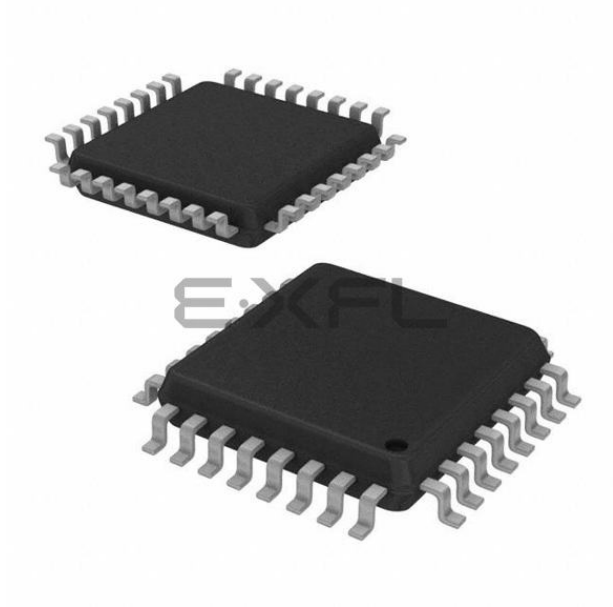


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	8051
Core Size	8-Bit
Speed	50MHz
Connectivity	SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	24
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2.25K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.25V
Data Converters	A/D 24x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f412-gqr

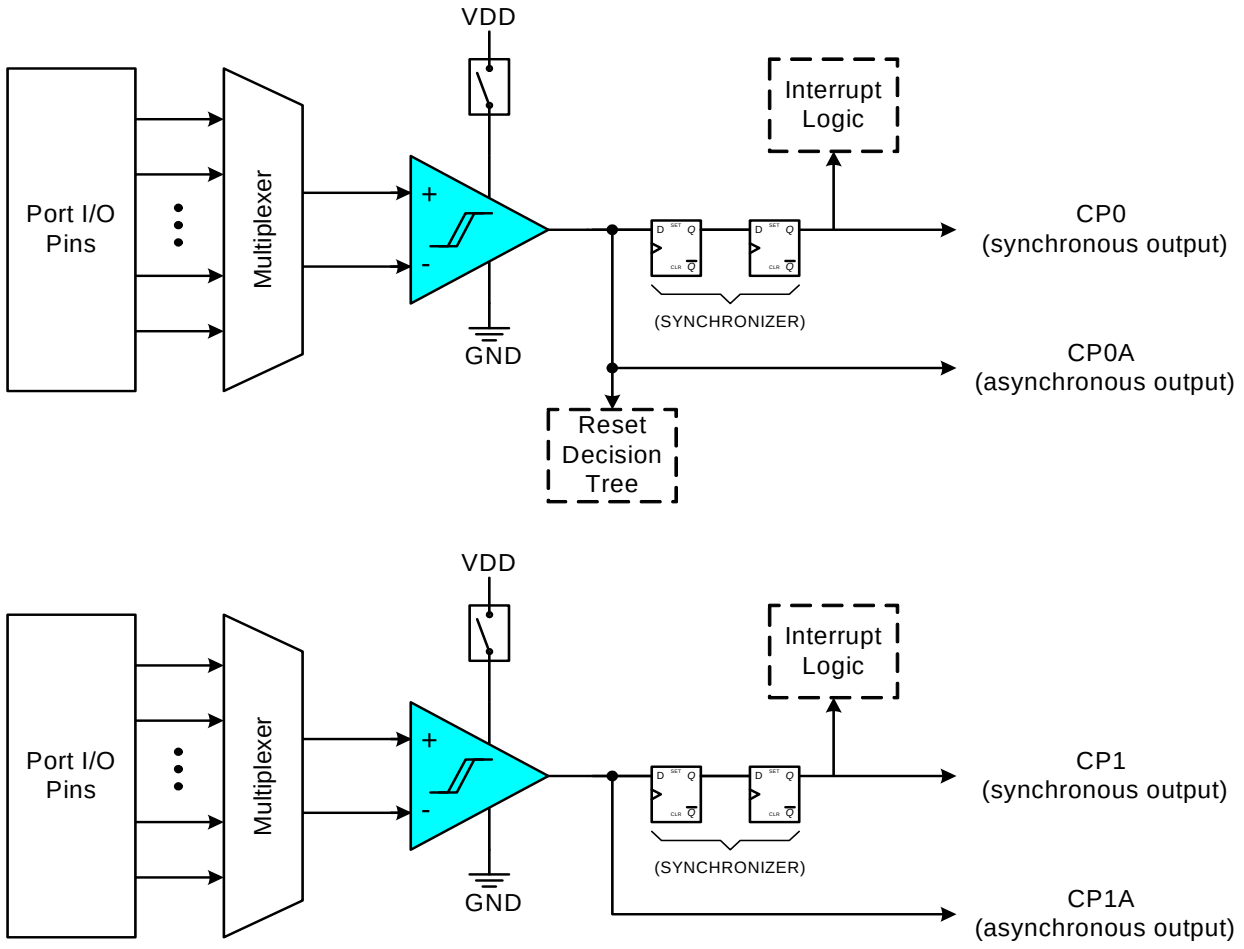


Figure 1.9. Comparators Block Diagram

1.8. Cyclic Redundancy Check Unit

C8051F41x devices include a cyclic redundancy check unit (CRC0) that can perform a CRC using a 16-bit or 32-bit polynomial. CRC0 accepts a stream of 8-bit data and outputs a 16-bit or 32-bit result. CRC0 also has a hardware bit reverse feature for quick data manipulation.

1.9. Voltage Regulator

C8051F41x devices include an on-chip low dropout voltage regulator (REG0). The input to REG0 at the V_{REGIN} pin can be as high as 5.25 V. The output can be selected by software to 2.0 V or 2.5 V. When enabled, the output of REG0 powers the device and drives the V_{DD} pin. The voltage regulator can be used to power external devices connected to V_{DD}.

1.10. Serial Ports

The C8051F41x Family includes an SMBus/I2C interface, a full-duplex UART with enhanced baud rate configuration, and an Enhanced SPI interface. Each of the serial buses is fully implemented in hardware and makes extensive use of the CIP-51's interrupts, thus requiring very little CPU intervention.

1.12. Port Input/Output

C8051F41x devices include up to 24 I/O pins. Port pins are organized as three byte-wide ports. The port pins behave like typical 8051 ports with a few enhancements. Each port pin can be configured as a digital or analog I/O pin. Pins selected as digital I/O can be configured for push-pull or open-drain operation. The "weak pullups" that are fixed on typical 8051 devices may be individually or globally disabled to save power.

The Digital Crossbar allows mapping of internal digital system resources to port I/O pins. On-chip counter/timers, serial buses, hardware interrupts, and other digital signals can be configured to appear on the port pins using the Crossbar control registers. This allows the user to select the exact mix of general-purpose port I/O, digital, and analog resources needed for the application.

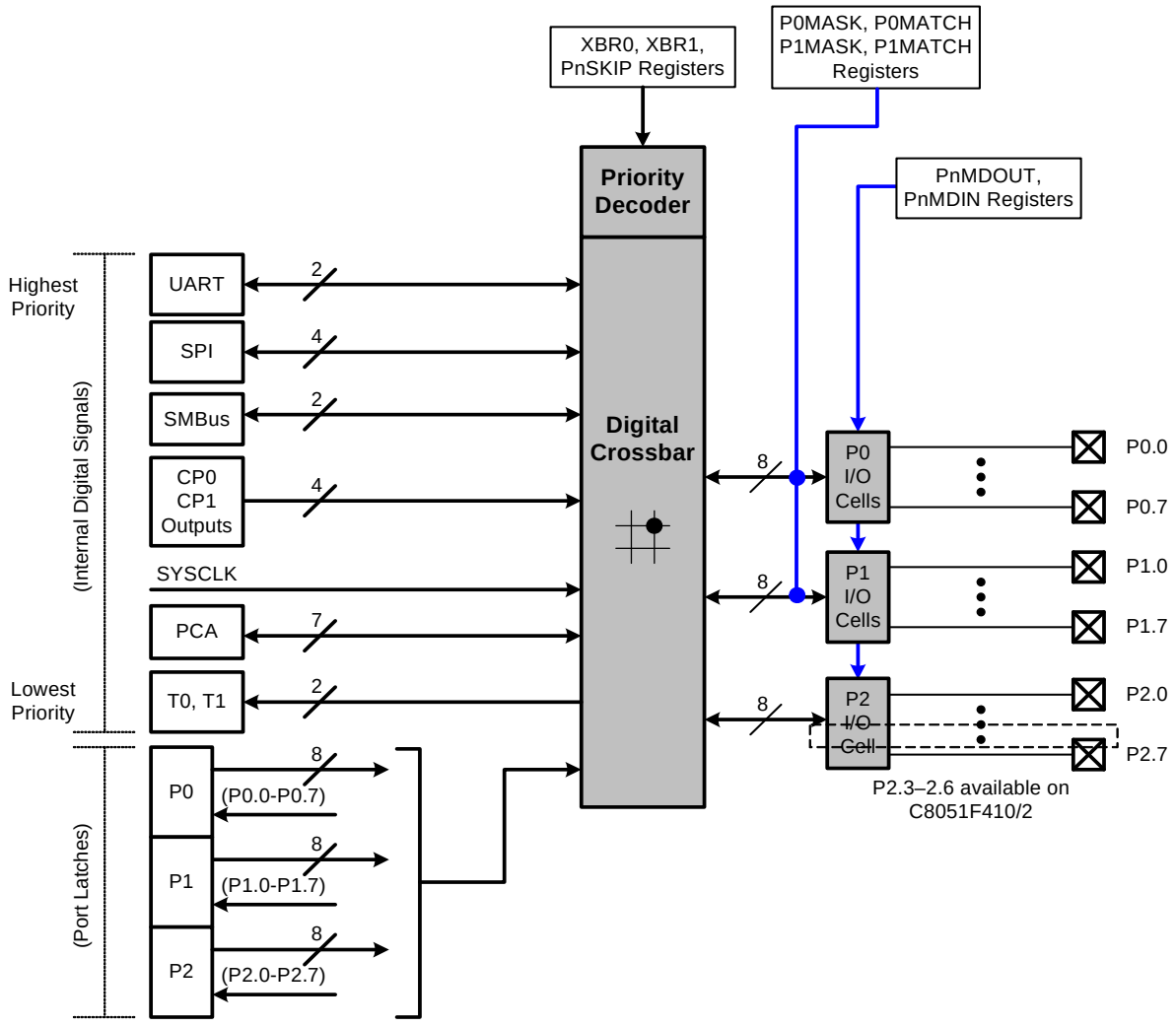


Figure 1.11. Port I/O Functional Block Diagram

1.13. Programmable Counter Array

The Programmable Counter Array (PCA0) provides enhanced timer functionality while requiring less CPU intervention than the standard 8051 counter/timers. The PCA consists of a dedicated 16-bit counter/timer and six 16-bit capture/compare modules. The counter/timer is driven by a programmable timebase that can select between seven sources: system clock, system clock divided by four, system clock divided by twelve, the external oscillator clock source divided by 8, real-time clock source divided by 8, Timer 0 overflow, or an external clock signal on the External Clock Input (ECI) pin.

Each capture/compare module may be configured to operate independently in one of six modes: Edge-Triggered Capture, Software Timer, High-Speed Output, Frequency Output, 8-Bit PWM, or 16-Bit PWM. Additionally, PCA Module 5 may be used as a watchdog timer (WDT), and is enabled in this mode following a system reset. The PCA Capture/Compare Module I/O and the External Clock Input may be routed to Port I/O using the digital crossbar.

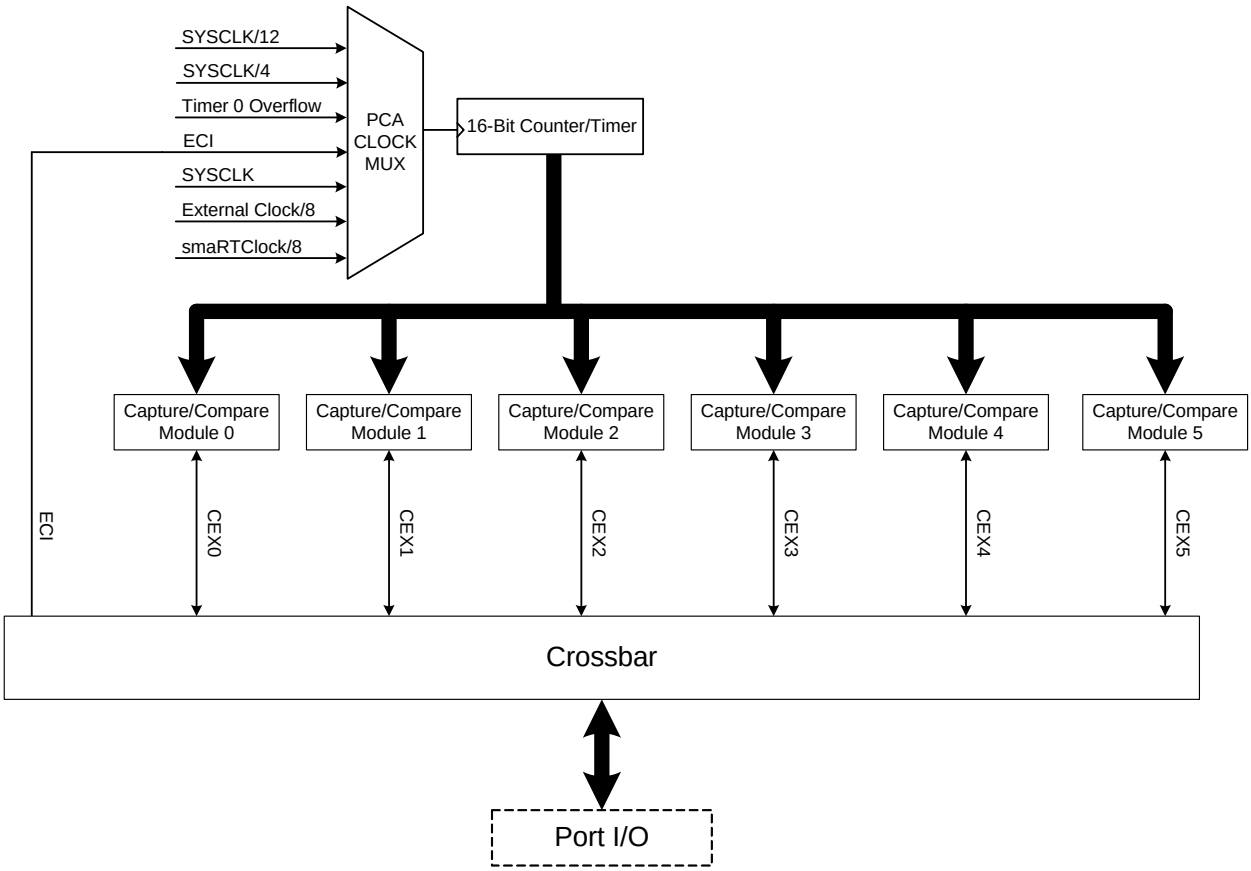


Figure 1.12. PCA Block Diagram

Table 3.2. Index to Electrical Characteristics Tables

Table Title	Page #
ADC0 Electrical Characteristics (V_{DD} = 2.5 V, V_{REF} = 2.2 V)	67
ADC0 Electrical Characteristics (V_{DD} = 2.1 V, V_{REF} = 1.5 V)	68
IDAC Electrical Characteristics	75
Voltage Reference Electrical Characteristics	79
Voltage Regulator Electrical Specifications	82
Comparator Electrical Characteristics	92
Reset Electrical Characteristics	134
Flash Electrical Characteristics	143
Port I/O DC Electrical Characteristics	163
Oscillator Electrical Characteristics	175

5.3.6. Settling Time Requirements

A minimum tracking time is required before an accurate conversion can be performed. This tracking time is determined by the AMUX0 resistance, the ADC0 sampling capacitance, any external source resistance, and the accuracy required for the conversion.

Figure 5.6 shows the equivalent ADC0 input circuit. The required ADC0 settling time for a given settling accuracy (SA) may be approximated by Equation 5.1. When measuring V_{DD} with respect to GND, R_{TOTAL} reduces to R_{MUX}. See Table 5.3 and Table 5.4 for ADC0 minimum settling time requirements.

$$t = \ln\left(\frac{2^n}{SA}\right) \times R_{TOTAL} C_{SAMPLE}$$

Equation 5.1. ADC0 Settling Time Requirements

Where:
SA is the settling accuracy, given as a fraction of an LSB (for example, 0.25 to settle within 1/4 LSB)
t is the required settling time in seconds
R_{TOTAL} is the sum of the AMUX0 resistance and any external source resistance.
n is the ADC resolution in bits (12).

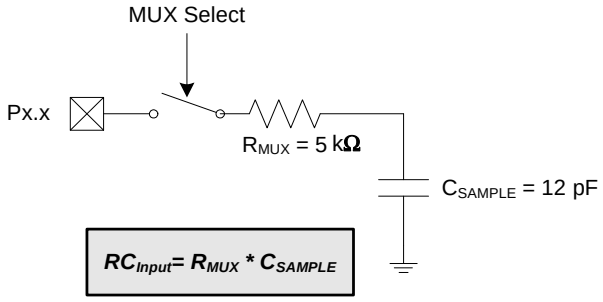


Figure 5.6. ADC0 Equivalent Input Circuits

NOTES:

The Comparator output can be polled in software, used as an interrupt source, internal oscillator suspend awakening source and/or routed to a Port pin. When routed to a Port pin, the Comparator output is available asynchronous or synchronous to the system clock; the asynchronous output is available even in STOP or SUSPEND mode (with no system clock active). When disabled, the Comparator output (if assigned to a Port I/O pin via the Crossbar) defaults to the logic low state, and its supply current falls to less than 100 nA. See **Section "18.1. Priority Crossbar Decoder" on page 149** for details on configuring Comparator outputs via the digital Crossbar. Comparator inputs can be externally driven from -0.25 V to (V_{DD}) + 0.25 V without damage or upset. The complete Comparator electrical specifications are given in Table 9.1.

The Comparator response time may be configured in software via the CPTnMD registers (see SFR Definition 9.3 and SFR Definition 9.5). Selecting a longer response time reduces the Comparator supply current. See Table 9.1 for complete timing and current consumption specifications.

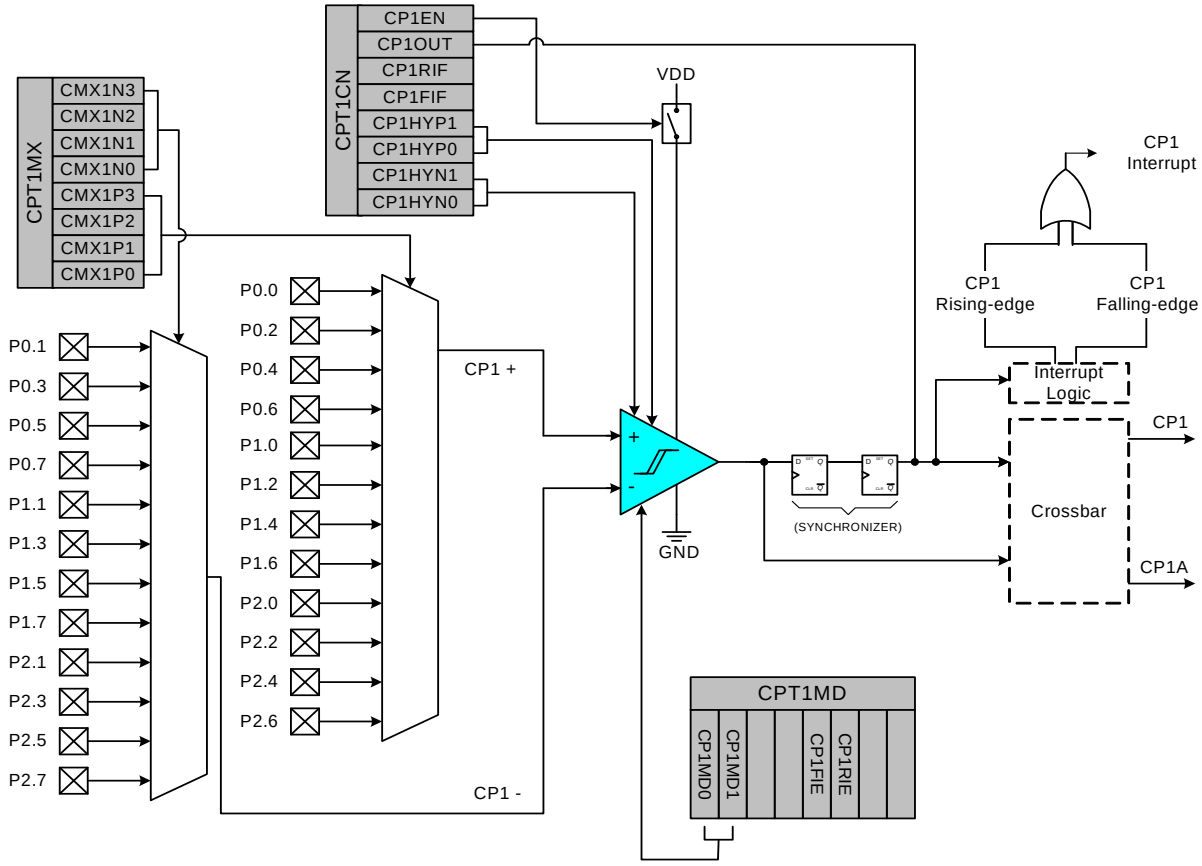


Figure 9.2. Comparator1 Functional Block Diagram

16.4.2. 16.4.2 PSWE Maintenance

7. Reduce the number of places in code where the PSWE bit (b0 in PSCTL) is set to a '1'. There should be exactly one routine in code that sets PSWE to a '1' to write Flash bytes and one routine in code that sets both PSWE and PSEE both to a '1' to erase Flash pages.
8. Minimize the number of variable accesses while PSWE is set to a '1'. Handle pointer address updates and loop maintenance outside the "PSWE = 1; ... PSWE = 0;" area. Code examples showing this can be found in AN201, "Writing to Flash from Firmware", available from the Silicon Laboratories web site.
9. Disable interrupts prior to setting PSWE to a '1' and leave them disabled until after PSWE has been reset to '0'. Any interrupts posted during the Flash write or erase operation will be serviced in priority order after the Flash operation has been completed and interrupts have been re-enabled by software.
10. Make certain that the Flash write and erase pointer variables are not located in XRAM. See your compiler documentation for instructions regarding how to explicitly locate variables in different memory areas.
11. Add address bounds checking to the routines that write or erase Flash memory to ensure that a routine called with an illegal address does not result in modification of the Flash.

16.4.3. System Clock

12. If operating from an external crystal, be advised that crystal performance is susceptible to electrical interference and is sensitive to layout and to changes in temperature. If the system is operating in an electrically noisy environment, use the internal oscillator or use an external CMOS clock.
13. If operating from the external oscillator, switch to the internal oscillator during Flash write or erase operations. The external oscillator can continue to run, and the CPU can switch back to the external oscillator after the Flash operation has completed.

NOTES:

19.2.3. External RC Example

If an RC network is used as an external oscillator source for the MCU, the circuit should be configured as shown in Figure 19.1, Option 2. The capacitor should be no greater than 100 pF; however for very small capacitors, the total capacitance may be dominated by parasitic capacitance in the PCB layout. To determine the required External Oscillator Frequency Control value (XFCN) in the OSCXCN Register, first select the RC network value to produce the desired frequency of oscillation. If the frequency desired is 100 kHz, let R = 246 kΩ and C = 50 pF:

f = 1.23(10^3) / RC = 1.23 (10^3) / [246 x 50] = 0.1 MHz = 100 kHz

Referring to the table in SFR Definition 19.3, the required XFCN setting is 010b. Programming XFCN to a higher setting in RC mode will improve frequency accuracy at a slightly increased external oscillator supply current.

19.2.4. External Capacitor Example

If a capacitor is used as an external oscillator for the MCU, the circuit should be configured as shown in Figure 19.1, Option 3. The capacitor should be no greater than 100 pF; however for very small capacitors, the total capacitance may be dominated by parasitic capacitance in the PCB layout. To determine the required External Oscillator Frequency Control value (XFCN) in the OSCXCN Register, select the frequency of oscillation and calculate the capacitance to be used from the equations below. Assume V_{DD} = 2.0 V and f = 75 kHz:

f = KF / (C x V_{DD})
0.075 MHz = KF / (C x 2.0)

Since the frequency of roughly 75 kHz is desired, select the K Factor from the table in SFR Definition 19.3 as KF = 7.7:

0.075 MHz = 7.7 / (C x 2.0)

C x 2.0 = 7.7 / 0.075 MHz

C = 102.6 / 2.0 pF = 51.3 pF

Therefore, the XFCN value to use in this example is 010b.

Internal Register Definition 20.5. RTC0XCN: smaRTClock Oscillator Control

R/W	R/W	R/W	R	R	R	R	R	Reset Value
AGCEN	XMODE	BIASX2	CLKVLD	-	-	-	VBATEN	Variable
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	smaRTClock Address:

Note: This register is not an SFR. It can only be accessed indirectly through RTC0ADR and RTC0DAT. 0x07

Bit 7: AGCEN: Crystal Oscillator Automatic Gain Control Enable Bit (Crystal Mode only).
0: Automatic Gain Control disabled.
1: Automatic Gain Control enabled.

Bit 6: XMODE: smaRTClock Mode Select Bit.
This bit selects whether smaRTClock will be used with or without a crystal.
0: smaRTClock is configured to Self-Oscillate Mode.
1: smaRTClock is configured to Crystal Mode.

Bit 5: BIASX2: smaRTClock Bias Double Enable Bit.
0: smaRTClock Bias Current Doubling is disabled.
1: smaRTClock Bias Current Doubling is enabled.

Bit 4: CLKVLD: smaRTClock Clock Valid Bit.
Set by hardware when the smaRTClock crystal oscillator is nearly stable. This bit always reads 1b when smaRTClock is used in Self-Oscillate Mode (XMODE = 0). This bit should be checked at least 1 ms after enabling the smaRTClock oscillator circuit and should not be used for an oscillator fail detect (use OSCFAIL in RTC0CN instead).

Bits 3–1: UNUSED. Read = 000b. Write = don't care.

Bit 0: VBATEN: smaRTClock V_{BAT} Indicator.
Note: This bit always reads 1b when smaRTClock is disabled (RTC0EN = 0).
For smaRTClock enabled (RTC0EN = 1):
0: smaRTClock is powered from V_{DD}.
1: smaRTClock is powered from the $V_{RTC\text{-}BACKUP}$ supply.

20.3. smaRTClock Timer and Alarm Function

The smaRTClock timer is a 47-bit counter that, when running (RTC0TR = 1), is incremented every RTC-CLK cycle. The timer has an alarm function that can be set to generate an interrupt, reset the MCU, or release the internal oscillator from Suspend Mode at a specific time.

20.3.1. Setting and Reading the smaRTClock Timer Value

The 47-bit smaRTClock timer can be set or read using the six CAPTUREn internal registers. Note that the timer does not need to be stopped before reading or setting its value. The following steps can be used to set the timer value:

Step 1. Write the desired 47-bit set value to the CAPTUREn registers (the LSB of CAPTURE0 is not used).
Step 2. Write '1' to RTC0SET. This will transfer the contents of the CAPTUREn registers to the timer.
Step 3. Operation is complete when RTC0SET is cleared to '0' by hardware.

The shift register contents are locked after the slave detects the first edge of SCK. Writes to SPI0DAT that occur after the first SCK edge will be held in the TX latch until the end of the current transfer.

When configured as a slave, SPI0 can be configured for 4-wire or 3-wire operation. The default, 4-wire slave mode, is active when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 1. In 4-wire mode, the NSS signal is routed to a port pin and configured as a digital input. SPI0 is enabled when NSS is logic 0, and disabled when NSS is logic 1. The bit counter is reset on a falling edge of NSS. Note that the NSS signal must be driven low at least 2 system clocks before the first active edge of SCK for each byte transfer. Figure 23.4 shows a connection diagram between two slave devices in 4-wire slave mode and a master device.

3-wire slave mode is active when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 0. NSS is not used in this mode, and is not mapped to an external port pin through the crossbar. Since there is not a way of uniquely addressing the device in 3-wire slave mode, SPI0 must be the only slave device present on the bus. It is important to note that in 3-wire slave mode there is no external means of resetting the bit counter that determines when a full byte has been received. The bit counter can only be reset by disabling and re-enabling SPI0 with the SPIEN bit. Figure 23.3 shows a connection diagram between a slave device in 3-wire slave mode and a master device.

23.4. SPI0 Interrupt Sources

When SPI0 interrupts are enabled, the following four flags will generate an interrupt when they are set to logic 1:

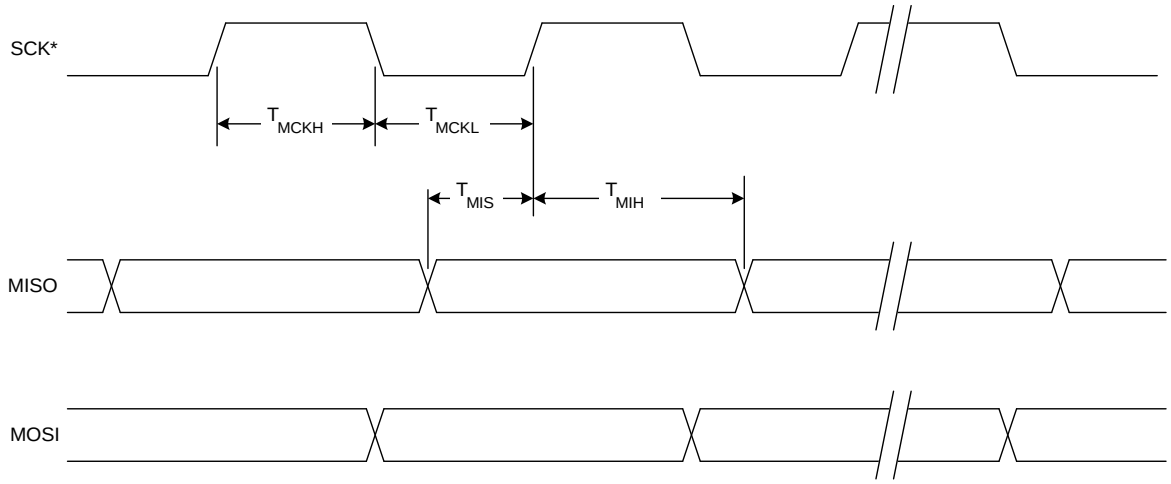
Note that all of the following interrupt bits must be cleared by software.

1. The SPI Interrupt Flag, SPIF (SPI0CN.7) is set to logic 1 at the end of each byte transfer. This flag can occur in all SPI0 modes.
2. The Write Collision Flag, WCOL (SPI0CN.6) is set to logic 1 if a write to SPI0DAT is attempted when the transmit buffer has not been emptied to the SPI shift register. When this occurs, the write to SPI0DAT will be ignored, and the transmit buffer will not be written.This flag can occur in all SPI0 modes.
3. The Mode Fault Flag MODF (SPI0CN.5) is set to logic 1 when SPI0 is configured as a master in multi-master mode and the NSS pin is pulled low. When a Mode Fault occurs, the MSTEN and SPIEN bits are set to logic 0 to disable SPI0 and allow another master device to access the bus.
4. The Receive Overrun Flag RXOVRN (SPI0CN.4) is set to logic 1 when configured as a slave, and a transfer is completed while the receive buffer still holds an unread byte from a previous transfer. The new byte is not transferred to the receive buffer, allowing the previously received data byte to be read. The data byte which caused the overrun is lost.

23.5. Serial Clock Timing

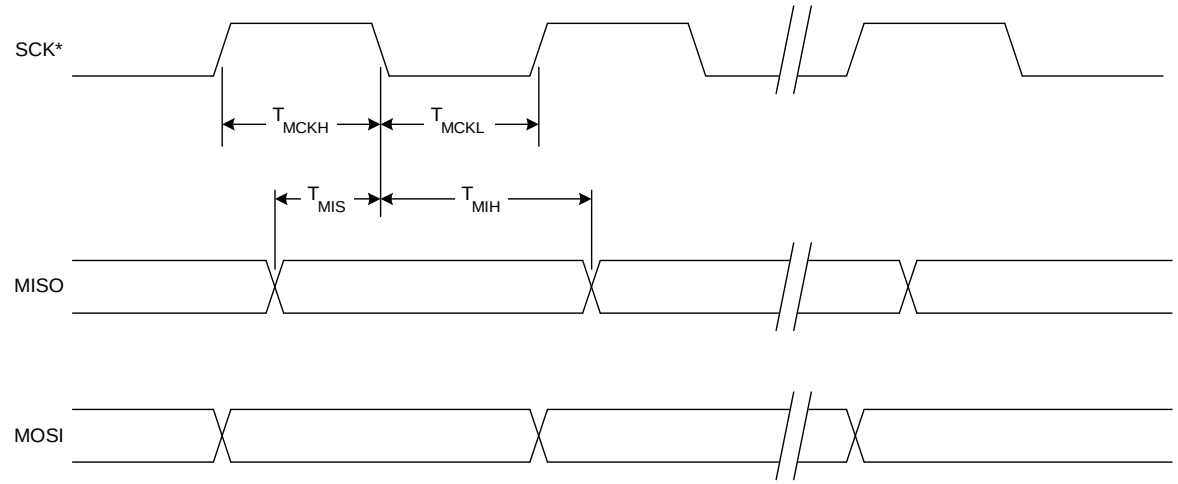
Four combinations of serial clock phase and polarity can be selected using the clock control bits in the SPI0 Configuration Register (SPI0CFG). The CKPHA bit (SPI0CFG.5) selects one of two clock phases (edge used to latch the data). The CKPOL bit (SPI0CFG.4) selects between a rising edge or a falling edge. Both master and slave devices must be configured to use the same clock phase and polarity. SPI0 should be disabled (by clearing the SPIEN bit, SPI0CN.0) when changing the clock phase or polarity. The clock and data line relationships are shown in Figure 23.5.

The SPI0 Clock Rate Register (SPI0CKR) as shown in SFR Definition 23.3 controls the master mode serial clock frequency. This register is ignored when operating in slave mode. When the SPI is configured as a master, the maximum data transfer rate (bits/sec) is one-half the system clock frequency or 12.5 MHz,



* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.

Figure 23.6. SPI Master Timing (CKPHA = 0)



* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.

Figure 23.7. SPI Master Timing (CKPHA = 1)

24.1.4. Mode 3: Two 8-bit Counter/Timers (Timer 0 Only)

In Mode 3, Timer 0 is configured as two separate 8-bit counter/timers held in TL0 and TH0. The counter/timer in TL0 is controlled using the Timer 0 control/status bits in TCON and TMOD: TR0, C/T0, GATE0 and TF0. TL0 can use either the system clock or an external input signal as its timebase. The TH0 register is restricted to a timer function sourced by the system clock or prescaled clock. TH0 is enabled using the Timer 1 run control bit TR1. TH0 sets the Timer 1 overflow flag TF1 on overflow and thus controls the Timer 1 interrupt.

Timer 1 is inactive in Mode 3. When Timer 0 is operating in Mode 3, Timer 1 can be operated in Modes 0, 1 or 2, but cannot be clocked by external signals nor set the TF1 flag and generate an interrupt. However, the Timer 1 overflow can be used to generate baud rates for the SMBus and UART. While Timer 0 is operating in Mode 3, Timer 1 run control is handled through its mode settings. To run Timer 1 while Timer 0 is in Mode 3, set the Timer 1 Mode as 0, 1, or 2. To disable Timer 1, configure it for Mode 3.

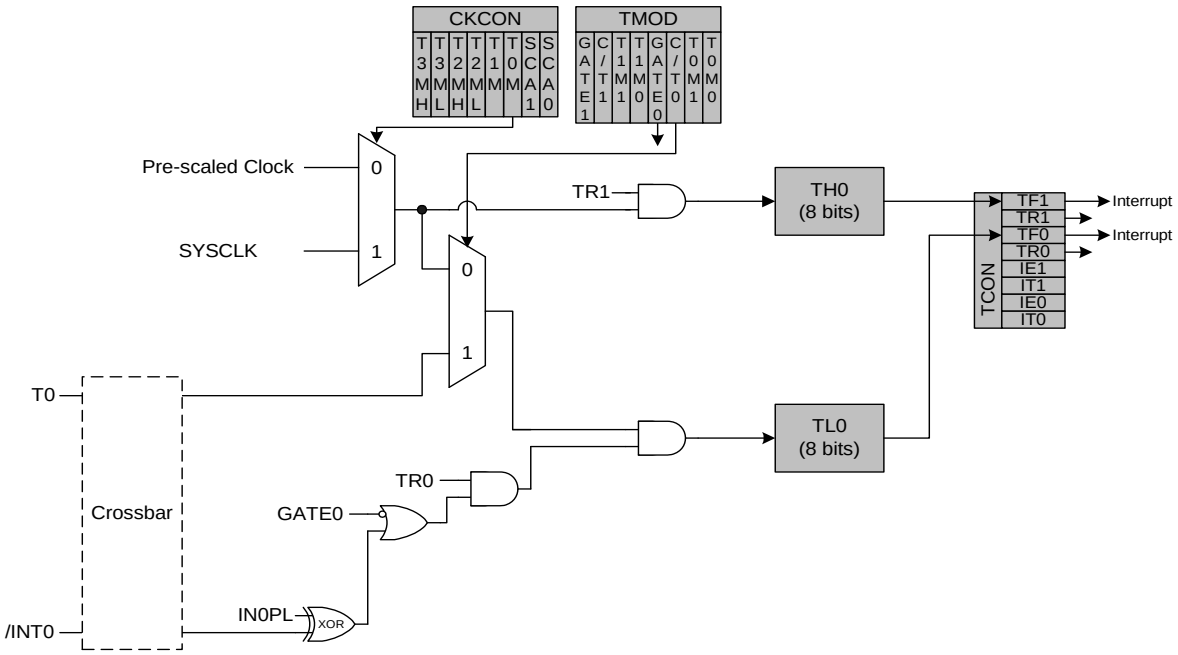


Figure 24.3. T0 Mode 3 Block Diagram

SFR Definition 24.3. CKCON: Clock Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
T3MH	T3ML	T2MH	T2ML	T1M	T0M	SCA1	SCA0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0x8E

Bit7: T3MH: Timer 3 High Byte Clock Select.
This bit selects the clock supplied to the Timer 3 high byte if Timer 3 is configured in split 8-bit timer mode. T3MH is ignored if Timer 3 is in any other mode.
0: Timer 3 high byte uses the clock defined by the T3XCLK bit in TMR3CN.
1: Timer 3 high byte uses the system clock.

Bit6: T3ML: Timer 3 Low Byte Clock Select.
This bit selects the clock supplied to Timer 3. If Timer 3 is configured in split 8-bit timer mode, this bit selects the clock supplied to the lower 8-bit timer.
0: Timer 3 low byte uses the clock defined by the T3XCLK bit in TMR3CN.
1: Timer 3 low byte uses the system clock.

Bit5: T2MH: Timer 2 High Byte Clock Select.
This bit selects the clock supplied to the Timer 2 high byte if Timer 2 is configured in split 8-bit timer mode. T2MH is ignored if Timer 2 is in any other mode.
0: Timer 2 high byte uses the clock defined by the T2XCLK bit in TMR2CN.
1: Timer 2 high byte uses the system clock.

Bit4: T2ML: Timer 2 Low Byte Clock Select.
This bit selects the clock supplied to Timer 2. If Timer 2 is configured in split 8-bit timer mode, this bit selects the clock supplied to the lower 8-bit timer.
0: Timer 2 low byte uses the clock defined by the T2XCLK bit in TMR2CN.
1: Timer 2 low byte uses the system clock.

Bit3: T1M: Timer 1 Clock Select.
This select the clock source supplied to Timer 1. T1M is ignored when C/T1 is set to logic 1.
0: Timer 1 uses the clock defined by the prescale bits, SCA1-SCA0.
1: Timer 1 uses the system clock.

Bit2: T0M: Timer 0 Clock Select.
This bit selects the clock source supplied to Timer 0. T0M is ignored when C/T0 is set to logic 1.
0: Counter/Timer 0 uses the clock defined by the prescale bits, SCA1-SCA0.
1: Counter/Timer 0 uses the system clock.

Bits1–0: SCA1–SCA0: Timer 0/1 Prescale Bits.
These bits control the division of the clock supplied to Timer 0 and Timer 1 if configured to use prescaled clock inputs.

SCA1	SCA0	Prescaled Clock
0	0	System clock divided by 12
0	1	System clock divided by 4
1	0	System clock divided by 48
1	1	External clock divided by 8

Note: External clock divided by 8 is synchronized with the system clock.

SFR Definition 24.9. TMR2RLL: Timer 2 Reload Register Low Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xCA

Bits 7–0: TMR2RLL: Timer 2 Reload Register Low Byte.
TMR2RLL holds the low byte of the reload value for Timer 2.

SFR Definition 24.10. TMR2RLH: Timer 2 Reload Register High Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xCB

Bits 7–0: TMR2RLH: Timer 2 Reload Register High Byte.
The TMR2RLH holds the high byte of the reload value for Timer 2.

SFR Definition 24.11. TMR2L: Timer 2 Low Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xCC

Bits 7–0: TMR2L: Timer 2 Low Byte.
In 16-bit mode, the TMR2L register contains the low byte of the 16-bit Timer 2. In 8-bit mode, TMR2L contains the 8-bit low byte timer value.

SFR Definition 24.12. TMR2H Timer 2 High Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xCD

Bits 7–0: TMR2H: Timer 2 High Byte.
In 16-bit mode, the TMR2H register contains the high byte of the 16-bit Timer 2. In 8-bit mode, TMR2H contains the 8-bit high byte timer value.

SFR Definition 24.13. TMR3CN: Timer 3 Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
TF3H	TF3L	TF3LEN	TF3CEN	T3SPLIT	TR3	T3RCLK	T3XCLK	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0x91

Bit7: TF3H: Timer 3 High Byte Overflow Flag.
Set by hardware when the Timer 3 high byte overflows from 0xFF to 0x00. In 16 bit mode, this will occur when Timer 3 overflows from 0xFFFF to 0x0000. When the Timer 3 interrupt is enabled, setting this bit causes the CPU to vector to the Timer 3 interrupt service routine. TF3H is not automatically cleared by hardware and must be cleared by software.

Bit6: TF3L: Timer 3 Low Byte Overflow Flag.
Set by hardware when the Timer 3 low byte overflows from 0xFF to 0x00. When this bit is set, an interrupt will be generated if TF3LEN is set and Timer 3 interrupts are enabled. TF3L will set when the low byte overflows regardless of the Timer 3 mode. This bit is not automatically cleared by hardware.

Bit5: TF3LEN: Timer 3 Low Byte Interrupt Enable.
This bit enables/disables Timer 3 Low Byte interrupts. If TF3LEN is set and Timer 3 interrupts are enabled, an interrupt will be generated when the low byte of Timer 3 overflows. This bit should be cleared when operating Timer 3 in 16-bit mode.
0: Timer 3 Low Byte interrupts disabled.
1: Timer 3 Low Byte interrupts enabled.

Bit4: TF3CEN: Timer 3 Capture Enable.
0: Timer 3 capture mode disabled.
1: Timer 3 capture mode enabled.

Bit3: T3SPLIT: Timer 3 Split Mode Enable.
When this bit is set, Timer 3 operates as two 8-bit timers with auto-reload.
0: Timer 3 operates in 16-bit auto-reload mode.
1: Timer 3 operates as two 8-bit auto-reload timers.

Bit2: TR3: Timer 3 Run Control.
This bit enables/disables Timer 3. In 8-bit mode, this bit enables/disables TMR3H only; TMR3L is always enabled in this mode.
0: Timer 3 disabled.
1: Timer 3 enabled.

Bit1: T3RCLK: Timer 3 Capture Mode.
This bit controls the Timer 3 capture source when TF3CEN=1. If T3XCLK = 1 and T3ML (CKCON.6) = 0, this bit also controls the clock source for Timer 3.
0: Capture every smaRTClock clock/8. If T3XCLK = 1 and T3ML (CKCON.6) = 0, count at external oscillator/8.
1: Capture every external oscillator/8. If T3XCLK = 1 and T3ML (CKCON.6) = 0, count at smaRTClock clock/8.

Bit0: T3XCLK: Timer 3 External Clock Select.
This bit selects the external clock source for Timer 3. If Timer 3 is in 8-bit mode, this bit selects the external oscillator clock source for both timer bytes. However, the Timer 3 Clock Select bits (T3MH and T3ML in register CKCON) may still be used to select between the external clock and the system clock for either timer.
0: Timer 3 external clock selection is the system clock divided by 12.
1: Timer 3 external clock uses the clock defined by the T3RCLK bit.

SFR Definition 24.14. TMR3RLL: Timer 3 Reload Register Low Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0x92

Bits 7-0: TMR3RLL: Timer 3 Reload Register Low Byte.
TMR3RLL holds the low byte of the reload value for Timer 3.

SFR Definition 24.15. TMR3RLH: Timer 3 Reload Register High Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0x93

Bits 7-0: TMR3RLH: Timer 3 Reload Register High Byte.
The TMR3RLH holds the high byte of the reload value for Timer 3.

SFR Definition 24.16. TMR3L: Timer 3 Low Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0x94

Bits 7-0: TMR3L: Timer 3 Low Byte.
In 16-bit mode, the TMR3L register contains the low byte of the 16-bit Timer 3. In 8-bit mode, TMR3L contains the 8-bit low byte timer value.

SFR Definition 24.17. TMR3H Timer 3 High Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0x95

Bits 7-0: TMR3H: Timer 3 High Byte.
In 16-bit mode, the TMR3H register contains the high byte of the 16-bit Timer 3. In 8-bit mode, TMR3H contains the 8-bit high byte timer value.

SFR Definition 25.4. PCA0L: PCA Counter/Timer Low Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xF9

Bits 7–0: PCA0L: PCA Counter/Timer Low Byte.
The PCA0L register holds the low byte (LSB) of the 16-bit PCA Counter/Timer.

SFR Definition 25.5. PCA0H: PCA Counter/Timer High Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:

SFR Address: 0xFA

Bits 7–0: PCA0H: PCA Counter/Timer High Byte.
The PCA0H register holds the high byte (MSB) of the 16-bit PCA Counter/Timer.

SFR Definition 25.6. PCA0CPLn: PCA Capture Module Low Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: PCA0CPL0: 0xFB, PCA0CPL1: 0xE9, PCA0CPL2: 0xEB, PCA0CPL3: 0xED, PCA0CPL4: 0xFD, PCA0CPL5: 0xD2

Bits7–0: PCA0CPLn: PCA Capture Module Low Byte.
The PCA0CPLn register holds the low byte (LSB) of the 16-bit capture module n.

SFR Definition 25.7. PCA0CPHn: PCA Capture Module High Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: PCA0CPH0: 0xFC, PCA0CPH1: 0xEA, PCA0CPH2: 0xEC, PCA0CPH3: 0xEE, PCA0CPH4: 0xFE, PCA0CPH5: 0xD3

Bits7–0: PCA0CPHn: PCA Capture Module High Byte.
The PCA0CPHn register holds the high byte (MSB) of the 16-bit capture module n.