Description																				
7	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.																				
6	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.																				
5	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.																				
4	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.																				
3	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.																				
2	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.																				
1	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.																				
0	KBDH	<b>Keyboard interrupt Priority High bit</b> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 15%;"><u>KB</u></th> <th style="width: 15%;"><u>DH</u></th> <th style="width: 15%;"><u>KBDL</u></th> <th style="width: 15%;"><u>Priority Level</u></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td></td> <td>Lowest</td> </tr> <tr> <td>0</td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>1</td> <td>0</td> <td></td> <td></td> </tr> <tr> <td>1</td> <td>1</td> <td></td> <td>Highest</td> </tr> </tbody> </table>	<u>KB</u>	<u>DH</u>	<u>KBDL</u>	<u>Priority Level</u>	0	0		Lowest	0	1			1	0			1	1		Highest
<u>KB</u>	<u>DH</u>	<u>KBDL</u>	<u>Priority Level</u>																			
0	0		Lowest																			
0	1																					
1	0																					
1	1		Highest																			

Reset Value = XXXX XXX0b

Not bit addressable

## Interrupt Sources and Vector Addresses

**Table 39.** Interrupt Sources and Vector Addresses

Number	Polling Priority	Interrupt Source	Interrupt Request	Vector Address
0	0	Reset		0000h
1	1	INT0	IE0	0003h
2	2	Timer 0	TF0	000Bh
3	3	INT1	IE1	0013h
4	4	Timer 1	IF1	001Bh
5	6	UART	RI+TI	0023h
6	7	Timer 2	TF2+EXF2	002Bh
7	5	PCA	CF + CCFn (n = 0-4)	0033h
8	8	Keyboard	KBDIT	003Bh

## Keyboard Interface

The T8xc51Rx2 implement a keyboard interface allowing the connection of a 8 x n matrix keyboard. It is based on 8 inputs with programmable interrupt capability on both high or low level. These inputs are available as alternate function of P1 and allow to exit from idle and power-down modes.

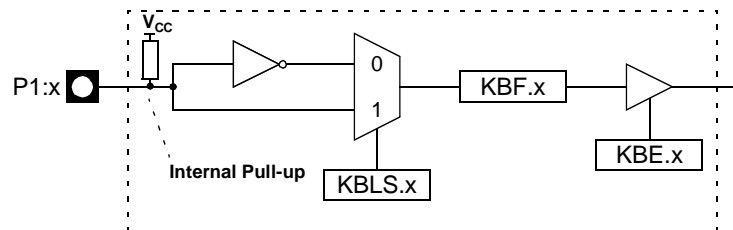
The keyboard interfaces with the C51 core through 3 special function registers: KBL5, the Keyboard Level Selection register (Table 42), KBE, The Keyboard Interrupt Enable register (Table 41), and KBF, the Keyboard Flag register (Table 40).

## Interrupt

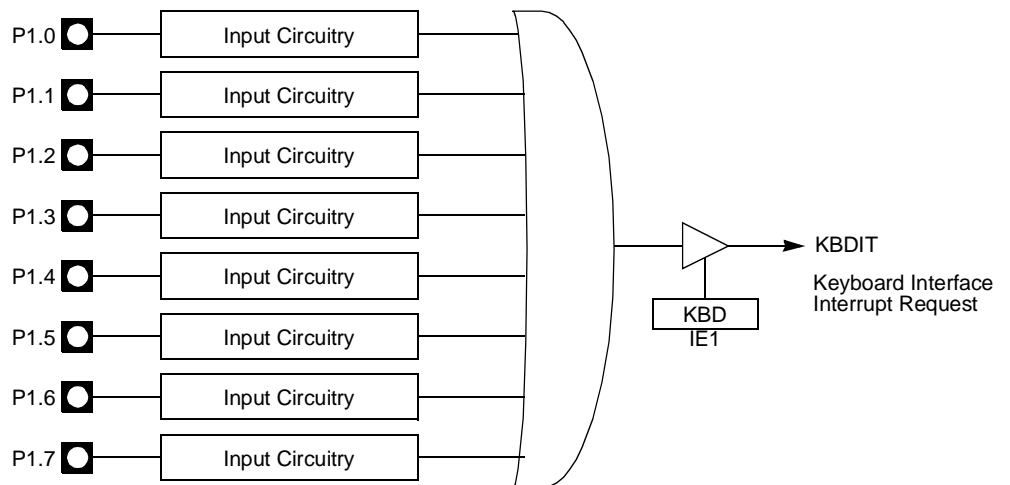
The keyboard inputs are considered as 8 independent interrupt sources sharing the same interrupt vector. An interrupt enable bit (KBD in IE1) allows global enable or disable of the keyboard interrupt (see Figure 20). As detailed in Figure 21 each keyboard input has the capability to detect a programmable level according to KBL5.x bit value. Level detection is then reported in interrupt flags KBF.x that can be masked by software using KBE.x bits.

This structure allow keyboard arrangement from 1 x n to 8 x n matrix and allows usage of P1 inputs for other purpose.

**Figure 20.** Keyboard Interface Block Diagram



**Figure 21.** Keyboard Input Circuitry



## Power Reduction Mode

P1 inputs allow exit from idle and power-down modes as detailed in Section “Power-down Mode”, page 59.

## Registers

**Table 40. KBF Register**

KBF - Keyboard Flag Register (9Eh)

7	6	5	4	3	2	1	0
KBF7	KBF6	KBF5	KBF4	KBF3	KBF2	KBF1	KBF0
Bit Number	Bit Mnemonic	Description					
7	KBF7	<b>Keyboard line 7 flag</b> Set by hardware when the Port line 7 detects a programmed level. It generates a Keyboard interrupt request if the KBKBIE.7 bit in KBIE register is set. Must be cleared by software.					
6	KBF6	<b>Keyboard line 6 flag</b> Set by hardware when the Port line 6 detects a programmed level. It generates a Keyboard interrupt request if the KBIE.6 bit in KBIE register is set. Must be cleared by software.					
5	KBF5	<b>Keyboard line 5 flag</b> Set by hardware when the Port line 5 detects a programmed level. It generates a Keyboard interrupt request if the KBIE.5 bit in KBIE register is set. Must be cleared by software.					
4	KBF4	<b>Keyboard line 4 flag</b> Set by hardware when the Port line 4 detects a programmed level. It generates a Keyboard interrupt request if the KBIE.4 bit in KBIE register is set. Must be cleared by software.					
3	KBF3	<b>Keyboard line 3 flag</b> Set by hardware when the Port line 3 detects a programmed level. It generates a Keyboard interrupt request if the KBIE.3 bit in KBIE register is set. Must be cleared by software.					
2	KBF2	<b>Keyboard line 2 flag</b> Set by hardware when the Port line 2 detects a programmed level. It generates a Keyboard interrupt request if the KBIE.2 bit in KBIE register is set. Must be cleared by software.					
1	KBF1	<b>Keyboard line 1 flag</b> Set by hardware when the Port line 1 detects a programmed level. It generates a Keyboard interrupt request if the KBIE.1 bit in KBIE register is set. Must be cleared by software.					
0	KBF0	<b>Keyboard line 0 flag</b> Set by hardware when the Port line 0 detects a programmed level. It generates a Keyboard interrupt request if the KBIE.0 bit in KBIE register is set. Must be cleared by software.					

Reset Value = 0000 0000b

**Table 41.** KBE Register

KBE - Keyboard Input Enable Register (9Dh)

7	6	5	4	3	2	1	0
KBE7	KBE6	KBE5	KBE4	KBE3	KBE2	KBE1	KBE0
Bit Number	Bit Mnemonic	Description					
7	KBE7	<b>Keyboard line 7 enable bit</b> Cleared to enable standard I/O pin. Set to enable KBF.7 bit in KBF register to generate an interrupt request.					
6	KBE6	<b>Keyboard line 6 enable bit</b> Cleared to enable standard I/O pin. Set to enable KBF.6 bit in KBF register to generate an interrupt request.					
5	KBE5	<b>Keyboard line 5 enable bit</b> Cleared to enable standard I/O pin. Set to enable KBF.5 bit in KBF register to generate an interrupt request.					
4	KBE4	<b>Keyboard line 4 enable bit</b> Cleared to enable standard I/O pin. Set to enable KBF.4 bit in KBF register to generate an interrupt request.					
3	KBE3	<b>Keyboard line 3 enable bit</b> Cleared to enable standard I/O pin. Set to enable KBF.3 bit in KBF register to generate an interrupt request.					
2	KBE2	<b>Keyboard line 2 enable bit</b> Cleared to enable standard I/O pin. Set to enable KBF.2 bit in KBF register to generate an interrupt request.					
1	KBE1	<b>Keyboard line 1 enable bit</b> Cleared to enable standard I/O pin. Set to enable KBF.1 bit in KBF register to generate an interrupt request.					
0	KBE0	<b>Keyboard line 0 enable bit</b> Cleared to enable standard I/O pin. Set to enable KBF.0 bit in KBF register to generate an interrupt request.					

Reset Value = 0000 0000b



**Table 42.** KBLS Register

KBLS - Keyboard Level Selector Register (9Ch)

7	6	5	4	3	2	1	0
KBLS7	KBLS6	KBLS5	KBLS4	KBLS3	KBLS2	KBLS1	KBLS0
Bit Number	Bit Mnemonic	Description					
7	KBLS7	<b>Keyboard line 7 level selection bit</b> Cleared to enable a low level detection on Port line 7. Set to enable a high level detection on Port line 7.					
6	KBLS6	<b>Keyboard line 6 level selection bit</b> Cleared to enable a low level detection on Port line 6. Set to enable a high level detection on Port line 6.					
5	KBLS5	<b>Keyboard line 5 level selection bit</b> Cleared to enable a low level detection on Port line 5. Set to enable a high level detection on Port line 5.					
4	KBLS4	<b>Keyboard line 4 level selection bit</b> Cleared to enable a low level detection on Port line 4. Set to enable a high level detection on Port line 4.					
3	KBLS3	<b>Keyboard line 3 level selection bit</b> Cleared to enable a low level detection on Port line 3. Set to enable a high level detection on Port line 3.					
2	KBLS2	<b>Keyboard line 2 level selection bit</b> Cleared to enable a low level detection on Port line 2. Set to enable a high level detection on Port line 2.					
1	KBLS1	<b>Keyboard line 1 level selection bit</b> Cleared to enable a low level detection on Port line 1. Set to enable a high level detection on Port line 1.					
0	KBLS0	<b>Keyboard line 0 level selection bit</b> Cleared to enable a low level detection on Port line 0. Set to enable a high level detection on Port line 0.					

Reset Value = 0000 0000b

## Power Management

### Idle Mode

An instruction that sets PCON.0 indicates that it is the last instruction to be executed before going into Idle mode. In 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 level.

There are two ways to terminate the Idle mode. 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 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.

### Power-down Mode

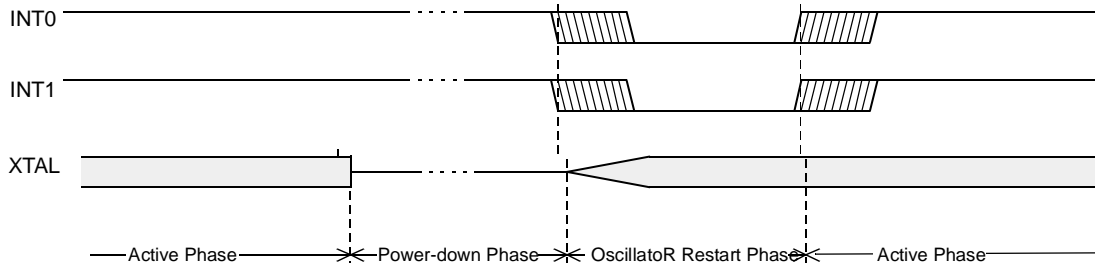
To save maximum power, a power-down mode can be invoked by software (refer to Table 30, PCON register).

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  $\overline{INT0}$ ,  $\overline{INT1}$  and Keyboard Interrupts are useful to exit from power-down. Thus, the interrupt must be enabled and configured as level - or edge - sensitive interrupt input. When Keyboard Interrupt occurs after a power-down mode, 1024 clocks are necessary to exit to power-down mode and enter in operating mode.

Holding the pin low restarts the oscillator but bringing the pin high completes the exit as detailed in Figure 22. 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 is 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 T8xc51Rx2 into power-down mode.

**Figure 22.** 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 43 shows the state of ports during idle and power-down modes.

**Table 43.** State of Ports

Mode	Program Memory	ALE	PSEN	PORT0	PORT1	PORT2	PORT3
Idle	Internal	1	1	Port Data <sup>(1)</sup>	Port Data	Port Data	Port Data
Idle	External	1	1	Floating	Port Data	Address	Port Data
Power-down	Internal	0	0	Port Data <sup>(1)</sup>	Port Data	Port Data	Port Data
Power-down	External	0	0	Floating	Port Data	Port Data	Port Data

Note: 1. Port 0 can force a 0 level. A "one" will leave port floating.

## Hardware Watchdog Timer

The WDT is intended as a recovery method in situations where the CPU may be subjected to software upset. The WDT consists of a 14-bit counter and the WatchDog Timer Reset (WDTRST) SFR. The WDT is by default disabled from exiting reset. To enable the WDT, user must write 01EH and 0E1H in sequence to the WDTRST, SFR location 0A6H. When WDT is enabled, it will increment every machine cycle while the oscillator is running and there is no way to disable the WDT except through reset (either hardware reset or WDT overflow reset). When WDT overflows, it will drive an output RESET HIGH pulse at the RST-pin.

## Using the WDT

To enable the WDT, user must write 01EH and 0E1H in sequence to the WDTRST, SFR location 0A6H. When WDT is enabled, the user needs to service it by writing to 01EH and 0E1H to WDTRST to avoid WDT overflow. The 14-bit counter overflows when it reaches 16383 (3FFFH) and this will reset the device. When WDT is enabled, it will increment every machine cycle while the oscillator is running. Therefore, the user must reset the WDT at least every 16383 machine cycles. To reset the WDT the user must write 01EH and 0E1H to WDTRST. WDTRST is a write only register. The WDT counter cannot be read or written. When WDT overflows, it will generate an output RESET pulse at the RST-pin. The RESET pulse duration is  $96 \times T_{CLK\ PERIPH}$ , where  $T_{CLK\ PERIPH} = 1/F_{CLK\ PERIPH}$ . To make the best use of the WDT, it should be serviced in those sections of code that will periodically be executed within the time required to prevent a WDT reset.

To have a more powerful WDT, a  $2^7$  counter has been added to extend the Time-out capability, ranking from 16 ms to 2s @  $F_{osc} = 12\ MHz$ . To manage this feature, refer to WDTPRG register description, Table 44.

**Table 44.** WDTRST Register

WDTRST - Watchdog Reset Register (0A6h)

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

Reset Value = XXXX XXXXb

Write only, this SFR is used to reset/enable the WDT by writing 01EH then 0E1H in sequence.

**Table 45.** WDTPRG Register

WDTPRG - Watchdog Timer Out Register (0A7h)

	7	6	5	4	3	2	1	0
	-	-	-	-	-	S2	S1	S0

Bit Number	Bit Mnemonic	Description																																				
7	-	<b>Reserved</b> The value read from this bit is undetermined. Do not try to set this bit.																																				
6	-																																					
5	-																																					
4	-																																					
3	-																																					
2	S2	<b>WDT Time-out select bit 2</b>																																				
1	S1	<b>WDT Time-out select bit 1</b>																																				
0	S0	<b>WDT Time-out select bit 0</b>																																				
		<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;"><b>S2</b></th> <th style="width: 10%;"><b>S1</b></th> <th style="width: 10%;"><b>S0</b></th> <th style="width: 70%;"><b>Selected Time-out</b></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td><math>(2^{14} - 1)</math> machine cycles, 16.3 ms @ <math>F_{osc}=12</math> MHz</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td><math>(2^{15} - 1)</math> machine cycles, 32.7 ms @ <math>F_{osc}=12</math> MHz</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td><math>(2^{16} - 1)</math> machine cycles, 65.5 ms @ <math>F_{osc}=12</math> MHz</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td><math>(2^{17} - 1)</math> machine cycles, 131 ms @ <math>F_{osc}=12</math> MHz</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td><math>(2^{18} - 1)</math> machine cycles, 262 ms @ <math>F_{osc}=12</math> MHz</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td><math>(2^{19} - 1)</math> machine cycles, 542 ms @ <math>F_{osc}=12</math> MHz</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td><math>(2^{20} - 1)</math> machine cycles, 1.05 s @ <math>F_{osc}=12</math> MHz</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td><math>(2^{21} - 1)</math> machine cycles, 2.09 s @ <math>F_{osc}=12</math> MHz</td> </tr> </tbody> </table>	<b>S2</b>	<b>S1</b>	<b>S0</b>	<b>Selected Time-out</b>	0	0	0	$(2^{14} - 1)$ machine cycles, 16.3 ms @ $F_{osc}=12$ MHz	0	0	1	$(2^{15} - 1)$ machine cycles, 32.7 ms @ $F_{osc}=12$ MHz	0	1	0	$(2^{16} - 1)$ machine cycles, 65.5 ms @ $F_{osc}=12$ MHz	0	1	1	$(2^{17} - 1)$ machine cycles, 131 ms @ $F_{osc}=12$ MHz	1	0	0	$(2^{18} - 1)$ machine cycles, 262 ms @ $F_{osc}=12$ MHz	1	0	1	$(2^{19} - 1)$ machine cycles, 542 ms @ $F_{osc}=12$ MHz	1	1	0	$(2^{20} - 1)$ machine cycles, 1.05 s @ $F_{osc}=12$ MHz	1	1	1	$(2^{21} - 1)$ machine cycles, 2.09 s @ $F_{osc}=12$ MHz
<b>S2</b>	<b>S1</b>	<b>S0</b>	<b>Selected Time-out</b>																																			
0	0	0	$(2^{14} - 1)$ machine cycles, 16.3 ms @ $F_{osc}=12$ MHz																																			
0	0	1	$(2^{15} - 1)$ machine cycles, 32.7 ms @ $F_{osc}=12$ MHz																																			
0	1	0	$(2^{16} - 1)$ machine cycles, 65.5 ms @ $F_{osc}=12$ MHz																																			
0	1	1	$(2^{17} - 1)$ machine cycles, 131 ms @ $F_{osc}=12$ MHz																																			
1	0	0	$(2^{18} - 1)$ machine cycles, 262 ms @ $F_{osc}=12$ MHz																																			
1	0	1	$(2^{19} - 1)$ machine cycles, 542 ms @ $F_{osc}=12$ MHz																																			
1	1	0	$(2^{20} - 1)$ machine cycles, 1.05 s @ $F_{osc}=12$ MHz																																			
1	1	1	$(2^{21} - 1)$ machine cycles, 2.09 s @ $F_{osc}=12$ MHz																																			

Reset Value = XXXX X000

### WDT During Power-down and Idle

In Power-down mode the oscillator stops, which means the WDT also stops. While in Power-down mode the user does not need to service the WDT. There are 2 methods of exiting Power-down mode: by a hardware reset or via a level activated external interrupt which is enabled prior to entering Power-down mode. When Power-down is exited with hardware reset, servicing the WDT should occur as normal, whenever the T8xc51Rx2 is reset. Exiting Power-down with an interrupt is significantly different. The interrupt is held low long enough for the oscillator to stabilize. When the interrupt is brought high, the interrupt is serviced. To prevent the WDT from resetting the device while the interrupt pin is held low, the WDT is not started until the interrupt is pulled high. It is suggested that the WDT be reset during the interrupt service routine.

To ensure that the WDT does not overflow within a few states of exiting of power-down, it is better to reset the WDT just before entering power-down.

In the Idle mode, the oscillator continues to run. To prevent the WDT from resetting the T8xc51Rx2 while in Idle mode, the user should always set up a timer that will periodically exit Idle, service the WDT, and re-enter Idle mode.

## ROM

### ROM Structure

The T89C51RB2/RC2 ROM memory is divided in two different arrays:

- The code array: 16/32K bytes.
- The config byte: 1 byte.

### Hardware Config Byte

The config byte sets the starting microcontroller options and the security levels.

The starting options are X2 mode, and XRAM.

7	6	5	4	3	2	1	0
X2	-	-	-	XRAM	-	LB1	LB0

Bit Number	Bit Mnemonic	Description
7	X2	<b>X2 Mode</b> Cleared to force X2 mode (6 clocks per instruction) Set to force X1 mode, Standard Mode.
6	-	<b>Reserved</b>
5	-	<b>Reserved</b>
4	-	<b>Reserved</b>
3	XRAM	<b>XRAM config bit</b> Set this bit to enable XRAM. Clear this bit to disable XRAM.
2	-	<b>Reserved</b>
1-0	LB0-1	User Program ROM Lock Bits see Table 46.

### ROM Lock System

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

### Program ROM Lock Bits

**Table 46.** Program Lock bits

Program Lock Bits			Protection Description
Security level	LB0	LB1	
1	U	U	No program lock features enabled.
2	P	U	Reserved. Do not use.
3	U	P	MOV <sub>C</sub> instruction executed from external program memory are disabled from fetching code bytes from internal memory, $\overline{EA}$ is sampled and latched on reset. Verify disable.

Notes: 1. U: unprogrammed  
P: programmed

2. The lock bits when programmed according to Table 46 will provide different level of protection for the on-chip code and data.

## 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 (Table 47). 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.

**Table 47.** 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	<b>Serial port Mode bit 1</b> Set to select double baud rate in mode 1, 2 or 3.					
6	SMOD0	<b>Serial port Mode bit 0</b> Cleared to select SM0 bit in SCON register. Set to select FE bit in SCON register.					
5	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit.					
4	POF	<b>Power-off Flag</b> Cleared 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	<b>General purpose Flag</b> Cleared by user for general purpose usage. Set by user for general purpose usage.					
2	GF0	<b>General purpose Flag</b> Cleared by user for general purpose usage. Set by user for general purpose usage.					
1	PD	<b>Power-down mode bit</b> Cleared by hardware when reset occurs. Set to enter power-down mode.					
0	IDL	<b>Idle mode bit</b> Cleared by hardware when interrupt or reset occurs. Set to enter idle mode.					

Reset Value = 00X1 0000b

Not bit addressable

## 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 48.** AUXR Register

AUXR - Auxiliary Register (8Eh)

7	6	5	4	3	2	1	0
-	-	M0	-	XRS1	XRS0	EXTRAM	AO
Bit Number	Bit Mnemonic	Description					
7	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit					
6	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit					
5	M0	<b>Pulse length</b> Cleared to stretch MOVX control: the $\overline{RD}$ and the $\overline{WR}$ pulse length is 6 clock periods (default). Set to stretch MOVX control: the $\overline{RD}$ and the $\overline{WR}$ pulse length is 30 clock periods.					
4	-	<b>Reserved</b> The value read from this bit is indeterminate. Do not set this bit					
3	XRS1	<b>XRAM Size</b>					
2	XRS0	<u>XRS1</u>	<u>XRS0</u>	<u>XRAM Size</u>			
		0	0	256 bytes (default)			
		0	1	512 bytes			
		1	0	768 bytes			
1	1	1024 bytes					
1	EXTRAM	<b>EXTRAM bit</b> Cleared to access internal XRAM using movx @ Ri/ @ DPTR. Set to access external memory. Programmed by hardware after Power-up regarding Hardware Security Byte (HSB), default setting, XRAM selected.					
0	AO	<b>ALE Output bit</b> Cleared, ALE is emitted at a constant rate of 1/6 the oscillator frequency (or 1/3 if X2 mode is used) (default). Set, ALE is active only during a MOVX or MOVC instruction is used.					





## Electrical Characteristics

### Absolute Maximum Ratings

<p>C = commercial.....0°C to 70°C          I = industrial .....-40°C to 85°C          Storage Temperature ..... -65°C to + 150°C          Voltage on V<sub>CC</sub> to V<sub>SS</sub> (standard voltage) .....-0.5V to + 6.5V          Voltage on V<sub>CC</sub> to V<sub>SS</sub> (low voltage).....-0.5V to + 4.5V          Voltage on Any Pin to V<sub>SS</sub>.....-0.5V to V<sub>CC</sub> + 0.5V          Power Dissipation ..... 1 W</p>	<p>Note: 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.          Power dissipation value is based on the maximum allowable die temperature and the thermal resistance of the package.</p>
--	--

### DC Parameters for Standard Voltage

T<sub>A</sub> = 0°C to +70°C; V<sub>SS</sub> = 0V; V<sub>CC</sub> = 4.5V to 5.5V; F = 10 to 40 MHz  
 T<sub>A</sub> = -40°C to +85°C; V<sub>SS</sub> = 0V; V<sub>CC</sub> = 4.5V to 5.5V; F = 10 to 40 MHz

Symbol	Parameter	Min	Typ	Max	Unit	Test Conditions
V <sub>IL</sub>	Input Low Voltage	-0.5		0.2 V <sub>CC</sub> - 0.1	V	
V <sub>IH</sub>	Input High Voltage except RST, XTAL1	0.2 V <sub>CC</sub> + 0.9		V <sub>CC</sub> + 0.5	V	
V <sub>IH1</sub>	Input High Voltage RST, XTAL1	0.7 V <sub>CC</sub>		V <sub>CC</sub> + 0.5	V	
V <sub>OL</sub>	Output Low Voltage, ports 1, 2, 3, 4 <sup>(6)</sup>			0.3	V	I <sub>OL</sub> = 100 μA <sup>(4)</sup>
				0.45	V	I <sub>OL</sub> = 1.6 mA <sup>(4)</sup>
				1.0	V	I <sub>OL</sub> = 3.5 mA <sup>(4)</sup>
V <sub>OL1</sub>	Output Low Voltage, port 0, ALE, $\overline{\text{PSEN}}$ <sup>(6)</sup>			0.3	V	I <sub>OL</sub> = 200 μA <sup>(4)</sup>
				0.45	V	I <sub>OL</sub> = 3.2 mA <sup>(4)</sup>
				1.0	V	I <sub>OL</sub> = 7.0 mA <sup>(4)</sup>
V <sub>OH</sub>	Output High Voltage, ports 1, 2, 3, 4	V <sub>CC</sub> - 0.3			V	I <sub>OH</sub> = -10 μA
		V <sub>CC</sub> - 0.7			V	I <sub>OH</sub> = -30 μA
		V <sub>CC</sub> - 1.5			V	I <sub>OH</sub> = -60 μA V <sub>CC</sub> = 5V ± 10%
V <sub>OH1</sub>	Output High Voltage, port 0, ALE, $\overline{\text{PSEN}}$	V <sub>CC</sub> - 0.3			V	I <sub>OH</sub> = -200 μA
		V <sub>CC</sub> - 0.7			V	I <sub>OH</sub> = -3.2 mA
		V <sub>CC</sub> - 1.5			V	I <sub>OH</sub> = -7.0 mA V <sub>CC</sub> = 5V ± 10%
R <sub>RST</sub>	RST Pull-down Resistor	50	200 <sup>(5)</sup>	250	kΩ	
I <sub>IL</sub>	Logical 0 Input Current ports 1, 2, 3, 4 and 5			-50	μA	V <sub>IN</sub> = 0.45V
I <sub>LI</sub>	Input Leakage Current			±10	μA	0.45V < V <sub>IN</sub> < V <sub>CC</sub>
I <sub>TL</sub>	Logical 1 to 0 Transition Current, ports 1, 2, 3, 4			-650	μA	V <sub>IN</sub> = 2.0 V
C <sub>IO</sub>	Capacitance of I/O Buffer			10	pF	F <sub>C</sub> = 3 MHz T <sub>A</sub> = 25°C
I <sub>PD</sub>	Power-down Current		100	150	μA	4.5V < V <sub>CC</sub> < 5.5V <sup>(3)</sup>
I <sub>CCOP</sub>	Power Supply Current on normal mode			0.29 x Frequency (MHz) + 4	mA	V <sub>CC</sub> = 5.5V <sup>(1)</sup>
I <sub>CCIDLE</sub>	Power Supply Current on idle mode			0.16 x Frequency (MHz) + 4	mA	V <sub>CC</sub> = 5.5V <sup>(2)</sup>

## DC Parameters for Standard Voltage (2)

$T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ ;  $V_{SS} = 0\text{ V}$ ;  $V_{CC} = 2.7\text{V}$  to  $5.5\text{V}$ ;  $F = 10$  to  $40\text{ MHz}$

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

Symbol	Parameter	Min	Typ <sup>(5)</sup>	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, 4 and 5 <sup>(6)</sup>			0.45	V	$I_{OL} = 0.8\text{ mA}^{(4)}$
$V_{OL1}$	Output Low Voltage, port 0, ALE, $\overline{\text{PSEN}}$ <sup>(6)</sup>			0.45	V	$I_{OL} = 1.6\text{ mA}^{(4)}$
$V_{OH}$	Output High Voltage, ports 1, 2, 3, 4 and 5	$0.9 V_{CC}$			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, 3, 4 and 5			-50	$\mu\text{A}$	$V_{IN} = 0.45\text{V}$
$I_{LI}$	Input Leakage Current			$\pm 10$	$\mu\text{A}$	$0.45\text{V} < V_{IN} < V_{CC}$
$I_{TL}$	Logical 1 to 0 Transition Current, ports 1, 2, 3, 4 and 5			-650	$\mu\text{A}$	$V_{IN} = 2.0\text{V}$
$R_{RST}$	RST Pulldown Resistor	50	200	250	$\text{k}\Omega$	
CIO	Capacitance of I/O Buffer			10	pF	$F_C = 3\text{ MHz}$ $T_A = 25^\circ\text{C}$
$I_{PD}$	Power-down Current		120	150	$\mu\text{A}$	$V_{CC} = 2.7\text{V}$ to $5.5\text{V}^{(3)}$
$I_{CCOP}$	Power Supply Current on normal mode			$0.29 \times \text{Frequency (MHz)} + 4$	mA	$V_{CC} = 5.5\text{V}^{(1)}$
$I_{CCIDLE}$	Power Supply Current on idle mode			$0.16 \times \text{Frequency (MHz)} + 4$	mA	$V_{CC} = 5.5\text{V}^{(2)}$

## DC Parameters for Low Voltage

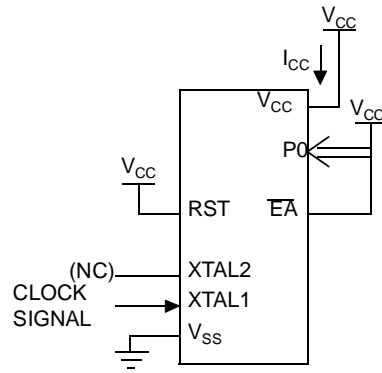
$T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ ;  $V_{SS} = 0\text{V}$ ;  $V_{CC} = 2.7\text{V}$  to  $3.6\text{V}$ ;  $F = 10$  to  $40$  MHz

$T_A = -40^\circ\text{C}$  to  $+85^\circ\text{C}$ ;  $V_{SS} = 0\text{V}$ ;  $V_{CC} = 2.7\text{V}$  to  $3.6\text{V}$ ;  $F = 10$  to  $40$  MHz

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 RST, XTAL1	$0.2 V_{CC} + 0.9$		$V_{CC} + 0.5$	V	
$V_{IH1}$	Input High Voltage, RST, XTAL1	$0.7 V_{CC}$		$V_{CC} + 0.5$	V	
$V_{OL}$	Output Low Voltage, ports 1, 2, 3, 4 <sup>(6)</sup>			0.45	V	$I_{OL} = 0.8 \text{ mA}^{(4)}$
$V_{OL1}$	Output Low Voltage, port 0, ALE, $\overline{\text{PSEN}}$ <sup>(6)</sup>			0.45	V	$I_{OL} = 1.6 \text{ mA}^{(4)}$
$V_{OH}$	Output High Voltage, ports 1, 2, 3, 4	$0.9 V_{CC}$			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, 3, 4			-50	$\mu\text{A}$	$V_{IN} = 0.45\text{V}$
$I_{LI}$	Input Leakage Current			$\pm 10$	$\mu\text{A}$	$0.45\text{V} < V_{IN} < V_{CC}$
$I_{TL}$	Logical 1 to 0 Transition Current, ports 1, 2, 3,			-650	$\mu\text{A}$	$V_{IN} = 2.0\text{V}$
$R_{RST}$	RST Pulldown Resistor	50	200 <sup>(5)</sup>	250	k $\Omega$	
CIO	Capacitance of I/O Buffer			10	pF	$F_c = 3 \text{ MHz}$ $T_A = 25^\circ\text{C}$
$I_{PD}$	Power-down Current		10 <sup>(5)</sup>	50	$\mu\text{A}$	$V_{CC} = 2.7\text{V}$ to $3.6\text{V}^{(3)}$
$I_{CCOP}$	Power Supply Current on normal mode			$0.31 \times \text{Frequency (MHz)} + 4$	mA	$V_{CC} = 3.6\text{V}^{(1)}$
$I_{CCIDLE}$	Power Supply Current on idle mode			$0.2 \times \text{Frequency (MHz)} + 4$	mA	$V_{CC} = 3.6\text{V}^{(2)}$

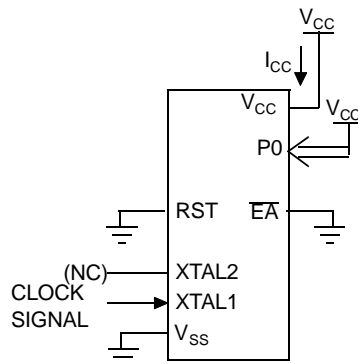
- Notes:
- Operating  $I_{CC}$  is measured with all output pins disconnected; XTAL1 driven with  $T_{CLCH}$ ,  $T_{CHCL} = 5 \text{ ns}$  (see Figure 26.),  
 $V_{IL} = V_{SS} + 0.5\text{V}$ ,  
 $V_{IH} = V_{CC} - 0.5\text{V}$ ; XTAL2 N.C.;  $\overline{\text{EA}} = \text{RST} = \text{Port 0} = V_{CC}$ .  $I_{CC}$  would be slightly higher if a crystal oscillator used (see Figure 23).
  - 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 =  $V_{CC}$ ;  $\overline{\text{EA}} = \text{RST} = V_{SS}$  (see Figure 24).
  - Power-down  $I_{CC}$  is measured with all output pins disconnected;  $\overline{\text{EA}} = V_{SS}$ , PORT 0 =  $V_{CC}$ ; XTAL2 NC.; RST =  $V_{SS}$  (see Figure 25).
  - 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.
  - Typical 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.

**Figure 23.**  $I_{CC}$  Test Condition, Active Mode



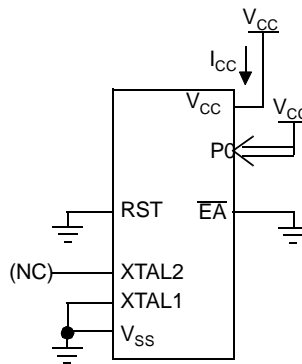
All other pins are disconnected.

**Figure 24.**  $I_{CC}$  Test Condition, Idle Mode



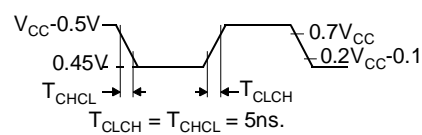
All other pins are disconnected.

**Figure 25.**  $I_{CC}$  Test Condition, Power-down Mode



All other pins are disconnected.

**Figure 26.** Clock Signal Waveform for  $I_{CC}$  Tests in Active and Idle Modes



## AC Parameters

### 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 PSEN Low.

(Load Capacitance for port 0, ALE and PSEN = 100 pF; Load Capacitance for all other outputs = 80 pF.)

Table 49 Table 52, and Table 54 give the description of each AC symbols.

Table 51, Table 53 and Table 55 give for each range the AC parameter.

Table 50, Table 51 and Table 56 gives the frequency derating formula of the AC parameter for each speed range description. To calculate each AC symbols. take the x value in the corresponding column (-M or -L) and use this value in the formula.

Example:  $T_{LLIU}$  for -M and 20 MHz, Standard clock.

$x = 35$  ns

$T = 50$  ns

$T_{CCIV} = 4T - x = 165$  ns

### External Program Memory Characteristics

**Table 49.** 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 $\overline{PSEN}$
$T_{PLPH}$	$\overline{PSEN}$ Pulse Width
$T_{PLIV}$	$\overline{PSEN}$ to Valid Instruction In
$T_{PXIX}$	Input Instruction Hold After $\overline{PSEN}$
$T_{PXIZ}$	Input Instruction FloatAfter $\overline{PSEN}$
$T_{AVIV}$	Address to Valid Instruction In
$T_{PLAZ}$	$\overline{PSEN}$ Low to Address Float

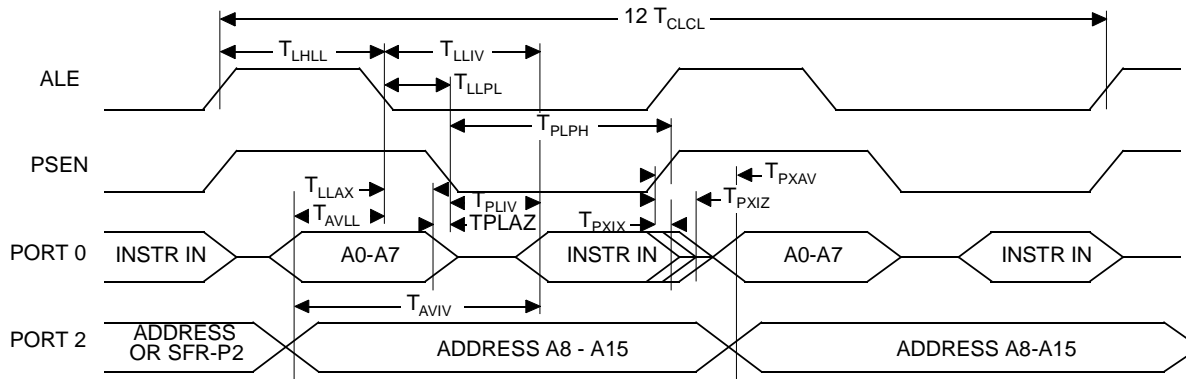
**Table 50.** AC Parameters for a Fix Clock

Symbol	-M		-L		Units
	Min	Max	Min	Max	
T	25		25		ns
T <sub>LHLL</sub>	35		35		ns
T <sub>AVLL</sub>	5		5		ns
T <sub>LLAX</sub>	5		5		ns
T <sub>LLIV</sub>		65		65	ns
T <sub>LLPL</sub>	5		5		ns
T <sub>PLPH</sub>	50		50		ns
T <sub>PLIV</sub>		30		30	ns
T <sub>PXIX</sub>	0		0		ns
T <sub>PXIZ</sub>		10		10	ns
T <sub>AVIV</sub>		80		80	ns
T <sub>PLAZ</sub>		10		10	ns

**Table 51.** AC Parameters for a Variable Clock

Symbol	Type	Standard Clock	X2 Clock	X Parameter for -M Range	X Parameter for -L Range	Units
T <sub>LHLL</sub>	Min	2 T - x	T - x	15	15	ns
T <sub>AVLL</sub>	Min	T - x	0.5 T - x	20	20	ns
T <sub>LLAX</sub>	Min	T - x	0.5 T - x	20	20	ns
T <sub>LLIV</sub>	Max	4 T - x	2 T - x	35	35	ns
T <sub>LLPL</sub>	Min	T - x	0.5 T - x	15	15	ns
T <sub>PLPH</sub>	Min	3 T - x	1.5 T - x	25	25	ns
T <sub>PLIV</sub>	Max	3 T - x	1.5 T - x	45	45	ns
T <sub>PXIX</sub>	Min	x	x	0	0	ns
T <sub>PXIZ</sub>	Max	T - x	0.5 T - x	15	15	ns
T <sub>AVIV</sub>	Max	5 T - x	2.5 T - x	45	45	ns
T <sub>PLAZ</sub>	Max	x	x	10	10	ns

## External Program Memory Read Cycle



## External Data Memory Characteristics

**Table 52.** 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

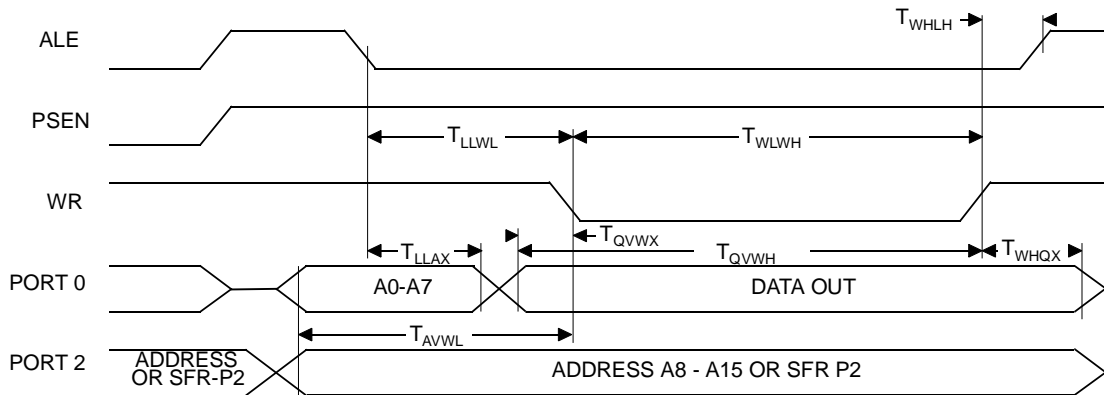
**Table 53.** AC Parameters for a Fix Clock

Symbol	-M		-L		Units
	Min	Max	Min	Max	
$T_{RLRH}$	125		125		ns
$T_{WLWH}$	125		125		ns
$T_{RLDV}$		95		95	ns
$T_{RHDX}$	0		0		ns
$T_{RHDZ}$		25		25	ns
$T_{LLDV}$		155		155	ns
$T_{AVDV}$		160		160	ns
$T_{LLWL}$	45	105	45	105	ns
$T_{AVWL}$	70		70		ns
$T_{QVWX}$	5		5		ns
$T_{QVWH}$	155		155		ns
$T_{WHQX}$	10		10		ns
$T_{RLAZ}$	0		0		ns
$T_{WHLH}$	5	45	5	45	ns

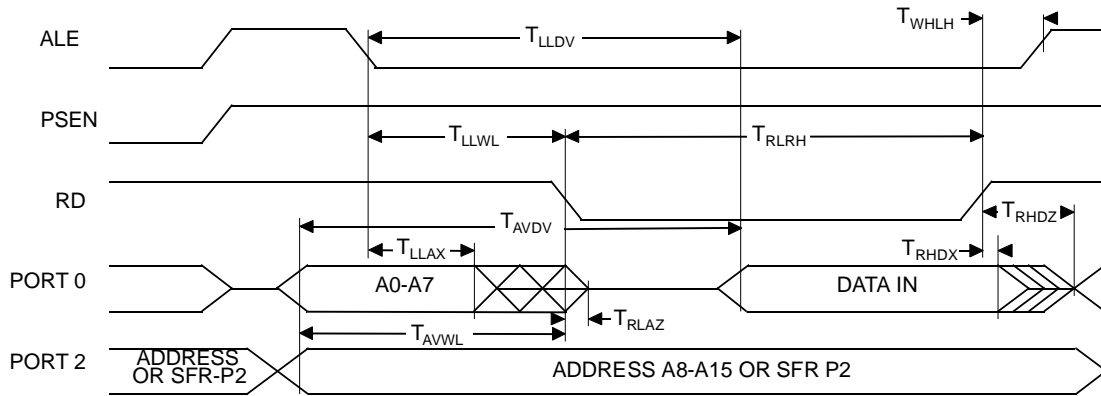


Symbol	Type	Standard Clock	X2 Clock	X parameter for - M range	X parameter for - L range	Units
$T_{RLRH}$	Min	6 T - x	3 T - x	25	25	ns
$T_{WLWH}$	Min	6 T - x	3 T - x	25	25	ns
$T_{RLDV}$	Max	5 T - x	2.5 T - x	30	30	ns
$T_{RHDX}$	Min	x	x	0	0	ns
$T_{RHDZ}$	Max	2 T - x	T - x	25	25	ns
$T_{LLDV}$	Max	8 T - x	4 T - x	45	45	ns
$T_{AVDV}$	Max	9 T - x	4.5 T - x	65	65	ns
$T_{LLWL}$	Min	3 T - x	1.5 T - x	30	30	ns
$T_{LLWL}$	Max	3 T + x	1.5 T + x	30	30	ns
$T_{AVWL}$	Min	4 T - x	2 T - x	30	30	ns
$T_{QVWX}$	Min	T - x	0.5 T - x	20	20	ns
$T_{QVWH}$	Min	7 T - x	3.5 T - x	20	20	ns
$T_{WHQX}$	Min	T - x	0.5 T - x	15	15	ns
$T_{RLAZ}$	Max	x	x	0	0	ns
$T_{WHLH}$	Min	T - x	0.5 T - x	20	20	ns
$T_{WHLH}$	Max	T + x	0.5 T + x	20	20	ns

### External Data Memory Write Cycle



## External Data Memory Read Cycle



## Serial Port Timing - Shift Register Mode

**Table 54.** Symbol Description

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

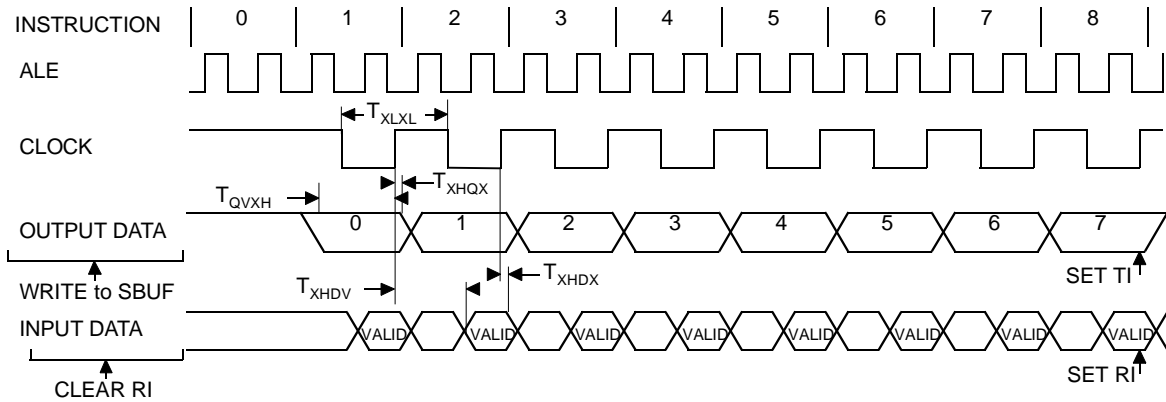
**Table 55.** AC Parameters for a Fix Clock

Symbol	-M		-L		Units
	Min	Max	Min	Max	
$T_{XLXL}$	300		300		ns
$T_{QVHX}$	200		200		ns
$T_{XHGX}$	30		30		ns
$T_{XHDX}$	0		0		ns
$T_{XHDV}$		117		117	ns

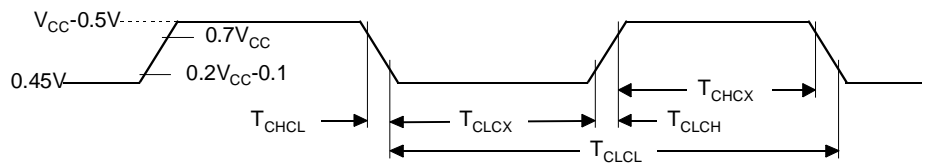
**Table 56.** AC Parameters for a Variable Clock

Symbol	Type	Standard Clock	X2 Clock	X Parameter for -M Range	X Parameter for -L Range	Units
$T_{XLXL}$	Min	12 T	6 T			ns
$T_{QVHX}$	Min	10 T - x	5 T - x	50	50	ns
$T_{XHGX}$	Min	2 T - x	T - x	20	20	ns
$T_{XHDX}$	Min	x	x	0	0	ns
$T_{XHDV}$	Max	10 T - x	5 T - x	133	133	ns

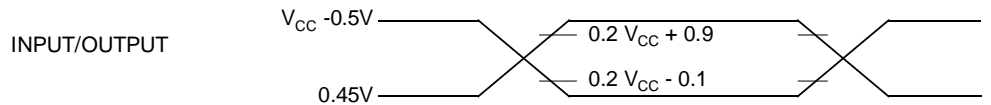
### Shift Register Timing Waveforms



### External Clock Drive Waveforms

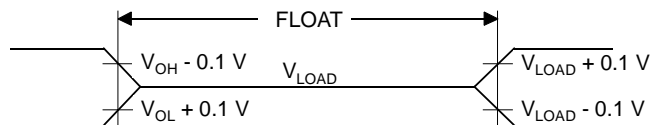


### AC Testing Input/Output Waveforms



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".

### Float Waveforms

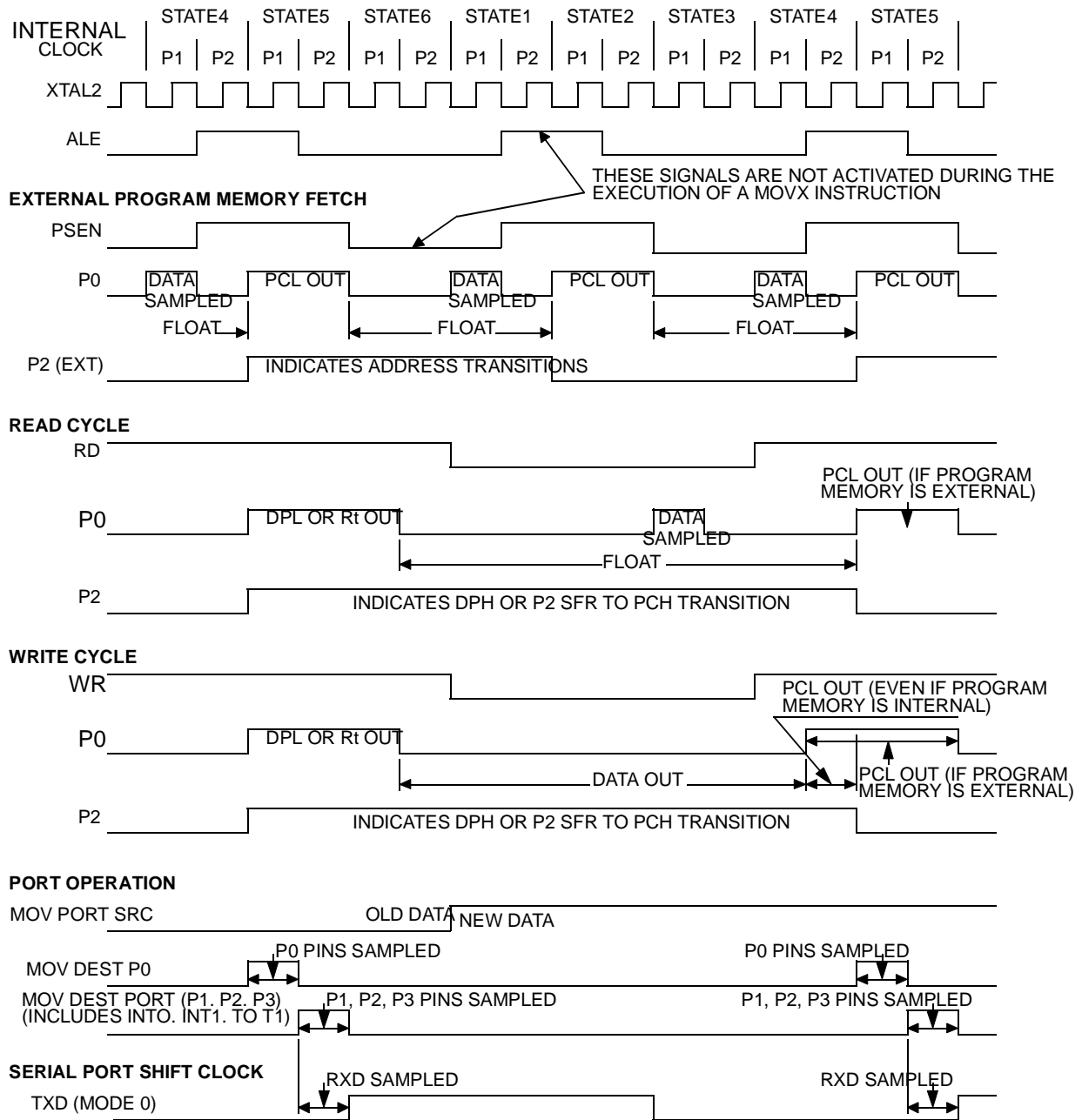


For timing purposes as port pin is no longer floating when a 100 mV changes from load voltage occurs and begins to float when a 100 mV change from the loaded  $V_{OH}/V_{OL}$  level occurs.  $I_{OL}/I_{OH} \geq \pm 20$  mA.

### Clock Waveforms

Valid in normal clock mode. In X2 mode XTAL2 must be changed to XTAL2/2.

**Figure 27. Internal Clock Signals**



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) RD and WR propagation delays are approximately 50 ns. The other signals are typically 85 ns. Propagation delays are incorporated in the AC specifications.

## Ordering Information

**Table 57.** Ordering Information

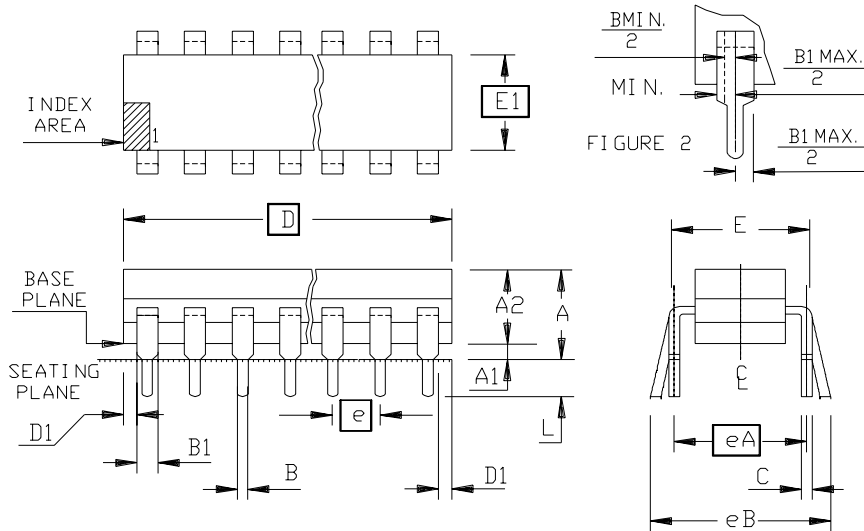
Part Number	Memory Size	Supply Voltage	Package	Temperature Range	Packing
AT80C51RD2-3CSCM	ROMLess	5V	PDIL40	Commercial	Stick
AT80C51RD2-3CSIM			PDIL40	Industrial	Stick
AT80C51RD2-SLSCM			PLCC44	Commercial	Stick
AT80C51RD2-SLSIM			PLCC44	Industrial	Stick
AT80C51RD2-RLTIM			VQFP44	Industrial	Tray
AT80C51RD2-SLSIL		3V	PLCC44	Industrial	Stick
AT80C51RD2-RLTIL			VQFP44	Industrial	Tray

Part Number	Memory Size	Supply Voltage	Package	Temperature Range	Packing
AT83C51RB2xxx-3CSCM	16K bytes	5V	PDIL40	Commercial	Stick
AT83C51RB2xxx-3CSIM			PDIL40	Industrial	Stick
AT83C51RB2xxx-SLSCM			PLCC44	Commercial	Stick
AT83C51RB2xxx-SLSIM			PLCC44	Industrial	Stick
AT83C51RB2xxx-RLTIM			VQFP44	Industrial	Tray
AT83C51RB2xxx-SLSIL		3V	PLCC44	Industrial	Stick
AT83C51RB2xxx-RLTIL			VQFP44	Industrial	Tray

Part Number	Memory Size	Supply Voltage	Package	Temperature Range	Packing
AT83C51RC2xxx-3CSCM	32K bytes	5V	PDIL40	Commercial	Stick
AT83C51RC2xxx-3CSIM			PDIL40	Industrial	Stick
AT83C51RC2xxx-SLSCM			PLCC44	Commercial	Stick
AT83C51RC2xxx-SLSIM			PLCC44	Industrial	Stick
AT83C51RC2xxx-RLTIM			VQFP44	Industrial	Tray
AT83C51RC2xxx-RLTIL		3V	VQFP44	Industrial	Tray
AT83C51RC2xxx-SLSIL			PLCC44	Industrial	Stick

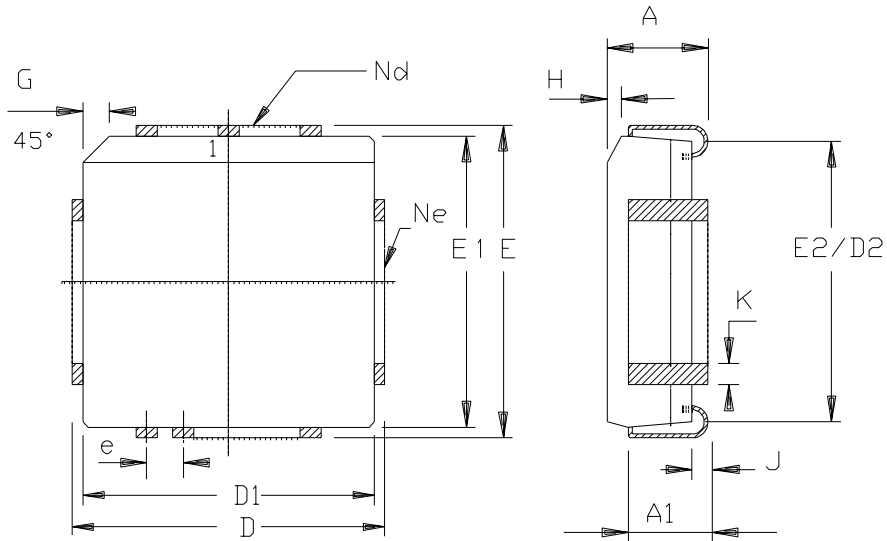
# Package Information

## PDIL40



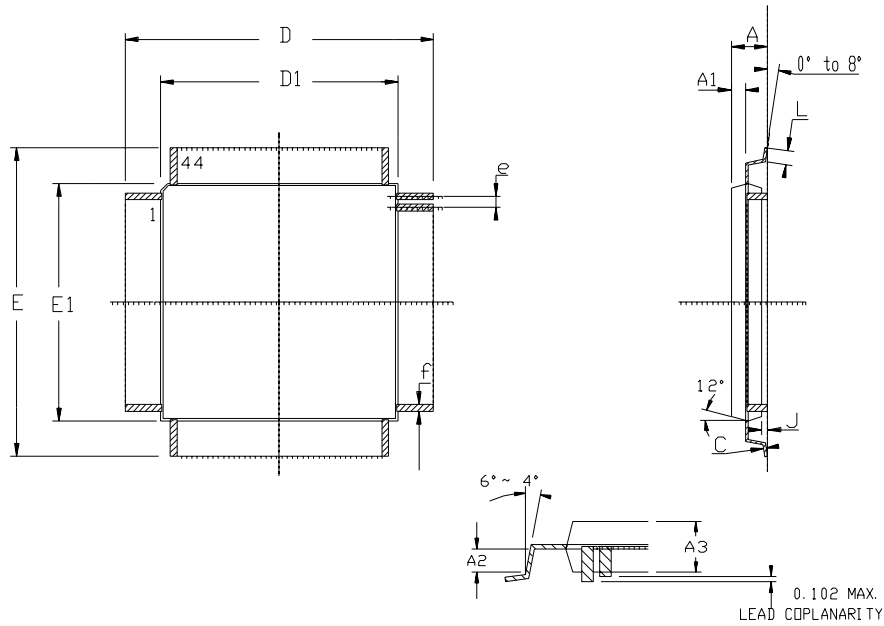
	MM		INCH	
A	-	5.08	-	.200
A1	0.38	-	.015	-
A2	3.18	4.95	.125	.195
B	0.36	0.56	.014	.022
B1	0.76	1.78	.030	.070
C	0.20	0.38	.008	.015
D	50.29	53.21	1.980	2.095
E	15.24	15.87	.600	.625
E1	12.32	14.73	.485	.580
e	2.54	B. S. C	.100	B. S. C
eA	15.24	B. S. C	.600	B. S. C
eB	-	17.78	-	.700
L	2.93	3.81	.115	.150
D1	0.13	-	.005	-
PKG STD	02			

PLCC44



	MM		INCH	
A	4.20	4.57	.165	.180
A1	2.29	3.04	.090	.120
D	17.40	17.65	.685	.695
D1	16.44	16.66	.647	.656
D2	14.99	16.00	.590	.630
E	17.40	17.65	.685	.695
E1	16.44	16.66	.647	.656
E2	14.99	16.00	.590	.630
e	1.27	BSC	.050	BSC
G	1.07	1.22	.042	.048
H	1.07	1.42	.042	.056
J	0.51	-	.020	-
K	0.33	0.53	.013	.021
Nd	11		11	
Ne	11		11	
PKG STD	00			

VQFP44



	MM		INCH	
	Min	Max	Min	Max
A	-	1.60	-	.063
A1	0.64 REF		.025 REF	
A2	0.64 REF		.025 REF	
A3	1.35	1.45	.053	.057
D	11.90	12.10	.468	.476
D1	9.90	10.10	.390	.398
E	11.90	12.10	.468	.476
E1	9.90	10.10	.390	.398
J	0.05	-	.002	-
L	0.45	0.75	.018	.030
e	0.80 BSC		.0315 BSC	
f	0.35 BSC		.014 BSC	





## Atmel Headquarters

### Corporate Headquarters

2325 Orchard Parkway  
San Jose, CA 95131  
TEL 1(408) 441-0311  
FAX 1(408) 487-2600

### Europe

Atmel Sarl  
Route des Arsenaux 41  
Case Postale 80  
CH-1705 Fribourg  
Switzerland  
TEL (41) 26-426-5555  
FAX (41) 26-426-5500

### Asia

Room 1219  
Chinachem Golden Plaza  
77 Mody Road Tsimhatsui  
East Kowloon  
Hong Kong  
TEL (852) 2721-9778  
FAX (852) 2722-1369

### Japan

9F, Tonetsu Shinkawa Bldg.  
1-24-8 Shinkawa  
Chuo-ku, Tokyo 104-0033  
Japan  
TEL (81) 3-3523-3551  
FAX (81) 3-3523-7581

## Atmel Operations

### Memory

2325 Orchard Parkway  
San Jose, CA 95131  
TEL 1(408) 441-0311  
FAX 1(408) 436-4314

### Microcontrollers

2325 Orchard Parkway  
San Jose, CA 95131  
TEL 1(408) 441-0311  
FAX 1(408) 436-4314

La Chantrerie  
BP 70602  
44306 Nantes Cedex 3, France  
TEL (33) 2-40-18-18-18  
FAX (33) 2-40-18-19-60

### ASIC/ASSP/Smart Cards

Zone Industrielle  
13106 Rousset Cedex, France  
TEL (33) 4-42-53-60-00  
FAX (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.  
Colorado Springs, CO 80906  
TEL 1(719) 576-3300  
FAX 1(719) 540-1759

Scottish Enterprise Technology Park  
Maxwell Building  
East Kilbride G75 0QR, Scotland  
TEL (44) 1355-803-000  
FAX (44) 1355-242-743

### RF/Automotive

Theresienstrasse 2  
Postfach 3535  
74025 Heilbronn, Germany  
TEL (49) 71-31-67-0  
FAX (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.  
Colorado Springs, CO 80906  
TEL 1(719) 576-3300  
FAX 1(719) 540-1759

### Biometrics/Imaging/Hi-Rel MPU/ High Speed Converters/RF Data- com

Avenue de Rochepleine  
BP 123  
38521 Saint-Egreve Cedex, France  
TEL (33) 4-76-58-30-00  
FAX (33) 4-76-58-34-80

---

### e-mail

[literature@atmel.com](mailto:literature@atmel.com)

### Web Site

<http://www.atmel.com>

### © Atmel Corporation 2002.

Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty which is detailed in Atmel's Terms and Conditions located on the Company's web site. The Company assumes no responsibility for any errors which may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property of Atmel are granted by the Company in connection with the sale of Atmel products, expressly or by implication. Atmel's products are not authorized for use as critical components in life support devices or systems.

ATMEL® is a registered trademark of Atmel.

Other terms and product names may be the trademarks of others.



Printed on recycled paper.