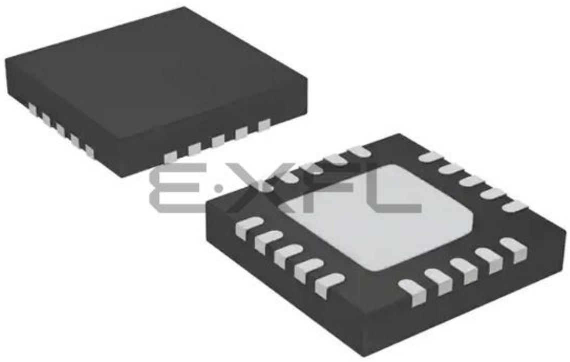


Welcome to E-XFL.COM

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	25MHz
Connectivity	LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	16
Program Memory Size	2KB (2K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.25V
Data Converters	A/D 16x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	20-VFQFN Exposed Pad
Supplier Device Package	20-QFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f536-c-im

Table 2.8. Reset Electrical Characteristics
–40 to +125 °C unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
$\overline{RST}$ Output Low Voltage	I_{OL} = 8.5 mA, V_{DD} = 2.1 V	—	—	0.8	V
$\overline{RST}$ Input High Voltage		0.7 x V_{REGIN}	—	—	V
$\overline{RST}$ Input Low Voltage		—	—	0.3 x V_{REGIN}	V
$\overline{RST}$ Input Pullup Impedance	V_{REGIN} = 1.8 V V_{REGIN} = 2.7 V V_{REGIN} = 3.3 V V_{REGIN} = 5 V	— — — —	330 160 130 80	— — — —	kΩ kΩ kΩ kΩ
Missing Clock Detector Timeout	Time from last system clock rising edge to reset initiation	100	350	650	µs
Reset Time Delay ($T_{PORDelay}$)[1]	Delay between release of any reset source and code execution at location 0x0000	—	—	350	µs
Minimum $\overline{RST}$ Low Time to Generate a System Reset		10	—	—	µs
V_{DD} Monitor (VDDMON0)					
Low Threshold ($V_{RST-LOW}$)[1,2,3]	C8051F52x/53x C8051F52xA/53xA C8051F52x-C/53x-C	1.8 1.65 1.65	1.9 1.75 1.75	2.0 1.8 1.8	V V V
High Threshold ($V_{RST-HIGH}$)[3]	C8051F52x/53x C8051F52xA/53xA C8051F52x-C/53x-C	2.1 2.25 2.25	2.2 2.3 2.3	2.3 2.4 2.45	V V V
Turn-on Time		—	83	—	µs
Supply Current	V_{DD} = 2.1 V	—	1	2	µA
Level-Sensitive V_{DD} Monitor (VDDMON1)[1]					
Threshold (V_{RST1})[1,2,3]	C8051F52x-C/53x-C	1.6	1.75	1.9	V
Supply Current	C8051F52x-C/53x-C	—	3	6	µA

Notes:
1. Refer to Section “20. Device Specific Behavior” on page 210.
2. The POR threshold (V_{RST}) is $V_{RST-LOW}$ or V_{RST1}, whichever is higher.
3. The V_{RST} threshold for power fail / brownout is the higher of VDDMON0 and VDDMON1 thresholds, if both are enabled.

Table 3.1. Pin Definitions for the C8051F52x and C8051F52xA (DFN 10)

Name	Pin Numbers		Type	Description
	‘F52xA ‘F52x-C	‘F52x		
$\overline{RST}$/ C2CK	1	1	D I/O D I/O	Device Reset. Open-drain output of internal POR or V_{DD} monitor. An external source can initiate a system reset by driving this pin low for at least the minimum $\overline{RST}$ low time to generate a system reset, as defined in Table 2.8 on page 32. A 1 kΩ pullup to V_{REGIN} is recommended. See Reset Sources Section for a complete description. Clock signal for the C2 Debug Interface.
P0.0/ V_{REF}	2	2	D I/O or A In A O or D In	Port 0.0. See Port I/O Section for a complete description. External V_{REF} Input. See V_{REF} Section.
GND	3	3		Ground.
V_{DD}	4	4		Core Supply Voltage.
V_{REGIN}	5	5		On-Chip Voltage Regulator Input.
P0.5/RX*/ CNVSTR	6	—	D I/O or A In D In	Port 0.5. See Port I/O Section for a complete description. External Converter start input for the ADC0, see Section “4. 12-Bit ADC (ADC0)” on page 52 for a complete description.
P0.5/ CNVSTR	—	6	D I/O or A In D In	Port 0.5. See Port I/O Section for a complete description. External Converter start input for the ADC0, see Section “4. 12-Bit ADC (ADC0)” on page 52 for a complete description.
P0.4/TX*	7	—	D I/O or A In	Port 0.4. See Port I/O Section for a complete description.
P0.4/RX*	—	7	D I/O or A In	Port 0.4. See Port I/O Section for a complete description.
P0.3 XTAL2	8	—	D I/O or A In D I/O	Port 0.3. See Port I/O Section for a complete description. External Clock Output. For an external crystal or resonator, this pin is the excitation driver. This pin is the external clock input for CMOS, capacitor, or RC oscillator configurations. See Section “14. Oscillators” on page 135.

Note: Please refer to Section “20. Device Specific Behavior” on page 210.

Table 3.4. Pin Definitions for the C8051F53x and C805153xA (TSSOP 20) (Continued)

Name	Pin Numbers		Type	Description
	'F53xA 'F53x-C	'F53x		
P0.4/TX*	19	—	D I/O or A In	Port 0.4. See Port I/O Section for a complete description.
P0.4/RX*	—	19	D I/O or A In	Port 0.4. See Port I/O Section for a complete description.
P0.3	20	—	D I/O or A In	Port 0.3. See Port I/O Section for a complete description.
P0.3/TX*	—	20	D I/O or A In	Port 0.3. See Port I/O Section for a complete description.

***Note:** Please refer to Section “20. Device Specific Behavior” on page 210.

Post-Tracking Mode is selected when AD0TM is set to 01b. A programmable tracking time based on AD0TK is started immediately following the convert start signal. Conversions are started after the programmed tracking time ends. After a conversion is complete, ADC0 does not track the input. Rather, the sampling capacitor remains disconnected from the input making the input pin high-impedance until the next convert start signal.

Dual-Tracking Mode is selected when AD0TM is set to 11b. A programmable tracking time based on AD0TK is started immediately following the convert start signal. Conversions are started after the programmed tracking time ends. After a conversion is complete, ADC0 tracks continuously until the next conversion is started.

Depending on the output connected to the ADC input, additional tracking time, more than is specified in Table 2.3 on page 28, may be required after changing MUX settings. See the settling time requirements described in Section “4.3.6. Settling Time Requirements” on page 60.

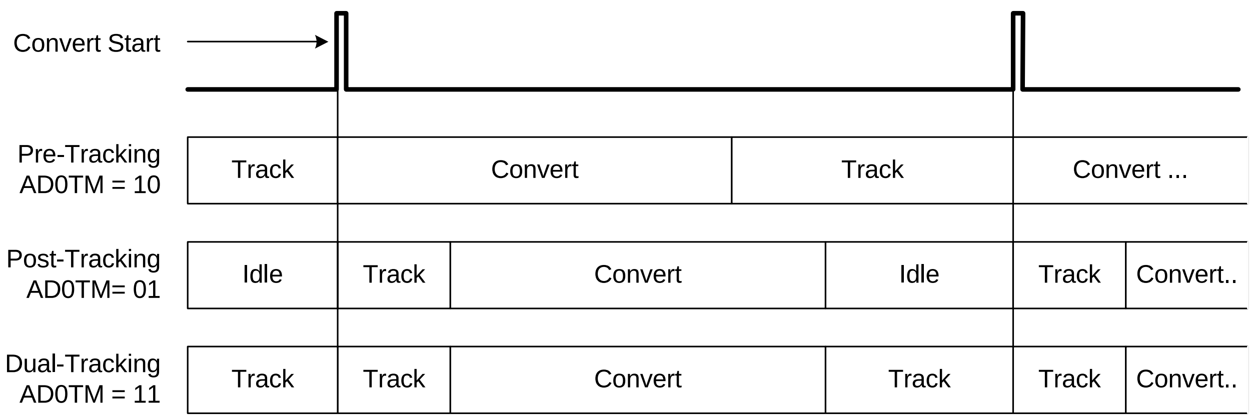


Figure 4.3. ADC0 Tracking Modes

4.3.3. Timing

ADC0 has a maximum conversion speed specified in Table 2.3 on page 28. ADC0 is clocked from the ADC0 Subsystem Clock (FCLK). The source of FCLK is selected based on the BURSTEN bit. When BURSTEN is logic 0, FCLK is derived from the current system clock. When BURSTEN is logic 1, FCLK is derived from the Burst Mode Oscillator, which is an independent clock source whose maximum frequency is specified in Table 2.3 on page 28.

When ADC0 is performing a conversion, it requires a clock source that is typically slower than FCLK. The ADC0 SAR conversion clock (SAR clock) is a divided version of FCLK. The divide ratio can be configured using the AD0SC bits in the ADC0CF register. The maximum SAR clock frequency is listed in Table 2.3 on page 28.

ADC0 can be in one of three states at any given time: tracking, converting, or idle. Tracking time depends on the tracking mode selected. For Pre-Tracking Mode, tracking is managed by software and ADC0 starts conversions immediately following the convert start signal. For Post-Tracking and Dual-Tracking Modes, the tracking time after the convert start signal is equal to the value determined by the AD0TK bits plus 2 FCLK cycles. Tracking is immediately followed by a conversion. The ADC0 conversion time is always 13 SAR clock cycles plus an additional 2 FCLK cycles to start and complete a conversion. Figure 4.4 shows timing diagrams for a conversion in Pre-Tracking Mode and tracking plus conversion in Post-Tracking or Dual-Tracking Mode. In this example, repeat count is set to one.

4.3.5. Output Conversion Code

The registers ADC0H and ADC0L contain the high and low bytes of the output conversion code. When the repeat count is set to 1, conversion codes are represented in 12-bit unsigned integer format and the output conversion code is updated after each conversion. Inputs are measured from 0 to V_{REF} x 4095/4096. Data can be right-justified or left-justified, depending on the setting of the AD0LJST bit (ADC0CN.2). Unused bits in the ADC0H and ADC0L registers are set to 0. Example codes are shown below for both right-justified and left-justified data.

Input Voltage	Right-Justified ADC0H:ADC0L (AD0LJST = 0)	Left-Justified ADC0H:ADC0L (AD0LJST = 1)
V_{REF} x 4095/4096	0x0FFF	0xFFF0
V_{REF} x 2048/4096	0x0800	0x8000
V_{REF} x 2047/4096	0x07FF	0x7FF0
0	0x0000	0x0000

When the ADC0 Repeat Count is greater than 1, the output conversion code represents the accumulated result of the conversions performed and is updated after the last conversion in the series is finished. Sets of 4, 8, or 16 consecutive samples can be accumulated and represented in unsigned integer format. The repeat count can be selected using the AD0RPT bits in the ADC0CF register. The value must be right-justified (AD0LJST = “0”), and unused bits in the ADC0H and ADC0L registers are set to '0'. The following example shows right-justified codes for repeat counts greater than 1. Notice that accumulating 2^n samples is equivalent to left-shifting by *n* bit positions when all samples returned from the ADC have the same value.

Input Voltage	Repeat Count = 4	Repeat Count = 8	Repeat Count = 16
V_{REF} x 4095/4096	0x3FFC	0x7FF8	0xFFF0
V_{REF} x 2048/4096	0x2000	0x4000	0x8000
V_{REF} x 2047/4096	0x1FFC	0x3FF8	0x7FF0
0	0x0000	0x0000	0x0000

SFR Definition 7.3. CPT0MD: Comparator0 Mode Selection

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
Reserved	—	CP0RIE	CP0FIE	—	—	CP0MD1	CP0MD0	00000010
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x9D

Bit7: **RESERVED**. Read = 0b. Must write 0b.
Bit6: **UNUSED**. Read = 0b. Write = don't care.
Bit5: **CP0RIE**: Comparator Rising-Edge Interrupt Enable.
0: Comparator rising-edge interrupt disabled.
1: Comparator rising-edge interrupt enabled.
Bit4: **CP0FIE**: Comparator Falling-Edge Interrupt Enable.
0: Comparator falling-edge interrupt disabled.
1: Comparator falling-edge interrupt enabled.
Note: It is necessary to enable both CP0xIE and the correspondent ECPx bit located in EIE1 SFR.
Bits3–2: **UNUSED**. Read = 00b. Write = don't care.
Bits1–0: **CP0MD1–CP0MD0**: Comparator0 Mode Select
These bits select the response time for Comparator0.

Mode	CP0MD1	CP0MD0	CP0 Falling Edge Response Time (TYP)
0	0	0	Fastest Response Time
1	0	1	—
2	1	0	—
3	1	1	Lowest Power Consumption

Note: Rising Edge response times are approximately double the Falling Edge response times.

Performance

The CIP-51 employs a pipelined architecture that greatly increases its instruction throughput over the standard 8051 architecture. In a standard 8051, all instructions except for MUL and DIV take 12 or 24 system clock cycles to execute, and usually have a maximum system clock of 12 MHz. By contrast, the CIP-51 core executes 70% of its instructions in one or two system clock cycles, with no instructions taking more than eight system clock cycles.

With the CIP-51's system clock running at 25 MHz, it has a peak throughput of 25 MIPS. The CIP-51 has a total of 109 instructions. The table below shows the total number of instructions that require each execution time.

Clocks to Execute	1	2	2/3	3	3/4	4	4/5	5	8
Number of Instructions	26	50	5	14	7	3	1	2	1

Programming and Debugging Support

In-system programming of the Flash program memory and communication with on-chip debug support logic is accomplished via the Silicon Labs 2-Wire (C2) interface. Note that the re-programmable Flash can also be read and written a single byte at a time by the application software using the MOVC and MOVX instructions. This feature allows program memory to be used for non-volatile data storage as well as updating program code under software control.

The on-chip debug support logic facilitates full speed in-circuit debugging, allowing the setting of hardware breakpoints, starting, stopping and single stepping through program execution (including interrupt service routines), examination of the program's call stack, and reading/writing the contents of registers and memory. This method of on-chip debugging is completely non-intrusive, requiring no RAM, Stack, timers, or other on-chip resources.

The CIP-51 is supported by development tools from Silicon Laboratories, Inc. and third party vendors. Silicon Laboratories provides an integrated development environment (IDE) including editor, evaluation compiler, assembler, debugger and programmer. The IDE's debugger and programmer interface to the CIP-51 via the on-chip debug logic to provide fast and efficient in-system device programming and debugging. Third party macro assemblers and C compilers are also available.

8.1. Instruction Set

The instruction set of the CIP-51 System Controller is fully compatible with the standard MCS-51™ instruction set. Standard 8051 development tools can be used to develop software for the CIP-51. All CIP-51 instructions are the binary and functional equivalent of their MCS-51™ counterparts, including opcodes, addressing modes and effect on PSW flags. However, instruction timing is different than that of the standard 8051.

8.1.1. Instruction and CPU Timing

In many 8051 implementations, a distinction is made between machine cycles and clock cycles, with machine cycles varying from 2 to 12 clock cycles in length. However, the CIP-51 implementation is based solely on clock cycle timing. All instruction timings are specified in terms of clock cycles.

Due to the pipelined architecture of the CIP-51, most instructions execute in the same number of clock cycles as there are program bytes in the instruction. Conditional branch instructions take one less clock cycle to complete when the branch is not taken as opposed to when the branch is taken. Table 8.1 is the CIP-51 Instruction Set Summary, which includes the mnemonic, number of bytes, and number of clock cycles for each instruction.

SFR Definition 8.7. PCON: Power Control

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

SFR Address: 0x87

Bits7–2: **RESERVED**.
Bit1: **STOP**: STOP Mode Select.
Writing a 1 to this bit will place the CIP-51 into STOP mode. This bit will always read 0.
1: CIP-51 forced into power-down mode. (Turns off internal oscillator).
Bit0: **IDLE**: IDLE Mode Select.
Writing a 1 to this bit will place the CIP-51 into IDLE mode. This bit will always read 0.
1: CIP-51 forced into IDLE mode. (Shuts off clock to CPU, but clock to Timers, Interrupts, and all peripherals remain active.)

SFR Definition 13.11. P1MDOUT: Port1 Output Mode

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: 0xA5

Bits7–0: Output Configuration Bits for P1.7–P1.0 (respectively): ignored if corresponding bit in register P1MDIN is logic 0.
0: Corresponding P1.n Output is open-drain.
1: Corresponding P1.n Output is push-pull.

Note: When SDA and SCL appear on any of the Port I/O, each are open-drain regardless of the value of P0MDOUT.

SFR Definition 13.12. P1SKIP: Port1 Skip

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: 0xD5

Bits7–0: **P1SKIP[7:0]**: Port1 Crossbar Skip Enable Bits.
These bits select Port pins to be skipped by the Crossbar Decoder. Port pins used as analog inputs (for ADC or Comparator) or used as special functions (V_{REF} input, external oscillator circuit, CNVSTR input) should be skipped by the Crossbar.
0: Corresponding P1.n pin is not skipped by the Crossbar.
1: Corresponding P1.n pin is skipped by the Crossbar.

SFR Definition 15.2. SBUF0: Serial (UART0) Port Data Buffer

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: 0x99

Bits7–0: **SBUF0[7:0]**: Serial Data Buffer Bits 7–0 (MSB–LSB)
This SFR accesses two registers; a transmit shift register and a receive latch register. When data is written to SBUF0, it goes to the transmit shift register and is held for serial transmission. Writing a byte to SBUF0 initiates the transmission. A read of SBUF0 returns the contents of the receive latch.

Table 15.1. Timer Settings for Standard Baud Rates Using the Internal Oscillator

	Frequency: 24.5 MHz						
	Target Baud Rate (bps)	**Baud Rate % Error**	**Oscillator Divide Factor**	**Timer Clock Source**	**SCA1–SCA0 (pre-scale select)***	**T1M***	**Timer 1 Reload Value (hex)**
SYSCLK from Internal Osc.	230400	–0.32%	106	SYSCLK	XX	1	0xCB
	115200	–0.32%	212	SYSCLK	XX	1	0x96
	57600	0.15%	426	SYSCLK	XX	1	0x2B
	28800	–0.32%	848	SYSCLK / 4	01	0	0x96
	14400	0.15%	1704	SYSCLK / 12	00	0	0xB9
	9600	–0.32%	2544	SYSCLK / 12	00	0	0x96
	2400	–0.32%	10176	SYSCLK / 48	10	0	0x96
	1200	0.15%	20448	SYSCLK / 48	10	0	0x2B

X = Don't care

Note: SCA1–SCA0 and T1M bit definitions can be found in Section 18.1.

16. Enhanced Serial Peripheral Interface (SPI0)

The Serial Peripheral Interface (SPI0) provides access to a flexible, full-duplex synchronous serial bus. SPI0 can operate as a master or slave device in both 3-wire or 4-wire modes, and supports multiple masters and slaves on a single SPI bus. The slave-select (NSS) signal can be configured as an input to select SPI0 in slave mode, or to disable Master Mode operation in a multi-master environment, avoiding contention on the SPI bus when more than one master attempts simultaneous data transfers. NSS can also be configured as a chip-select output in master mode, or disabled for 3-wire operation. Additional general purpose port I/O pins can be used to select multiple slave devices in master mode.

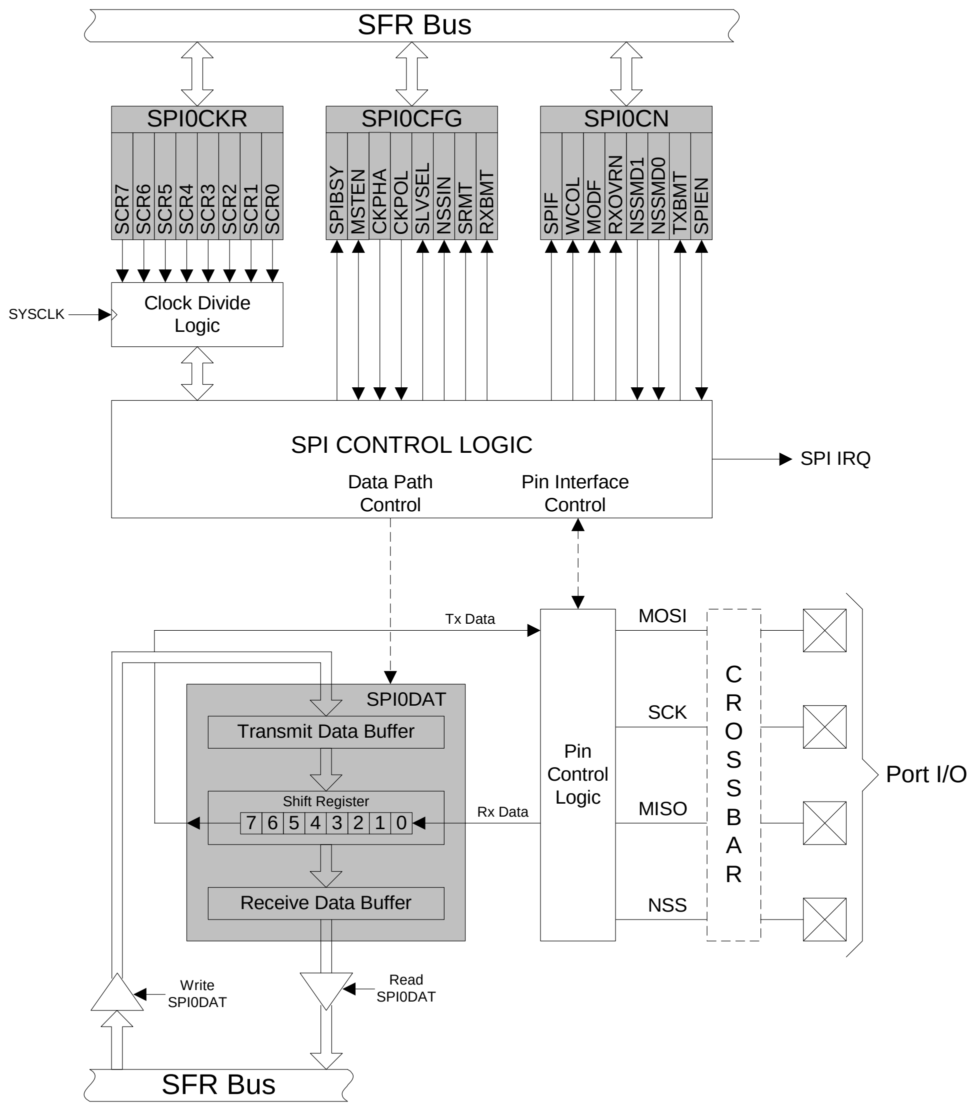


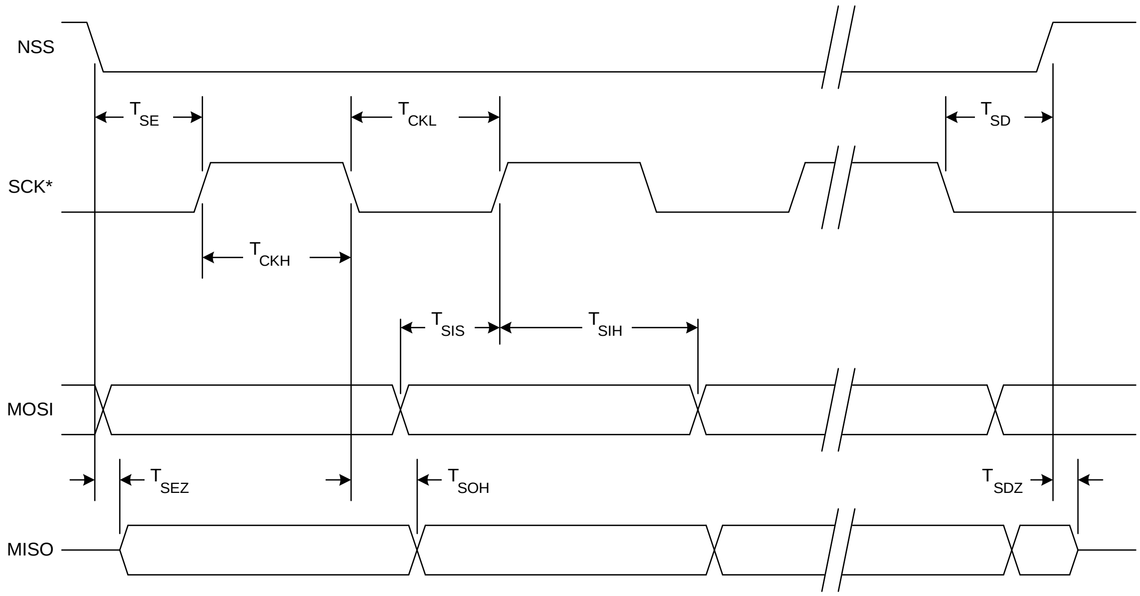
Figure 16.1. SPI Block Diagram

SFR Definition 16.4. SPI0DAT: SPI0 Data

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	

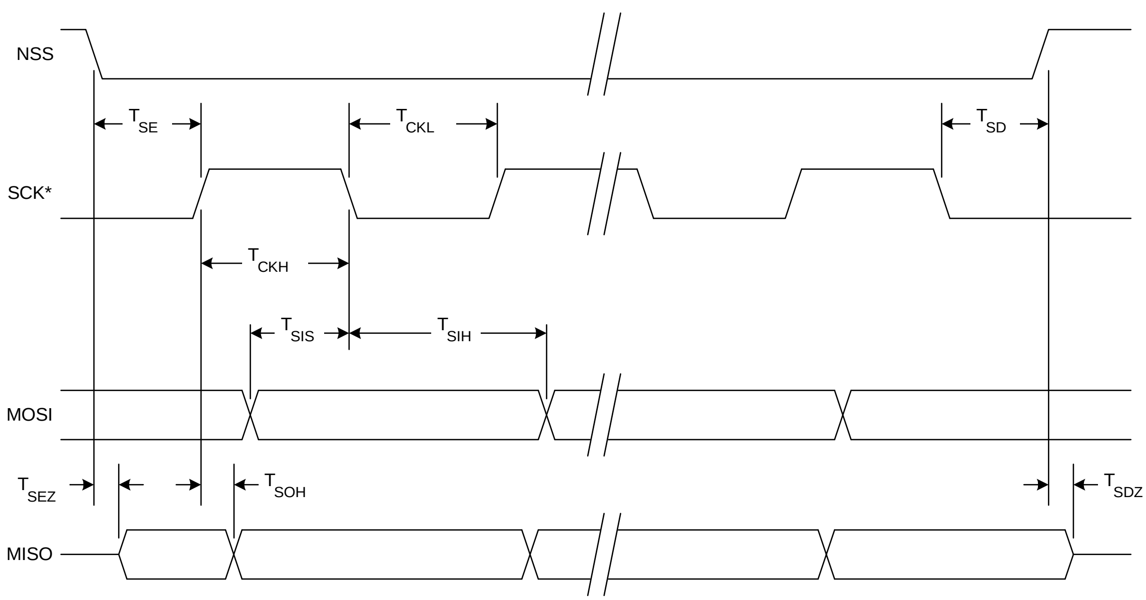
SFR Address: 0xA3

Bits7–0: **SPI0DAT**: SPI0 Transmit and Receive Data.
The SPI0DAT register is used to transmit and receive SPI0 data. Writing data to SPI0DAT places the data into the transmit buffer and initiates a transfer when in Master Mode. A read of SPI0DAT returns the contents of the receive buffer.



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

Figure 16.8. SPI Slave Timing (CKPHA = 0)



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

Figure 16.9. SPI Slave Timing (CKPHA = 1)

SFR Definition 17.3. LINCF Control Mode Register

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

SFR Address: 0x95

Bit7: **LINEN**: LIN Interface Enable bit
0: LIN0 is disabled.
1: LIN0 is enabled.

Bit6: **MODE**: LIN Mode Selection
0: LIN0 operates in Slave mode.
1: LIN0 operates in Master mode.

Bit5: **ABAUD**: LIN Mode Automatic Baud Rate Selection (**slave mode only**).
0: Manual baud rate selection is enabled.
1: Automatic baud rate selection is enabled.

SFR Definition 17.8. LIN0DT5: LIN0 Data Byte 5

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	

Address: 0x04 (indirect)

Bit7–0: **LIN0DT5**: LIN Data Byte 5.
Serial Data Byte 5 that is received or transmitted across the LIN interface.

SFR Definition 17.9. LIN0DT6: LIN0 Data Byte 6

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	

Address: 0x05 (indirect)

Bit7–0: **LIN0DT6**: LIN Data Byte 6.
Serial Data Byte 6 that is received or transmitted across the LIN interface.

SFR Definition 17.10. LIN0DT7: LIN0 Data Byte 7

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	

Address: 0x06 (indirect)

Bit7–0: **LIN0DT7**: LIN Data Byte 7.
Serial Data Byte 7 that is received or transmitted across the LIN interface.

SFR Definition 17.11. LIN0DT8: LIN0 Data Byte 8

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	

Address: 0x07 (indirect)

Bit7–0: **LIN0DT8**: LIN Data Byte 8.
Serial Data Byte 8 that is received or transmitted across the LIN interface.

SFR Definition 18.8. TMR2CN: Timer 2 Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
TF2H	TF2L	TF2LEN	TF2CEN	T2SPLIT	TR2	—	T2XCLK	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Bit Addressable

SFR Address: 0xC8

Bit7: **TF2H**: Timer 2 High Byte Overflow Flag.
Set by hardware when the Timer 2 high byte overflows from 0xFF to 0x00. In 16 bit mode, this will occur when Timer 2 overflows from 0xFFFF to 0x0000. When the Timer 2 interrupt is enabled, setting this bit causes the CPU to vector to the Timer 2 interrupt service routine. TF2H is not automatically cleared by hardware and must be cleared by software.

Bit6: **TF2L**: Timer 2 Low Byte Overflow Flag.
Set by hardware when the Timer 2 low byte overflows from 0xFF to 0x00. When this bit is set, an interrupt will be generated if TF2LEN is set and Timer 2 interrupts are enabled. TF2L will set when the low byte overflows regardless of the Timer 2 mode. This bit is not automatically cleared by hardware.

Bit5: **TF2LEN**: Timer 2 Low Byte Interrupt Enable.
This bit enables/disables Timer 2 Low Byte interrupts. If TF2LEN is set and Timer 2 interrupts are enabled, an interrupt will be generated when the low byte of Timer 2 overflows. This bit should be cleared when operating Timer 2 in 16-bit mode.
0: Timer 2 Low Byte interrupts disabled.
1: Timer 2 Low Byte interrupts enabled.

Bit4: **TF2CEN**. Timer 2 Capture Enable.
0: Timer 2 capture mode disabled.
1: Timer 2 capture mode enabled.

Bit3: **T2SPLIT**: Timer 2 Split Mode Enable.
When this bit is set, Timer 2 operates as two 8-bit timers with auto-reload.
0: Timer 2 operates in 16-bit auto-reload mode.
1: Timer 2 operates as two 8-bit auto-reload timers.

Bit2: **TR2**: Timer 2 Run Control.
This bit enables/disables Timer 2. In 8-bit mode, this bit enables/disables TMR2H only; TMR2L is always enabled in this mode.
0: Timer 2 disabled.
1: Timer 2 enabled.

Bit1: **Unused**. Read = 0b. Write = don't care.

Bit0: **T2XCLK**: Timer 2 External Clock Select.
This bit selects the external clock source for Timer 2. If Timer 2 is in 8-bit mode, this bit selects the external oscillator clock source for both timer bytes. However, the Timer 2 Clock Select bits (T2MH and T2ML in register CKCON) may still be used to select between the external clock and the system clock for either timer.
0: Timer 2 external clock selection is the system clock divided by 12.
1: Timer 2 external clock selection is the external clock divided by 8.

19.1. PCA Counter/Timer

The 16-bit PCA counter/timer consists of two 8-bit SFRs: PCA0L and PCA0H. PCA0H is the high byte (MSB) of the 16-bit counter/timer and PCA0L is the low byte (LSB). Reading PCA0L automatically latches the value of PCA0H into a “snapshot” register; the following PCA0H read accesses this “snapshot” register. **Reading the PCA0L Register first guarantees an accurate reading of the entire 16-bit PCA0 counter.** Reading PCA0H or PCA0L does not disturb the counter operation. The CPS2-CPS0 bits in the PCA0MD register select the timebase for the counter/timer as shown in Table 19.1.

When the counter/timer overflows from 0xFFFF to 0x0000, the Counter Overflow Flag (CF) in PCA0MD is set to logic 1 and an interrupt request is generated if CF interrupts are enabled. Setting the ECF bit in PCA0MD to logic 1 enables the CF flag to generate an interrupt request. The CF bit is not automatically cleared by hardware when the CPU vectors to the interrupt service routine, and must be cleared by software (Note: PCA0 interrupts must be globally enabled before CF interrupts are recognized. PCA0 interrupts are globally enabled by setting the EA bit (IE.7) and the EPCA0 bit in EIE1 to logic 1). Clearing the CIDL bit in the PCA0MD register allows the PCA to continue normal operation while the CPU is in Idle mode.

Table 19.1. PCA Timebase Input Options

CPS2	CPS1	CPS0	Timebase
0	0	0	System clock divided by 12
0	0	1	System clock divided by 4
0	1	0	Timer 0 overflow
0	1	1	High-to-low transitions on ECI (max rate = system clock divided by 4)
1	0	0	System clock
1	0	1	External oscillator source divided by 8*

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

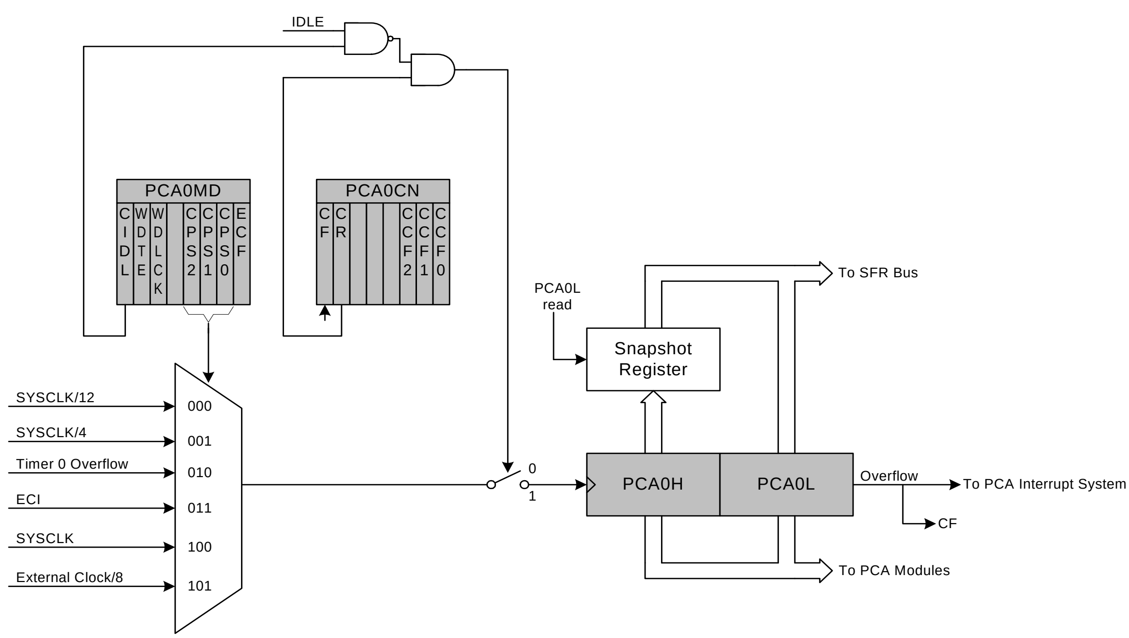


Figure 19.2. PCA Counter/Timer Block Diagram

19.2. Capture/Compare Modules

Each module can be configured to operate independently in one of six operation modes: Edge-triggered Capture, Software Timer, High Speed Output, Frequency Output, 8-Bit Pulse Width Modulator, or 16-Bit Pulse Width Modulator. Each module has Special Function Registers (SFRs) associated with it in the CIP-51 system controller. These registers are used to exchange data with a module and configure the module's mode of operation.

Table 19.2 summarizes the bit settings in the PCA0CPMn registers used to select the PCA capture/compare module's operating modes. Setting the ECCFn bit in a PCA0CPMn register enables the module's CCFn interrupt. Note that PCA0 interrupts must be globally enabled before individual CCFn interrupts are recognized. PCA0 interrupts are globally enabled by setting the EA bit and the EPCA0 bit to logic 1. See Figure 19.3 for details on the PCA interrupt configuration.

Table 19.2. PCA0CPM Register Settings for PCA Capture/Compare Modules

PWM16	ECOM	CAPP	CAPN	MAT	TOG	PWM	ECCF	Operation Mode
X	X	1	0	0	0	0	X	Capture triggered by positive edge on CEXn
X	X	0	1	0	0	0	X	Capture triggered by negative edge on CEXn
X	X	1	1	0	0	0	X	Capture triggered by transition on CEXn
X	1	0	0	1	0	0	X	Software Timer
X	1	0	0	1	1	0	X	High Speed Output
X	1	0	0	X	1	1	X	Frequency Output
0	1	0	0	X	0	1	X	8-Bit Pulse Width Modulator
1	1	0	0	X	0	1	X	16-Bit Pulse Width Modulator
X = Don't Care								

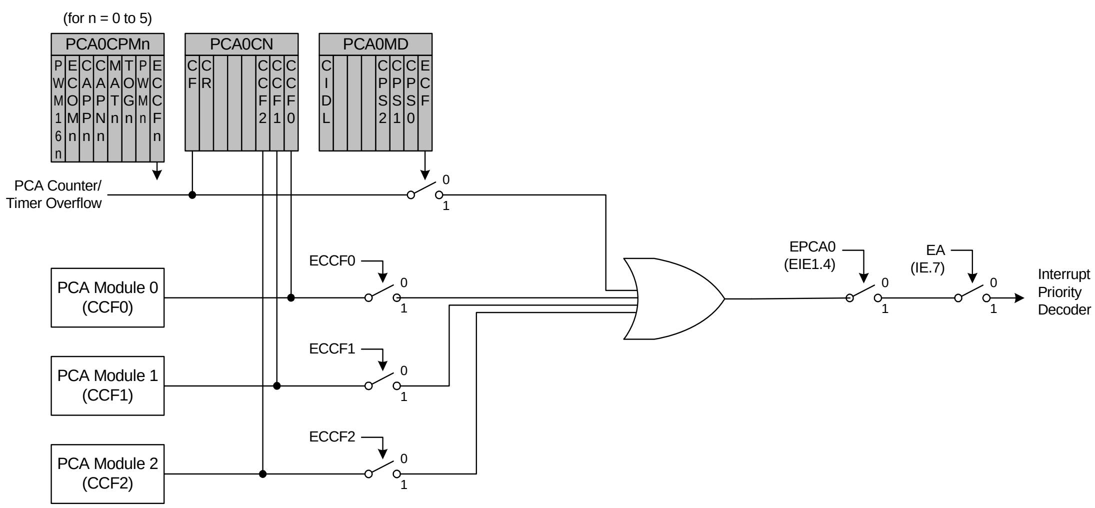


Figure 19.3. PCA Interrupt Block Diagram

19.2.4. Frequency Output Mode

Frequency Output Mode produces a programmable-frequency square wave on the module's associated CEXn pin. The capture/compare module high byte holds the number of PCA clocks to count before the output is toggled. The frequency of the square wave is then defined by Equation 19.1.

$$F_{CEXn} = \frac{F_{PCA}}{2 \times PCA0CPHn}$$

Note: A value of 0x00 in the PCA0CPHn register is equal to 256 for this equation.

Equation 19.1. Square Wave Frequency Output

Where F_{PCA} is the frequency of the clock selected by the CPS2-0 bits in the PCA mode register, PCA0MD. The lower byte of the capture/compare module is compared to the PCA counter low byte; on a match, CEXn is toggled and the offset held in the high byte is added to the matched value in PCA0CPLn. Frequency Output Mode is enabled by setting the ECOMn, TOGn, and PWMn bits in the PCA0CPMn register.

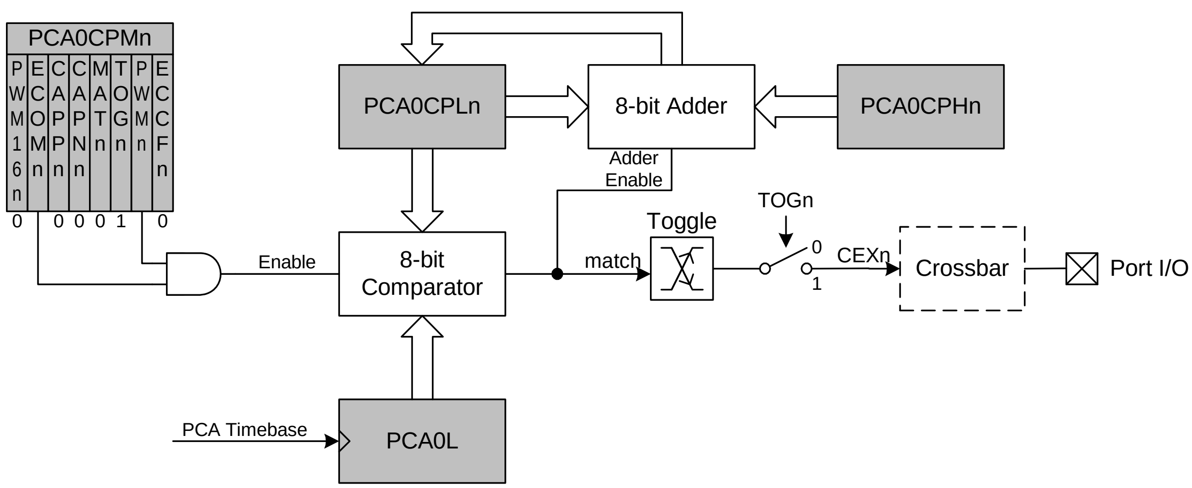


Figure 19.7. PCA Frequency Output Mode

SFR Definition 19.4. 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

Bits7–0: **PCA0L**: PCA Counter/Timer Low Byte.
The PCA0L register holds the low byte (LSB) of the 16-bit PCA Counter/Timer.

SFR Definition 19.5. 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

Bits7–0: **PCA0H**: PCA Counter/Timer High Byte.
The PCA0H register holds the high byte (MSB) of the 16-bit PCA Counter/Timer.

SFR Definition 19.6. 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

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 19.7. 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: 0xE9, PCA0CPH2: 0xEC

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