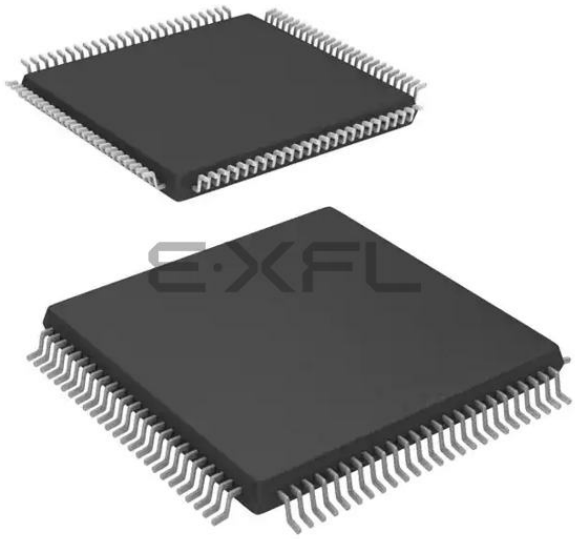

Welcome to E-XFL.COM

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	Not For New Designs
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	CANbus, EBI/EMI, SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	64
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4.25K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x8b, 13x12b; D/A 2x10b, 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TQFP
Supplier Device Package	100-TQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f040-gqr

List of Registers

1.10. Comparators and DACs

Each C8051F040/1/2/3 MCU has two 12-bit DACs, and all C8051F04x devices have three comparators on chip. The MCU data and control interface to each comparator and DAC is via the Special Function Registers. The MCU can place any DAC or comparator in low power shutdown mode.

The comparators have software programmable hysteresis and response time. Each comparator can generate an interrupt on its rising edge, falling edge, or both; these interrupts are capable of waking up the MCU from sleep mode. The comparators' output state can also be polled in software. The comparator outputs can be programmed to appear on the Port I/O pins via the Crossbar.

The DACs are voltage output mode and include a flexible output scheduling mechanism. This scheduling mechanism allows DAC output updates to be forced by a software write or a Timer 2, 3, or 4 overflow. The DAC voltage reference is supplied via the dedicated VREFD input pin on C8051F040/2 devices or via the internal voltage reference on C8051F041/3 devices. The DACs are especially useful as references for the comparators or offsets for the differential inputs of the ADC.

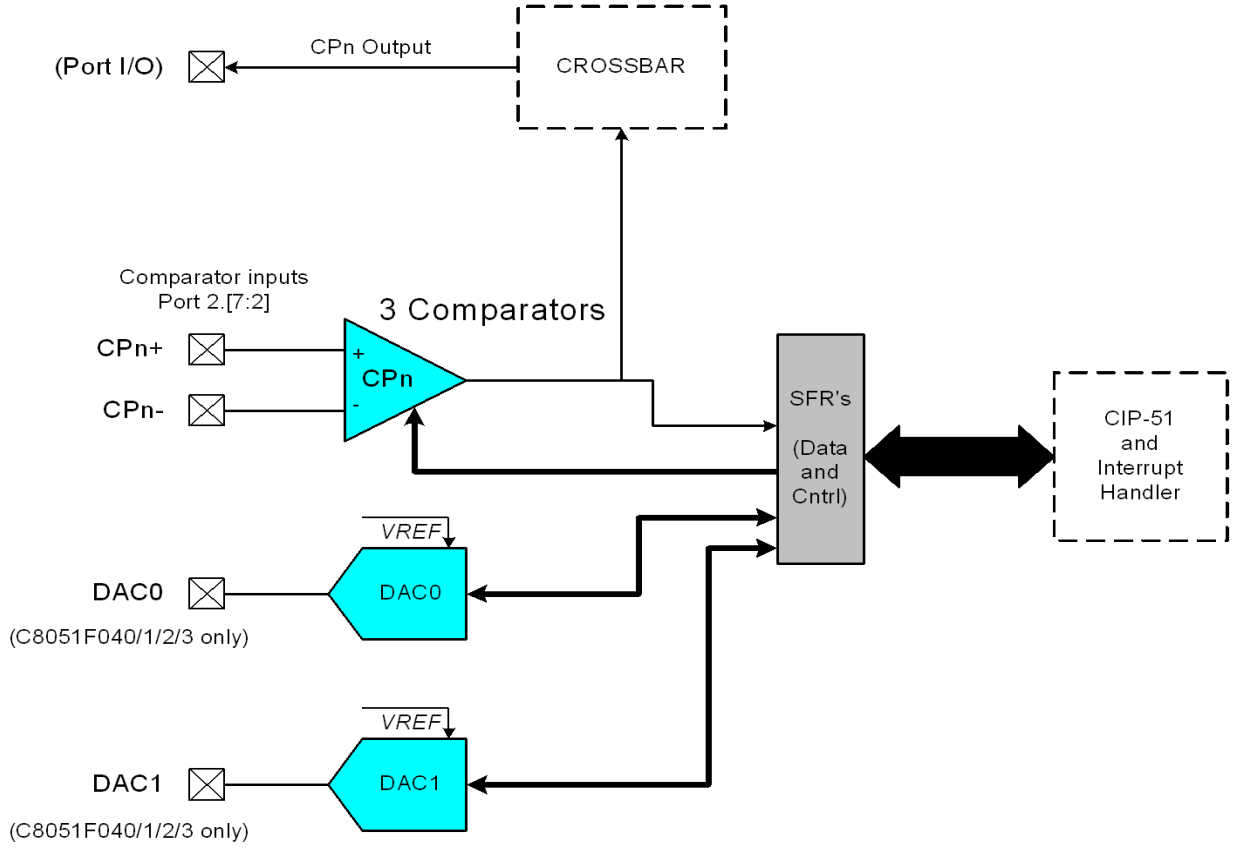

Figure 1.14. Comparator and DAC Diagram

5.2. High-Voltage Difference Amplifier

The High Voltage Difference Amplifier (HVDA) can be used to measure high differential voltages up to 60 V peak-to-peak, reject high common-mode voltages up to ±60 V, and condition the signal voltage range to be suitable for input to ADC0. The input signal to the HVDA may be below AGND to –60 volts, and as high as +60 volts, making the device suitable for both single and dual supply applications. The HVDA provides a common-mode signal for the ADC via the High Voltage Reference Input (HVREF), allowing measurement of signals outside the specified ADC input range using on-chip circuitry. The HVDA has a gain of 0.05 V/V to 14 V/V. The first stage 20:1 difference amplifier has a gain of 0.05 V/V when the output amplifier is used as a unity gain buffer. When the output amplifier is set to a gain of 280 (selected using the HVGAIN bits in the High Voltage Control Register), an overall gain of 14 can be attained.

The HVDA uses four available external pins: +HVAIN, –HVAIN, HVCAP, and HVREF. HVAIN+ and HVAIN- serve as the differential inputs to the HVDA. HVREF should be used to provide a common mode reference for input to ADC0, and to prevent the output of the HVDA circuit from saturating. The output from the HVDA circuit as calculated by Equation 5.1 must remain within the “Output Voltage Range” specification listed in Table 5.3. The ideal value for HVREF in most applications is equal to 1/2 the supply voltage for the device. When the ADC is configured for differential measurement, the HVREF signal is applied to the AIN- input of the ADC, thereby removing HVREF from the measurement. HVCAP facilitates the use of a capacitor for noise filtering in conjunction with R7 (see Figure 5.3 for R7 and other approximate resistor values). Alternatively, the HVCAP could also be used to access amplification of the first stage of the HVDA at an external pin. (See Table 5.3 on page 68 for electrical specifications of the HVDA.)

$$V_{OUT} = [(HVAIN+) - (HVAIN-)] \cdot Gain + HVREF$$

Note: *The output voltage of the HVDA is selected as an input to the AIN+ input of ADC0 via its analog multiplexer (AMUX0). HVDA output voltages outside the ADC’s input range will result in saturation of the ADC input. Allow for adequate settle/tracking time for proper voltage measurements.*

Equation 5.1. Calculating HVDA Output Voltage to AIN+

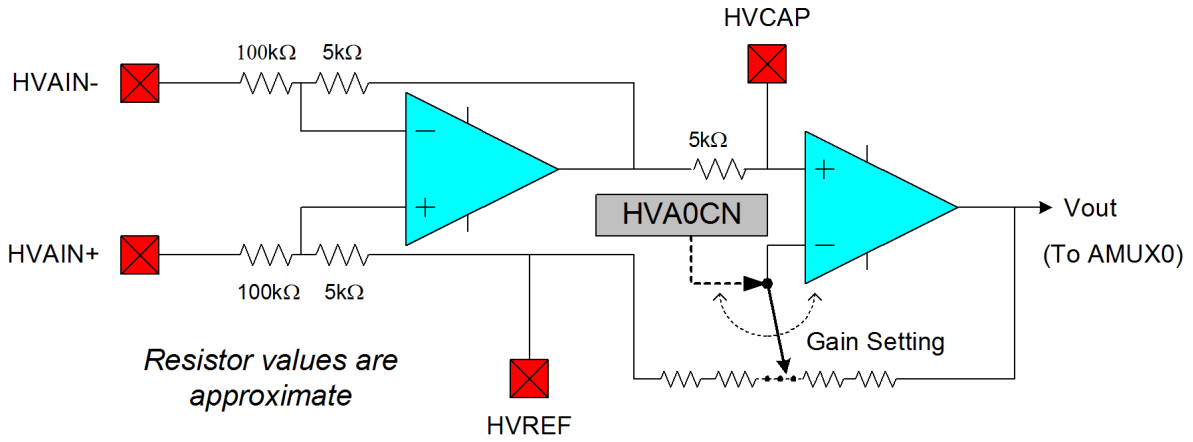

Figure 5.3. High Voltage Difference Amplifier Functional Diagram

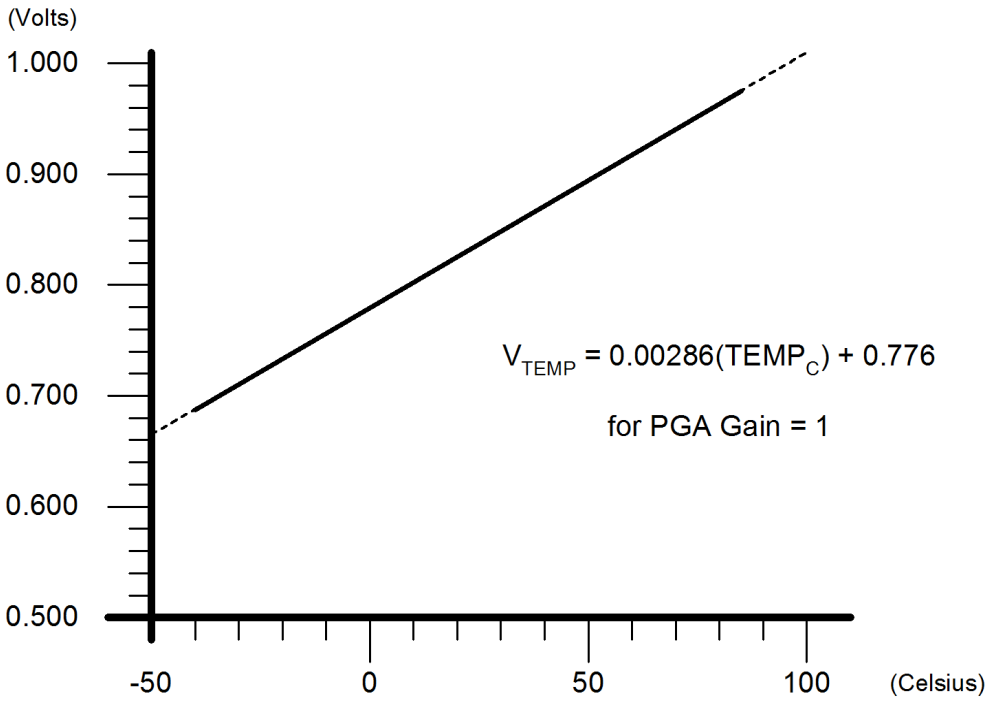

Figure 5.6. Temperature Sensor Transfer Function

SFR Definition 5.6. ADC0CN: ADC0 Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
AD0EN	AD0TM	AD0INT	AD0BUSY	AD0CM1	AD0CM0	AD0WINT	AD0LJST	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Bit Addressable

SFR Address: 0xE8
SFR Page: 0

Bit7: AD0EN: ADC0 Enable Bit.
0: ADC0 Disabled. ADC0 is in low-power shutdown.
1: ADC0 Enabled. ADC0 is active and ready for data conversions.

Bit6: AD0TM: ADC Track Mode Bit
0: When the ADC is enabled, tracking is continuous unless a conversion is in process
1: Tracking Defined by AD0CM1-0 bits

Bit5: AD0INT: ADC0 Conversion Complete Interrupt Flag.
This flag must be cleared by software.
0: ADC0 has not completed a data conversion since the last time this flag was cleared.
1: ADC0 has completed a data conversion.

Bit4: AD0BUSY: ADC0 Busy Bit.
Read:
0: ADC0 Conversion is complete or a conversion is not currently in progress. AD0INT is set to logic 1 on the falling edge of AD0BUSY.
1: ADC0 Conversion is in progress.
Write:
0: No Effect.
1: Initiates ADC0 Conversion if AD0CM1-0 = 00b

Bit3-2: AD0CM1-0: ADC0 Start of Conversion Mode Select.
If AD0TM = 0:
00: ADC0 conversion initiated on every write of ‘1’ to AD0BUSY.
01: ADC0 conversion initiated on overflow of Timer 3.
10: ADC0 conversion initiated on rising edge of external CNVSTR0.
11: ADC0 conversion initiated on overflow of Timer 2.
If AD0TM = 1:
00: Tracking starts with the write of ‘1’ to AD0BUSY and lasts for 3 SAR clocks, followed by conversion.
01: Tracking started by the overflow of Timer 3 and last for 3 SAR clocks, followed by conversion.
10: ADC0 tracks only when CNVSTR0 input is logic low; conversion starts on rising CNVSTR0 edge.
11: Tracking started by the overflow of Timer 2 and last for 3 SAR clocks, followed by conversion.

Bit1: AD0WINT: ADC0 Window Compare Interrupt Flag.
This bit must be cleared by software.
0: ADC0 Window Comparison Data match has not occurred since this flag was last cleared.
1: ADC0 Window Comparison Data match has occurred.

Bit0: AD0LJST: ADC0 Left Justify Select.
0: Data in ADC0H:ADC0L registers are right-justified.
1: Data in ADC0H:ADC0L registers are left-justified.

Table 5.2. 12-Bit ADC0 Electrical Characteristics

V_{DD} = 3.0 V, AV+ = 3.0 V, VREF = 2.40 V (REFBE = 0), PGA Gain = 1, –40 to +85 °C unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
DC Accuracy					
Resolution			12		bits
Integral Nonlinearity		—	—	±1	LSB
Differential Nonlinearity	Guaranteed Monotonic	—	—	±1	LSB
Offset Error	Note 1	—	0.5±3	—	LSB
Full Scale Error	Differential mode; See Note 1	—	0.4±3	—	LSB
Offset Temperature Coefficient		—	±0.25	—	ppm/°C
Dynamic Performance (10 kHz sine-wave input, 0 to 1 dB below Full Scale, 100 ksps)					
Signal-to-Noise Plus Distortion		66	—	—	dB
Total Harmonic Distortion	Up to the 5th harmonic	—	–75	—	dB
Spurious-Free Dynamic Range		—	80	—	dB
Conversion Rate					
Maximum SAR Clock Frequency		—	—	2.5	MHz
Conversion Time in SAR Clocks		16	—	—	clocks
Track/Hold Acquisition Time		1.5	—	—	µs
Throughput Rate		—	—	100	ksps
Analog Inputs					
Input Voltage Range	Single-ended operation	0	—	VREF	V
Common-mode Voltage Range	Differential operation	AGND	—	AV+	V
Input Capacitance		—	10	—	pF
Temperature Sensor					
Nonlinearity	Notes 1, 2	—	±1	—	°C
Absolute Accuracy	Notes 1, 2	—	±3	—	°C
Gain	Notes 1, 2	—	2.86 ±0.034	—	mV/°C
Offset	Notes 1, 2 (Temp = 0 °C)	—	0.776 ±0.009	—	V
Power Specifications					
Power Supply Current (AV+ supplied to ADC)	Operating Mode, 100 ksps	—	450	900	µA
Power Supply Rejection		—	±0.3	—	mV/V

Notes:
1. Represents one standard deviation from the mean.
2. Includes ADC offset, gain, and linearity variations.

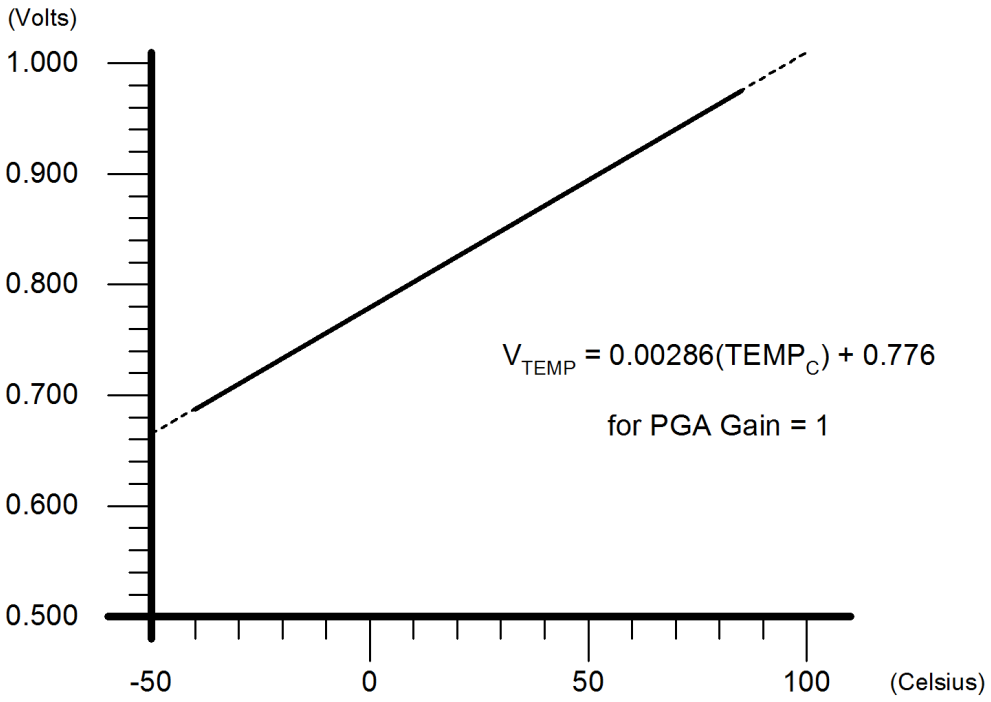

Figure 6.6. Temperature Sensor Transfer Function

SFR Definition 6.6. ADC0CN: ADC0 Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
AD0EN	AD0TM	AD0INT	AD0BUSY	AD0CM1	AD0CM0	AD0WINT	AD0LJST	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Bit Addressable

SFR Address: 0xE8
SFR Page: 0

Bit7: AD0EN: ADC0 Enable Bit.
0: ADC0 Disabled. ADC0 is in low-power shutdown.
1: ADC0 Enabled. ADC0 is active and ready for data conversions.

Bit6: AD0TM: ADC Track Mode Bit
0: When the ADC is enabled, tracking is continuous unless a conversion is in process
1: Tracking Defined by AD0CM1-0 bits

Bit5: AD0INT: ADC0 Conversion Complete Interrupt Flag.
This flag must be cleared by software.
0: ADC0 has not completed a data conversion since the last time this flag was cleared.
1: ADC0 has completed a data conversion.

Bit4: AD0BUSY: ADC0 Busy Bit.
Read:
0: ADC0 Conversion is complete or a conversion is not currently in progress. AD0INT is set to logic 1 on the falling edge of AD0BUSY.
1: ADC0 Conversion is in progress.
Write:
0: No Effect.
1: Initiates ADC0 Conversion if AD0CM1-0 = 00b

Bit3-2: AD0CM1-0: ADC0 Start of Conversion Mode Select.
If AD0TM = 0:
00: ADC0 conversion initiated on every write of ‘1’ to AD0BUSY.
01: ADC0 conversion initiated on overflow of Timer 3.
10: ADC0 conversion initiated on rising edge of external CNVSTR0.
11: ADC0 conversion initiated on overflow of Timer 2.
If AD0TM = 1:
00: Tracking starts with the write of ‘1’ to AD0BUSY and lasts for 3 SAR clocks, followed by conversion.
01: Tracking started by the overflow of Timer 3 and last for 3 SAR clocks, followed by conversion.
10: ADC0 tracks only when CNVSTR0 input is logic low; conversion starts on rising CNVSTR0 edge.
11: Tracking started by the overflow of Timer 2 and last for 3 SAR clocks, followed by conversion.

Bit1: AD0WINT: ADC0 Window Compare Interrupt Flag.
This bit must be cleared by software.
0: ADC0 Window Comparison Data match has not occurred since this flag was last cleared.
1: ADC0 Window Comparison Data match has occurred.

Bit0: AD0LJST: ADC0 Left Justify Select.
0: Data in ADC0H:ADC0L registers are right-justified.
1: Data in ADC0H:ADC0L registers are left-justified.

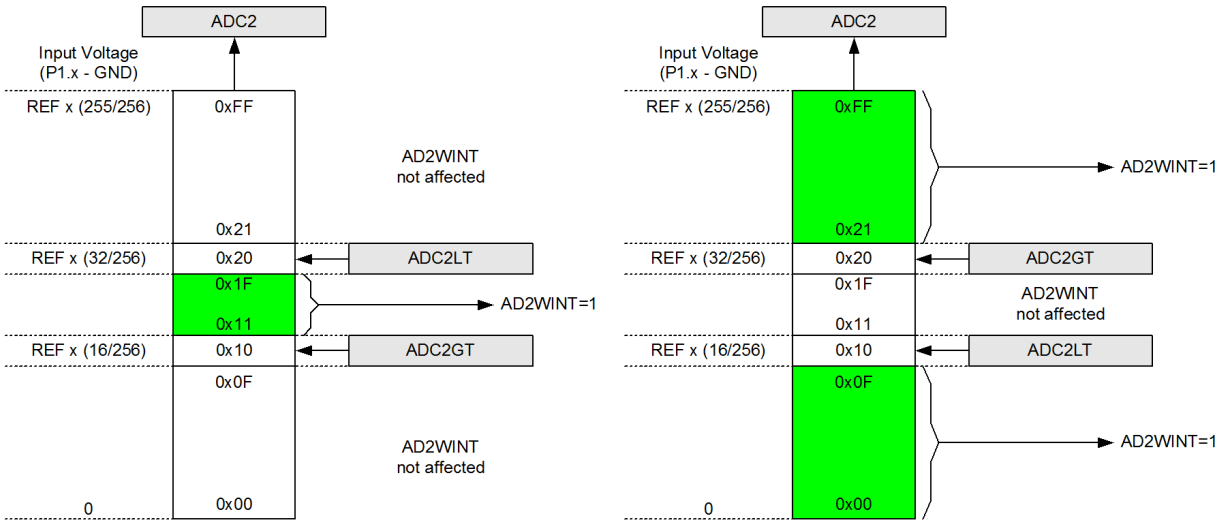

Figure 7.5. ADC Window Compare Examples, Single-Ended Mode

12.3. Interrupt Handler

The CIP-51 includes an extended interrupt system supporting a total of 20 interrupt sources with two priority levels. The allocation of interrupt sources between on-chip peripherals and external inputs pins varies according to the specific version of the device. Each interrupt source has one or more associated interrupt-pending flag(s) located in an SFR. When a peripheral or external source meets a valid interrupt condition, the associated interrupt-pending flag is set to logic 1.

If interrupts are enabled for the source, an interrupt request is generated when the interrupt-pending flag is set. As soon as execution of the current instruction is complete, the CPU generates an LCALL to a predetermined address to begin execution of an interrupt service routine (ISR). Each ISR must end with an RETI instruction, which returns program execution to the next instruction that would have been executed if the interrupt request had not occurred. If interrupts are not enabled, the interrupt-pending flag is ignored by the hardware and program execution continues as normal. The interrupt-pending flag is set to logic 1 regardless of the interrupt's enable/disable state.

Each interrupt source can be individually enabled or disabled through the use of an associated interrupt enable bit in an SFR (IE-EIE2). However, interrupts must first be globally enabled by setting the EA bit (IE.7) to logic 1 before the individual interrupt enables are recognized. Setting the EA bit to logic 0 disables all interrupt sources regardless of the individual interrupt-enable settings.

Note: Any instruction that clears the EA bit should be immediately followed by an instruction that has two or more opcode bytes. For example:

```
// in 'C':
EA = 0;     // clear EA bit
EA = 0;     // ... followed by another 2-byte opcode

; in assembly:
CLR  EA     ; clear EA bit
CLR  EA     ; ... followed by another 2-byte opcode
```

If an interrupt is posted during the execution phase of a "CLR EA" opcode (or any instruction which clears the EA bit), and the instruction is followed by a single-cycle instruction, the interrupt may be taken. However, a read of the EA bit will return a '0' inside the interrupt service routine. When the "CLR EA" opcode is followed by a multi-cycle instruction, the interrupt will not be taken.

Some interrupt-pending flags are automatically cleared by the hardware when the CPU vectors to the ISR. However, most are not cleared by the hardware and must be cleared by software before returning from the ISR. If an interrupt-pending flag remains set after the CPU completes the return-from-interrupt (RETI) instruction, a new interrupt request will be generated immediately and the CPU will re-enter the ISR after the completion of the next instruction.

12.3.1. MCU Interrupt Sources and Vectors

The MCUs support 20 interrupt sources. Software can simulate an interrupt event by setting any interrupt-pending flag to logic 1. If interrupts are enabled for the flag, an interrupt request will be generated and the CPU will vector to the ISR address associated with the interrupt-pending flag. MCU interrupt sources, associated vector addresses, priority order and control bits are summarized in Table 12.4. Refer to the datasheet section associated with a particular on-chip peripheral for information regarding valid interrupt conditions for the peripheral and the behavior of its interrupt-pending flag(s).

12.3.2. External Interrupts

The external interrupt sources (/INT0 and /INT1) are configurable as active-low level-sensitive or active-low edge-sensitive inputs depending on the setting of bits IT0 (TCON.0) and IT1 (TCON.2). IE0 (TCON.1) and IE1 (TCON.3) serve as the interrupt-pending flag for the /INT0 and /INT1 external interrupts, respectively. If an /INT0 or /INT1 external interrupt is configured as edge-sensitive, the corresponding interrupt-pending flag is automatically cleared by the hardware when the CPU vectors to the ISR. When configured as level sensitive, the interrupt-pending flag follows the state of the external interrupt's input pin. The external interrupt source must hold the input active until the interrupt request is recognized. It must then deactivate the interrupt request before execution of the ISR completes or another interrupt request will be generated.

Table 12.4. Interrupt Summary

Interrupt Source	Interrupt Vector	Priority Order	Pending Flag	Bit addressable?	Cleared by HW?	SFRPAGE (SFRPGEN = 1)	Enable Flag	Priority Control
Reset	0x0000	Top	None	N/A	N/A	0	Always Enabled	Always Highest
External Interrupt 0 (/INT0)	0x0003	0	IE0 (TCON.1)	Y	Y	0	EX0 (IE.0)	PX0 (IP.0)
Timer 0 Overflow	0x000B	1	TF0 (TCON.5)	Y	Y	0	ET0 (IE.1)	PT0 (IP.1)
External Interrupt 1 (/INT1)	0x0013	2	IE1 (TCON.3)	Y	Y	0	EX1 (IE.2)	PX1 (IP.2)
Timer 1 Overflow	0x001B	3	TF1 (TCON.7)	Y	Y	0	ET1 (IE.3)	PT1 (IP.3)
UART0	0x0023	4	RI0 (SCON0.0) TI0 (SCON0.1)	Y		0	ES0 (IE.4)	PS0 (IP.4)
Timer 2	0x002B	5	TF2 (TMR2CN.7)	Y		0	ET2 (IE.5)	PT2 (IP.5)
Serial Peripheral Interface	0x0033	6	SPIF (SPI0CN.7) WCOL (SPI0CN.6) MODF (SPI0CN.5) RXOVRN (SPI0CN.4)	Y		0	ESPI0 (EIE1.0)	PSPI0 (EIP1.0)
SMBus Interface	0x003B	7	SI (SMB0CN.3)	Y		0	ESMB0 (EIE1.1)	PSMB0 (EIP1.1)
ADC0 Window Comparator	0x0043	8	AD0WINT (ADC0CN.2)	Y		0	EWADC0 (EIE1.2)	PWADC0 (EIP1.2)
Programmable Counter Array	0x004B	9	CF (PCA0CN.7) CCFn (PCA0CN.n)	Y		0	EPCA0 (EIE1.3)	PPCA0 (EIP1.3)
Comparator 0	0x0053	10	CP0FIF/CP0RIF (CPT0CN.4/.5)			1	CP0IE (EIE1.4)	PCP0 (EIP1.4)

14.4. External Crystal Example

If a crystal or ceramic resonator is used as an external oscillator source for the MCU, the circuit should be configured as shown in Figure 14.1, Option 1. The External Oscillator Frequency Control value (XFCN) should be chosen from the Crystal column of the table in SFR Definition 14.4 (OSCXCN register). For example, an 11.0592 MHz crystal requires an XFCN setting of 111b.

When the crystal oscillator is enabled, the oscillator amplitude detection circuit requires a settle time to achieve proper bias. Introducing a delay of at least 1 ms between enabling the oscillator and checking the XTLVLD bit will prevent a premature switch to the external oscillator as the system clock. Switching to the external oscillator before the crystal oscillator has stabilized can result in unpredictable behavior. The recommended procedure is:

Step 1. Enable the external oscillator in crystal oscillator mode.
Step 2. Wait at least 1 ms.
Step 3. Poll for XTLVLD => '1'.
Step 4. Switch the system clock to the external oscillator.

Note: Tuning-fork crystals may require additional settling time before XTLVLD returns a valid result.

The capacitors shown in the external crystal configuration provide the load capacitance required by the crystal for correct oscillation. These capacitors are "in series" as seen by the crystal and "in parallel" with the stray capacitance of the XTAL1 and XTAL2 pins.

Note: The load capacitance depends upon the crystal and the manufacturer. Please refer to the crystal data sheet when completing these calculations.

For example, a tuning-fork crystal of 32.768 kHz with a recommended load capacitance of 12.5 pF should use the configuration shown in Figure 14.1, Option 1. The total value of the capacitors and the stray capacitance of the XTAL pins should equal 25 pF. With a stray capacitance of 3 pF per pin, the 22 pF capacitors yield an equivalent capacitance of 12.5 pF across the crystal, as shown in Figure 14.2.

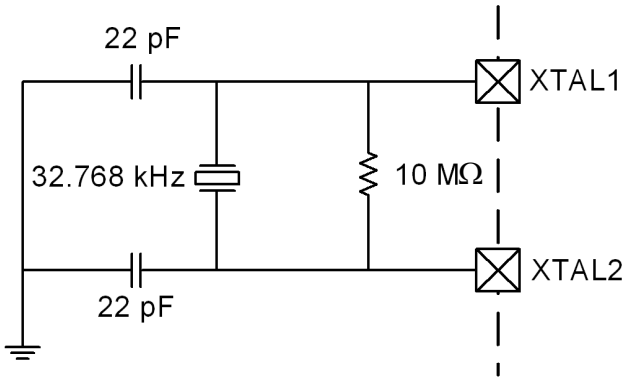

Figure 14.2. 32.768 kHz External Crystal Example

Important Note on External Crystals: Crystal oscillator circuits are quite sensitive to PCB layout. The crystal should be placed as close as possible to the XTAL pins on the device. The traces should be as short as possible and shielded with ground plane from any other traces which could introduce noise or interference.

SFR Definition 15.3. PSCTL: Program Store Read/Write Control

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

SFR Address: 0x8F
SFR Page: 0

Bits7-3: UNUSED. Read = 00000b, Write = don't care.

Bit2: SFLE: Scratchpad Flash Memory Access Enable
When this bit is set, Flash reads and writes from user software are directed to the 128-byte Scratchpad Flash sector. When SFLE is set to logic 1, Flash accesses out of the address range 0x00-0x7F should not be attempted. Reads/Writes out of this range will yield undefined results.
0: Flash access from user software directed to the Program/Data Flash sector.
1: Flash access from user software directed to the 128 byte Scratchpad sector.

Bit1: PSEE: Program Store Erase Enable.
Setting this bit allows an entire page of the Flash program memory to be erased provided the PSWE bit is also set. After setting this bit, a write to Flash memory using the MOVX instruction will erase the entire page that contains the location addressed by the MOVX instruction. The value of the data byte written does not matter. **Note: The Flash page containing the Read Lock Byte and Write/Erase Lock Bytes cannot be erased by software.**
0: Flash program memory erasure disabled.
1: Flash program memory erasure enabled.

Bit0: PSWE: Program Store Write Enable.
Setting this bit allows writing a byte of data to the Flash program memory using the MOVX write instruction. The location must be erased prior to writing data.
0: Write to Flash program memory disabled. MOVX write operations target External RAM.
1: Write to Flash program memory enabled. MOVX write operations target Flash memory.

16.4. Multiplexed and Non-multiplexed Selection

The External Memory Interface is capable of acting in a Multiplexed mode or a Non-multiplexed mode, depending on the state of the EMD2 (EMI0CF.4) bit.

16.4.1. Multiplexed Configuration

In Multiplexed mode, the Data Bus and the lower 8-bits of the Address Bus share the same Port pins: AD[7:0]. In this mode, an external latch (74HC373 or equivalent logic gate) is used to hold the lower 8-bits of the RAM address. The external latch is controlled by the ALE (Address Latch Enable) signal, which is driven by the External Memory Interface logic. An example of a Multiplexed Configuration is shown in Figure 16.1.

In Multiplexed mode, the external MOVX operation can be broken into two phases delineated by the state of the ALE signal. During the first phase, ALE is high and the lower 8-bits of the Address Bus are presented to AD[7:0]. During this phase, the address latch is configured such that the 'Q' outputs reflect the states of the 'D' inputs. When ALE falls, signaling the beginning of the second phase, the address latch outputs remain fixed and are no longer dependent on the latch inputs. Later in the second phase, the Data Bus controls the state of the AD[7:0] port at the time /RD or /WR is asserted.

See **Section "16.6.2. Multiplexed Mode" on page 199** for more information.

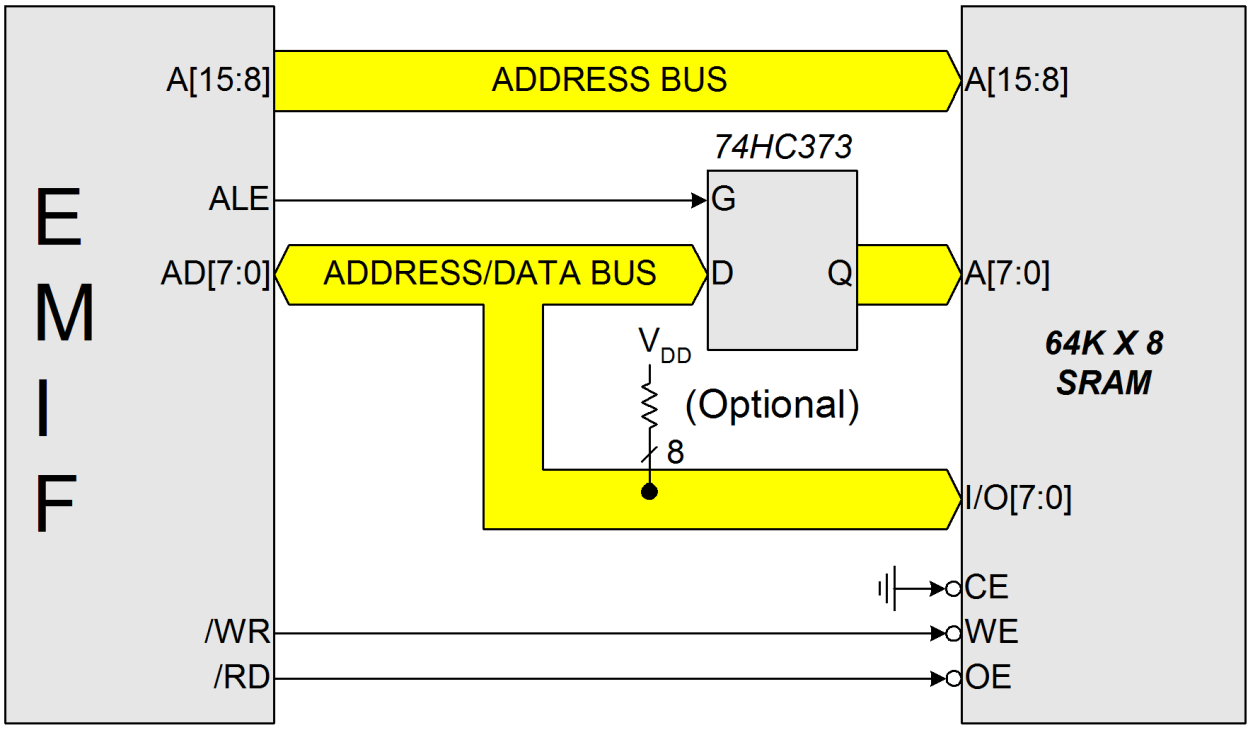

Figure 16.1. Multiplexed Configuration Example

19.4. SMBus Special Function Registers

The SMBus0 serial interface is accessed and controlled through five SFRs: SMB0CN Control Register, SMB0CR Clock Rate Register, SMB0ADR Address Register, SMB0DAT Data Register and SMB0STA Status Register. The five special function registers related to the operation of the SMBus0 interface are described in the following sections.

19.4.1. Control Register

The SMBus0 Control register SMB0CN is used to configure and control the SMBus0 interface. All of the bits in the register can be read or written by software. Two of the control bits are also affected by the SMBus0 hardware. The Serial Interrupt flag (SI, SMB0CN.3) is set to logic 1 by the hardware when a valid serial interrupt condition occurs. It can only be cleared by software. The Stop flag (STO, SMB0CN.4) is set to logic 1 by software. It is cleared to logic 0 by hardware when a STOP condition is detected on the bus.

Setting the ENSMB flag to logic 1 enables the SMBus0 interface. Clearing the ENSMB flag to logic 0 disables the SMBus0 interface and removes it from the bus. Momentarily clearing the ENSMB flag and then resetting it to logic 1 will reset SMBus0 communication. However, ENSMB should not be used to temporarily remove a device from the bus since the bus state information will be lost. Instead, the Assert Acknowledge (AA) flag should be used to temporarily remove the device from the bus (see description of AA flag below).

Setting the Start flag (STA, SMB0CN.5) to logic 1 will put SMBus0 in a master mode. If the bus is free, SMBus0 will generate a START condition. If the bus is not free, SMBus0 waits for a STOP condition to free the bus and then generates a START condition after a 5 µs delay per the SMB0CR value (In accordance with the SMBus protocol, the SMBus0 interface also considers the bus free if the bus is idle for 50 µs and no STOP condition was recognized). If STA is set to logic 1 while SMBus0 is in master mode and one or more bytes have been transferred, a repeated START condition will be generated.

When the Stop flag (STO, SMB0CN.4) is set to logic 1 while the SMBus0 interface is in master mode, the interface generates a STOP condition. In a slave mode, the STO flag may be used to recover from an error condition. In this case, a STOP condition is not generated on the bus, but the SMBus hardware behaves as if a STOP condition has been received and enters the "not addressed" slave receiver mode. Note that this simulated STOP will not cause the bus to appear free to SMBus0. The bus will remain occupied until a STOP appears on the bus or a Bus Free Timeout occurs. Hardware automatically clears the STO flag to logic 0 when a STOP condition is detected on the bus.

The Serial Interrupt flag (SI, SMB0CN.3) is set to logic 1 by hardware when the SMBus0 interface enters any one of the 28 possible states except the Idle state. If interrupts are enabled for the SMBus0 interface, an interrupt request is generated when the SI flag is set. The SI flag must be cleared by software.

Important Note: If SI is set to logic 1 while the SCL line is low, the clock-low period of the serial clock will be stretched and the serial transfer is suspended until SI is cleared to logic 0. A high level on SCL is not affected by the setting of the SI flag.

The Assert Acknowledge flag (AA, SMB0CN.2) is used to set the level of the SDA line during the acknowledge clock cycle on the SCL line. Setting the AA flag to logic 1 will cause an ACK (low level on SDA) to be sent during the acknowledge cycle if the device has been addressed. Setting the AA flag to logic 0 will cause a NACK (high level on SDA) to be sent during acknowledge cycle. After the transmission of a byte in slave mode, the slave can be temporarily removed from the bus by clearing the AA flag. The slave's own address and general call address will be ignored. To resume operation on the bus, the AA flag must be reset to logic 1 to allow the slave's address to be recognized.

Table 19.1. SMB0STA Status Codes and States

Mode	Status Code	SMBus State	Typical Action
MT/ MR	0x08	START condition transmitted.	Load SMB0DAT with Slave Address + R/W. Clear STA.
	0x10	Repeated START condition transmitted.	Load SMB0DAT with Slave Address + R/W. Clear STA.
Master Transmitter	0x18	Slave Address + W transmitted. ACK received.	Load SMB0DAT with data to be transmitted.
	0x20	Slave Address + W transmitted. NACK received.	Acknowledge poll to retry. Set STO + STA.
	0x28	Data byte transmitted. ACK received.	1) Load SMB0DAT with next byte, OR 2) Set STO, OR 3) Clear STO then set STA for repeated START.
	0x30	Data byte transmitted. NACK received.	1) Retry transfer OR 2) Set STO.
	0x38	Arbitration Lost.	Save current data.
Master Receiver	0x40	Slave Address + R transmitted. ACK received.	If only receiving one byte, clear AA (send NACK after received byte). Wait for received data.
	0x48	Slave Address + R transmitted. NACK received.	Acknowledge poll to retry. Set STO + STA.
	0x50	Data byte received. ACK transmitted.	Read SMB0DAT. Wait for next byte. If next byte is last byte, clear AA.
	0x58	Data byte received. NACK transmitted.	Set STO.

Table 22.3. Timer Settings for Standard Baud Rates Using an External 22.1184 MHz Oscillator

	Frequency: 22.1184 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)
	230400	0.00%	96	SYSCLK	XX	1	0xD0
	115200	0.00%	192	SYSCLK	XX	1	0xA0
	57600	0.00%	384	SYSCLK	XX	1	0x40
	28800	0.00%	768	SYSCLK / 12	00	0	0xE0
	14400	0.00%	1536	SYSCLK / 12	00	0	0xC0
	9600	0.00%	2304	SYSCLK / 12	00	0	0xA0
	2400	0.00%	9216	SYSCLK / 48	10	0	0xA0
	1200	0.00%	18432	SYSCLK / 48	10	0	0x40
	230400	0.00%	96	EXTCLK / 8	11	0	0xFA
	115200	0.00%	192	EXTCLK / 8	11	0	0xF4
	57600	0.00%	384	EXTCLK / 8	11	0	0xE8
	28800	0.00%	768	EXTCLK / 8	11	0	0xD0
	14400	0.00%	1536	EXTCLK / 8	11	0	0xA0
	9600	0.00%	2304	EXTCLK / 8	11	0	0x70

X = Don't care

***Note:** SCA1-SCA0 and T1M bit definitions can be found in **Section 23.1**.

SFR Definition 23.8. TMRnCN: Timer n Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
TFn	EXFn	-	-	EXENn	TRn	C/Tn	CP/RLn	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Bit Addressable

SFR Address: TMR2CN:0xC8;TMR3CN:0xC8;TMR4CN:0xC8
SFR Page: TMR2CN: page 0;TMR3CN: page 1;TMR4CN: page 2

Bit7: TFn: Timer n Overflow/Underflow Flag.
Set by hardware when either the Timer overflows from 0xFFFF to 0x0000, underflows from the value placed in RCAPnH:RCAPnL to 0xFFFF (in Auto-reload Mode), or underflows from 0x0000 to 0xFFFF (in Capture Mode). When the Timer interrupt is enabled, setting this bit causes the CPU to vector to the Timer interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit6: EXFn: Timer 2, 3, or 4 External Flag.
Set by hardware when either a capture or reload is caused by a high-to-low transition on the TnEX input pin and EXENn is logic 1. When the Timer interrupt is enabled, setting this bit causes the CPU to vector to the Timer Interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit5-4: Reserved.

Bit3: EXENn: Timer n External Enable.
Enables high-to-low transitions on TnEX to trigger captures, reloads, and control the direction of the timer/counter (up or down count). If DECEN = 1, TnEX will determine if the timer counts up or down when in Auto-reload Mode. If EXENn = 1, TnEX should be configured as a digital input.
0: Transitions on the TnEX pin are ignored.
1: Transitions on the TnEX pin cause capture, reload, or control the direction of timer count (up or down) as follows:
Capture Mode: '1'-to-'0' Transition on TnEX pin causes RCAPnH:RCAPnL to capture timer value.
Auto-Reload Mode:
DCEN = 0: '1'-to-'0' transition causes reload of timer and sets the EXFn Flag.
DCEN = 1: TnEX logic level controls direction of timer (up or down).

Bit2: TRn: Timer n Run Control.
This bit enables/disables the respective Timer.
0: Timer disabled.
1: Timer enabled and running/counting.

Bit1: C/Tn: Counter/Timer Select.
0: Timer Function: Timer incremented by clock defined by TnM1:TnM0 (TMRnCF.4:TMRnCF.3).
1: Counter Function: Timer incremented by high-to-low transitions on external input pin.

Bit0: CP/RLn: Capture/Reload Select.
This bit selects whether the Timer functions in capture or auto-reload mode.
0: Timer is in Auto-Reload Mode.
1: Timer is in Capture Mode.

24.3. Register Descriptions for PCA0

Following are detailed descriptions of the special function registers related to the operation of PCA0.

SFR Definition 24.1. PCA0CN: PCA Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
CF	CR	CCF5	CCF4	CCF3	CCF2	CCF1	CCF0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xD8
SFR Page: 0

Bit7: CF: PCA Counter/Timer Overflow Flag.
Set by hardware when the PCA0 Counter/Timer overflows from 0xFFFF to 0x0000. When the Counter/Timer Overflow (CF) interrupt is enabled, setting this bit causes the CPU to vector to the CF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit6: CR: PCA0 Counter/Timer Run Control.
This bit enables/disables the PCA0 Counter/Timer.
0: PCA0 Counter/Timer disabled.
1: PCA0 Counter/Timer enabled.

Bit5: CCF5: PCA0 Module 5 Capture/Compare Flag.
This bit is set by hardware when a match or capture occurs. When the CCF interrupt is enabled, setting this bit causes the CPU to vector to the CCF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit4: CCF4: PCA0 Module 4 Capture/Compare Flag.
This bit is set by hardware when a match or capture occurs. When the CCF interrupt is enabled, setting this bit causes the CPU to vector to the CCF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit3: CCF3: PCA0 Module 3 Capture/Compare Flag.
This bit is set by hardware when a match or capture occurs. When the CCF interrupt is enabled, setting this bit causes the CPU to vector to the CCF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit2: CCF2: PCA0 Module 2 Capture/Compare Flag.
This bit is set by hardware when a match or capture occurs. When the CCF interrupt is enabled, setting this bit causes the CPU to vector to the CCF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit1: CCF1: PCA0 Module 1 Capture/Compare Flag.
This bit is set by hardware when a match or capture occurs. When the CCF interrupt is enabled, setting this bit causes the CPU to vector to the CCF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit0: CCF0: PCA0 Module 0 Capture/Compare Flag.
This bit is set by hardware when a match or capture occurs. When the CCF interrupt is enabled, setting this bit causes the CPU to vector to the CCF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

25.3. Debug Support

Each MCU has on-chip JTAG and debug logic that provides non-intrusive, full speed, in-circuit debug support using the production part installed in the end application, via the four pin JTAG I/F. Silicon Labs' debug system supports inspection and modification of memory and registers, breakpoints, and single stepping. No additional target RAM, program memory, or communications channels are required. All the digital and analog peripherals are functional and work correctly (remain synchronized) while debugging. The Watchdog Timer (WDT) is disabled when the MCU is halted during single stepping or at a breakpoint.

The C8051F040DK is a development kit with all the hardware and software necessary to develop application code and perform in-circuit debug with each MCU in the C8051F04x family. Each kit includes an Integrated Development Environment (IDE) which has a debugger and integrated 8051 assembler. The kit also includes a JTAG interface module referred to as the Serial Adapter. There is also a target application board with a C8051F040 installed. The required cables and wall-mount power supply are also included.