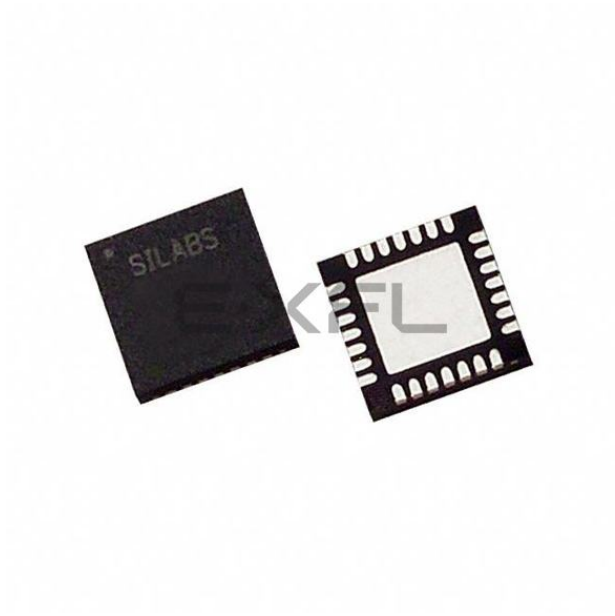


**Welcome to E-XFL.COM**

## 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 | Obsolete |
| Core Processor | 8051 |
| Core Size | 8-Bit |
| Speed | 25MHz |
| Connectivity | SMBus (2-Wire/I²C), SPI, UART/USART |
| Peripherals | POR, PWM, Temp Sensor, WDT |
| Number of I/O | 25 |
| Program Memory Size | 16KB (16K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 1.25K x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.7V ~ 3.6V |
| Data Converters | A/D 17x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 28-VFQFN Exposed Pad |
| Supplier Device Package | 28-QFN (5x5) |
| Purchase URL | https://www.e-xfl.com/product-detail/silicon-labs/c8051f311r |

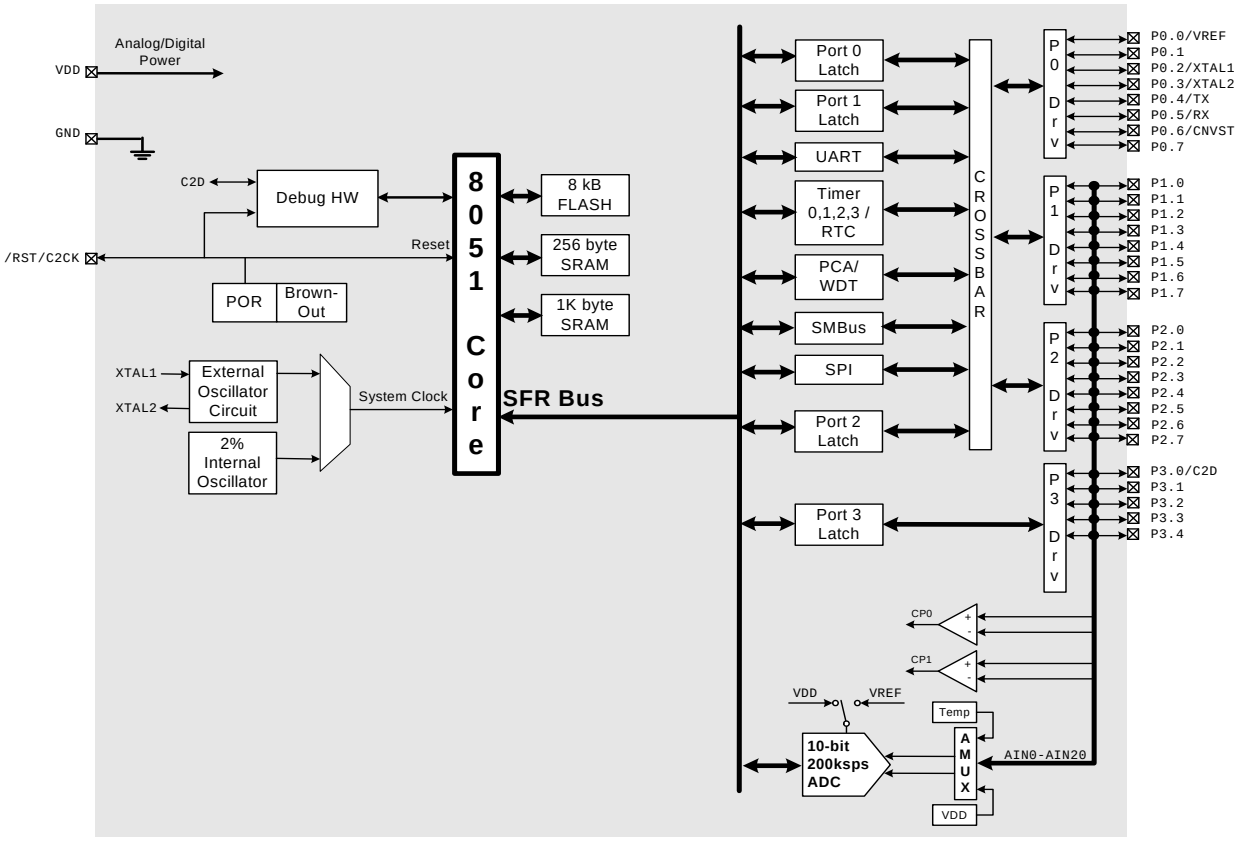


**Figure 1.3. C8051F312 Block Diagram**

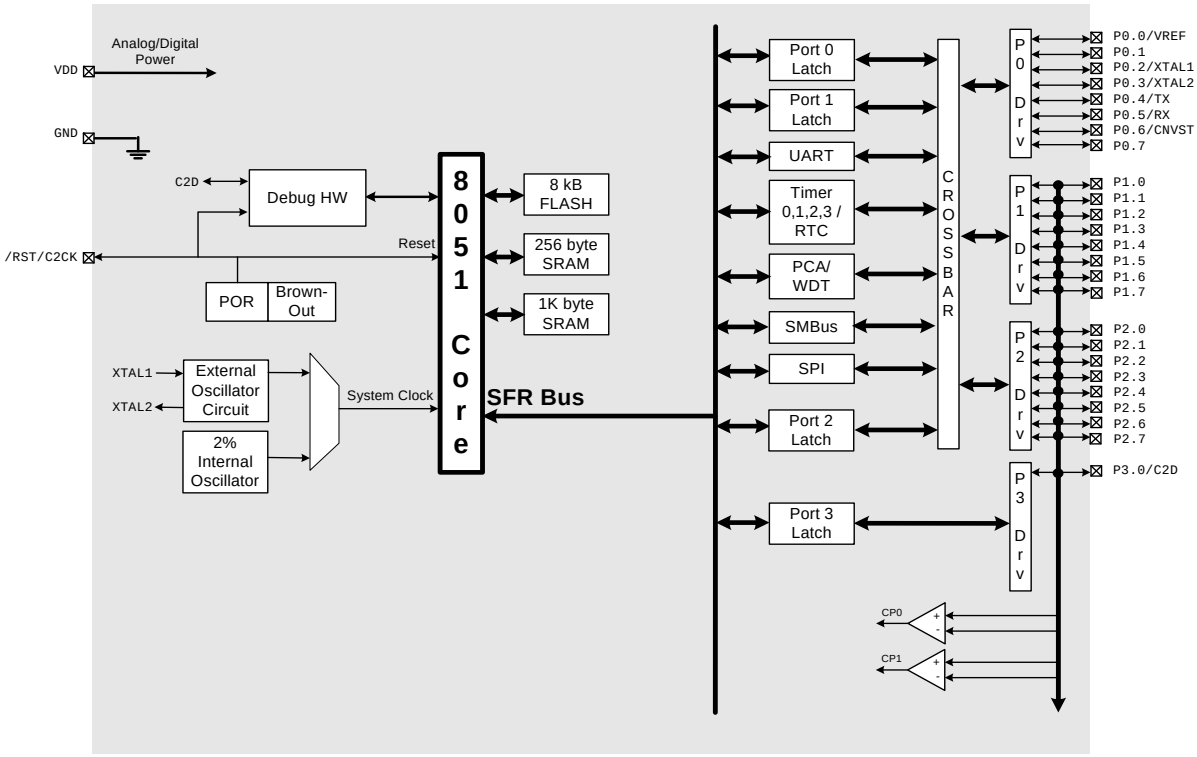


**Figure 1.6. C8051F315 Block Diagram**

## 1.3. On-Chip Debug Circuitry

The C8051F31x devices include on-chip Silicon Labs 2-Wire (C2) debug circuitry that provides non-intrusive, full speed, in-circuit debugging of the production part *installed in the end application.*

Silicon Labs' debugging system supports inspection and modification of memory and registers, breakpoints, and single stepping. No additional target RAM, program memory, timers, or communications channels are required. All the digital and analog peripherals are functional and work correctly while debugging. All the peripherals (except for the ADC and SMBus) are stalled when the MCU is halted, during single stepping, or at a breakpoint in order to keep them synchronized.

The C8051F310DK development kit provides all the hardware and software necessary to develop application code and perform in-circuit debugging with the C8051F31x MCUs. The kit includes software with a developer's studio and debugger, an integrated 8051 assembler, a debug adapter, a target application board with the associated MCU installed, and the required cables and wall-mount power supply.

The Silicon Labs IDE interface is a vastly superior developing and debugging configuration, compared to standard MCU emulators that use on-board "ICE Chips" and require the MCU in the application board to be socketed. Silicon Labs' debug paradigm increases ease of use and preserves the performance of the precision analog peripherals.

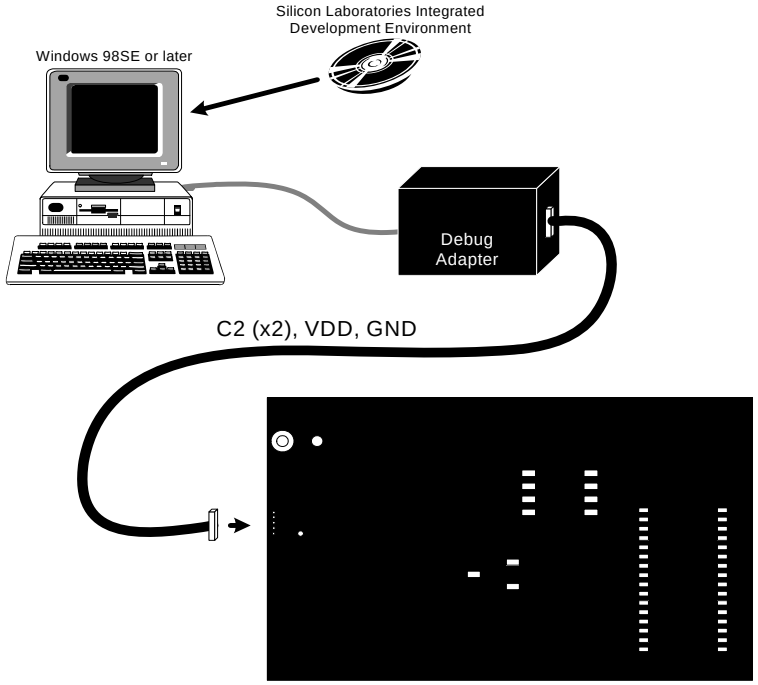


**Figure 1.12. Development/In-System Debug Diagram**

# 3. Global DC Electrical Characteristics

## Table 3.1. Global DC Electrical Characteristics

–40°C to +85°C, 25 MHz System Clock unless otherwise specified.

| Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| Digital Supply Voltage | | $V_{RST}$[1] | 3.0 | 3.6 | V |
| Digital Supply RAM Data Retention Voltage | | — | 1.5 | — | V |
| Specified Operating Temperature Range | | –40 | — | +85 | °C |
| SYSCLK (system clock frequency) | | 0[2] | — | 25 | MHz |
| Tsysl (SYSCLK low time) | | 18 | — | — | ns |
| Tsysh (SYSCLK high time) | | 18 | — | — | ns |
| **Digital Supply Current—CPU Active (Normal Mode, fetching instructions from Flash)** | | | | | |
| $I_{DD}$ (Note 3) | $V_{DD}$ = 3.0 V, F = 25 MHz | — | 7.8 | 8.6 | mA |
| | $V_{DD}$ = 3.0 V, F = 1 MHz | — | 0.38 | — | mA |
| | $V_{DD}$ = 3.0 V, F = 80 kHz | — | 31 | — | µA |
| | $V_{DD}$ = 3.6 V, F = 25 MHz | — | 10.7 | 12.1 | mA |
| $I_{DD}$ Supply Sensitivity (Note 3, Note 4) | F = 25 MHz | — | 67 | — | %/V |
| | F = 1 MHz | — | 62 | — | %/V |
| $I_{DD}$ Frequency Sensitivity (Note 3, Note 5) | $V_{DD}$ = 3.0 V, F ≤ 15 MHz, T = 25 ºC | — | 0.39 | — | mA/MHz |
| | $V_{DD}$ = 3.0 V, F > 15 MHz, T = 25 ºC | — | 0.21 | — | mA/MHz |
| | $V_{DD}$ = 3.6 V, F ≤ 15 MHz, T = 25 ºC | — | 0.55 | — | mA/MHz |
| | $V_{DD}$ = 3.6 V, F > 15 MHz, T = 25 ºC | — | 0.27 | — | mA/MHz |

**Notes:**

1. Given in Table 9.1 on page 110.
2. SYSCLK must be at least 32 kHz to enable debugging.
3. Based on device characterization data, not production tested.
4. Active and Inactive $I_{DD}$ at voltages and frequencies other than those specified can be calculated using the $I_{DD}$ Supply Sensitivity. For example, if the $V_{DD}$ is 3.3 V instead of 3.0 V at 25 MHz: $I_{DD}$ = 7.8 mA typical at 3.0 V and f = 25 MHz. From this, $I_{DD}$ = 7.8 mA + 0.67 x (3.3 V – 3.0 V) = 8 mA at 3.3 V and f = 25 MHz.
5. $I_{DD}$ can be estimated for frequencies ≤ 15 MHz by multiplying the frequency of interest by the frequency sensitivity number for that range. When using these numbers to estimate $I_{DD}$ for > 15 MHz, the estimate should be the current at 25 MHz minus the difference in current indicated by the frequency sensitivity number. For example:
   $V_{DD}$ = 3.0 V; F = 20 MHz, $I_{DD}$ = 7.8 mA – (25 MHz – 20 MHz) x 0.21 mA/MHz = 6.75 mA.
6. Idle $I_{DD}$ can be estimated for frequencies ≤ 1 MHz by multiplying the frequency of interest by the frequency sensitivity number for that range. When using these numbers to estimate Idle $I_{DD}$ for > 1 MHz, the estimate should be the current at 25 MHz minus the difference in current indicated by the frequency sensitivity number. For example:
   $V_{DD}$ = 3.0 V; F = 5 MHz, Idle $I_{DD}$ = 4.8 mA – (25 MHz – 5 MHz) x 0.15 mA/MHz = 1.8 mA.

Other electrical characteristics tables are found in the data sheet section corresponding to the associated peripherals. For more information on electrical characteristics for a specific peripheral, refer to the page indicated in Table 3.2.

**Table 3.2. Electrical Characteristics Quick Reference**

| Peripheral Electrical Characteristics | Page No. |
|---|---|
| ADC0 Electrical Characteristics | 65 |
| External Voltage Reference Circuit Electrical Characteristics | 68 |
| Comparator Electrical Characteristics | 78 |
| Reset Electrical Characteristics | 110 |
| Flash Electrical Characteristics | 112 |
| Internal Oscillator Electrical Characteristics | 123 |
| Port I/O DC Electrical Characteristics | 143 |

# 5. 10-Bit ADC (ADC0, C8051F310/1/2/3/6 only)

The ADC0 subsystem for the C8051F310/1/2/3/6 consists of two analog multiplexers (referred to collectively as AMUX0) with 25 total input selections, and a 200 ksps, 10-bit successive-approximation-register ADC with integrated track-and-hold and programmable window detector. The AMUX0, data conversion modes, and window detector are all configurable under software control via the Special Function Registers shown in Figure 5.1. ADC0 operates in both Single-ended and Differential modes, and may be configured to measure P1.0–P3.4, the Temperature Sensor output, or $V_{DD}$ with respect to P1.0–P3.4, VREF, or GND. The ADC0 subsystem is enabled only when the AD0EN bit in the ADC0 Control register (ADC0CN) is set to logic 1. The ADC0 subsystem is in low power shutdown when this bit is logic 0.

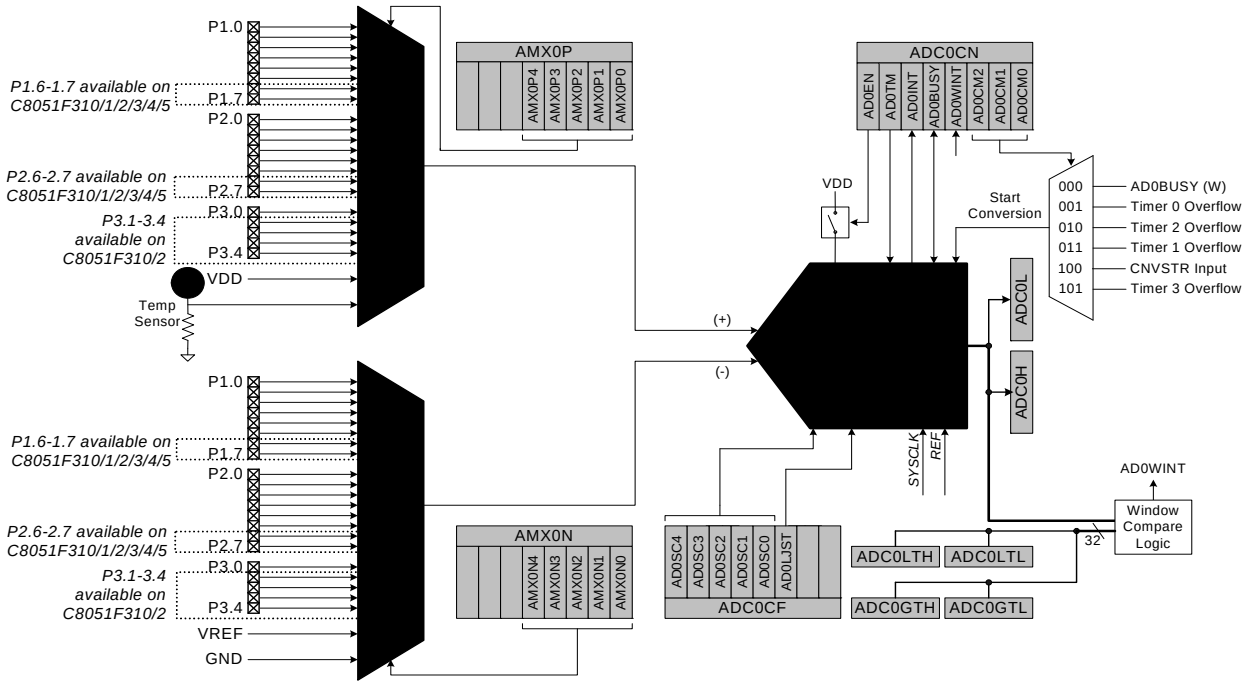


**Figure 5.1. ADC0 Functional Block Diagram**

## 5.1. Analog Multiplexer

AMUX0 selects the positive and negative inputs to the ADC. Any of the following may be selected as the positive input: P1.0-P3.4, the on-chip temperature sensor, or the positive power supply ($V_{DD}$). Any of the following may be selected as the negative input: P1.0-P3.4, VREF, or GND. **When GND is selected as the negative input, ADC0 operates in Single-ended Mode; all other times, ADC0 operates in Differential Mode.** The ADC0 input channels are selected in the AMX0P and AMX0N registers as described in SFR Definition 5.1 and SFR Definition 5.2.

The conversion code format differs between Single-ended and Differential modes. The registers ADC0H and ADC0L contain the high and low bytes of the output conversion code from the ADC at the completion of each conversion. Data can be right-justified or left-justified, depending on the setting of the AD0LJST bit (ADC0CN.0). When in Single-ended Mode, conversion codes are represented as 10-bit unsigned integers.

## 5.4. Programmable Window Detector

The ADC Programmable Window Detector continuously compares the ADC0 output registers to user-programmed limits, and notifies the system when a desired condition is detected. This is especially effective in an interrupt-driven system, saving code space and CPU bandwidth while delivering faster system response times. The window detector interrupt flag (AD0WINT in register ADC0CN) can also be used in polled mode. The ADC0 Greater-Than (ADC0GTH, ADC0GTL) and Less-Than (ADC0LTH, ADC0LTL) registers hold the comparison values. The window detector flag can be programmed to indicate when measured data is inside or outside of the user-programmed limits, depending on the contents of the ADC0 Less-Than and ADC0 Greater-Than registers.

### SFR Definition 5.7. ADC0GTH: ADC0 Greater-Than Data High Byte

| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | 11111111 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | SFR Address: |
| | | | | | | | | 0xC4 |

Bits7–0: High byte of ADC0 Greater-Than Data Word.

### SFR Definition 5.8. ADC0GTL: ADC0 Greater-Than Data Low Byte

| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | 11111111 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | SFR Address: |
| | | | | | | | | 0xC3 |

Bits7–0: Low byte of ADC0 Greater-Than Data Word.

### 5.4.2. Window Detector In Differential Mode

Figure 5.8 shows two example window comparisons for right-justified, differential data, with ADC0LTH:ADC0LTL = 0x0040 (+64d) and ADC0GTH:ADC0GTH = 0xFFFF (-1d). In differential mode, the measurable voltage between the input pins is between -VREF and VREF*(511/512). Output codes are represented as 10-bit 2's complement signed integers. In the left example, an AD0WINT interrupt will be generated if the ADC0 conversion word (ADC0H:ADC0L) is within the range defined by ADC0GTH:ADC0GTL and ADC0LTH:ADC0LTL (if 0xFFFF (-1d) < ADC0H:ADC0L < 0x0040 (64d)). In the right example, an AD0WINT interrupt will be generated if the ADC0 conversion word is outside of the range defined by the ADC0GT and ADC0LT registers (if ADC0H:ADC0L < 0xFFFF (-1d) or ADC0H:ADC0L > 0x0040 (+64d)). Figure 5.9 shows an example using left-justified data with the same comparison values.

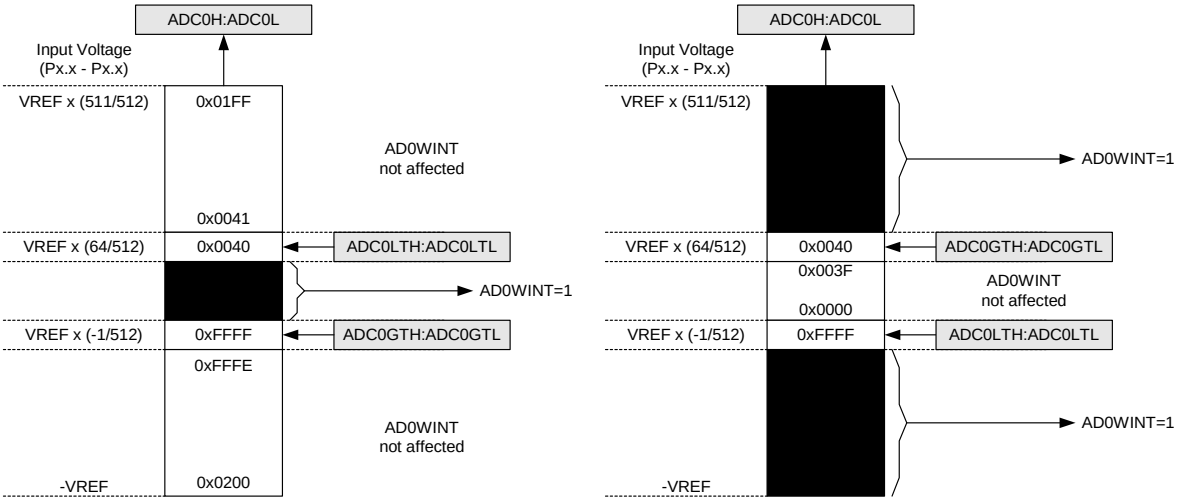


**Figure 5.8. ADC Window Compare Example: Right-Justified Differential Data**

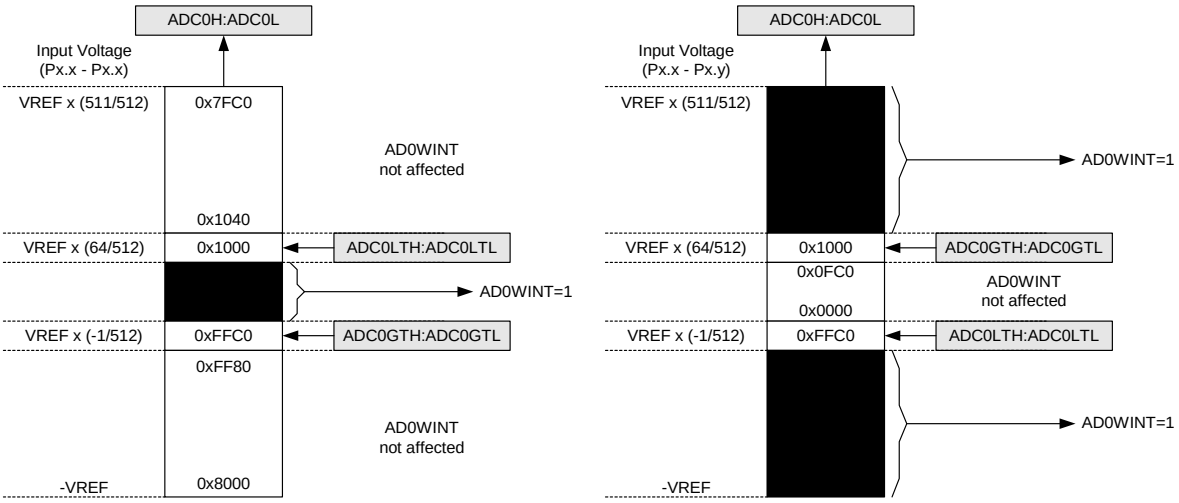


**Figure 5.9. ADC Window Compare Example: Left-Justified Differential Data**

## SFR Definition 8.8. IP: Interrupt Priority

| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| - | PSPI0 | PT2 | PS0 | PT1 | PX1 | PT0 | PX0 | 10000000 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | SFR Address: |
| | | | | | | | (bit addressable) | 0xB8 |

Bit7: UNUSED. Read = 1, Write = don't care.

Bit6: PSPI0: Serial Peripheral Interface (SPI0) Interrupt Priority Control.
This bit sets the priority of the SPI0 interrupt.
0: SPI0 interrupt set to low priority level.
1: SPI0 interrupt set to high priority level.

Bit5: PT2: Timer 2 Interrupt Priority Control.
This bit sets the priority of the Timer 2 interrupt.
0: Timer 2 interrupts set to low priority level.
1: Timer 2 interrupts set to high priority level.

Bit4: PS0: UART0 Interrupt Priority Control.
This bit sets the priority of the UART0 interrupt.
0: UART0 interrupts set to low priority level.
1: UART0 interrupts set to high priority level.

Bit3: PT1: Timer 1 Interrupt Priority Control.
This bit sets the priority of the Timer 1 interrupt.
0: Timer 1 interrupts set to low priority level.
1: Timer 1 interrupts set to high priority level.

Bit2: PX1: External Interrupt 1 Priority Control.
This bit sets the priority of the External Interrupt 1 interrupt.
0: External Interrupt 1 set to low priority level.
1: External Interrupt 1 set to high priority level.

Bit1: PT0: Timer 0 Interrupt Priority Control.
This bit sets the priority of the Timer 0 interrupt.
0: Timer 0 interrupt set to low priority level.
1: Timer 0 interrupt set to high priority level.

Bit0: PX0: External Interrupt 0 Priority Control.
This bit sets the priority of the External Interrupt 0 interrupt.
0: External Interrupt 0 set to low priority level.
1: External Interrupt 0 set to high priority level.

any other reset source. For example, if the $V_{DD}$ monitor is enabled and a software reset is performed, the $V_{DD}$ monitor will still be enabled after the reset.

**Important Note:** The $V_{DD}$ monitor must be enabled before it is selected as a reset source. Selecting the $V_{DD}$ monitor as a reset source before it is enabled and stabilized may cause a system reset. The procedure for configuring the $V_{DD}$ monitor as a reset source is shown below:

Step 1. Enable the $V_{DD}$ monitor (VDMEN bit in VDM0CN = '1').
Step 2. Wait for the $V_{DD}$ monitor to stabilize (see Table 9.1 for the $V_{DD}$ Monitor turn-on time).
Note: This delay should be omitted if software contains routines that erase or write Flash memory.
Step 3. Select the $V_{DD}$ monitor as a reset source (PORSF bit in RSTSRC = '1').

See Figure 9.2 for $V_{DD}$ monitor timing; note that the reset delay is not incurred after a $V_{DD}$ monitor reset. See Table 9.1 for complete electrical characteristics of the $V_{DD}$ monitor.

### SFR Definition 9.1. VDM0CN: $V_{DD}$ Monitor Control

| R/W | R | R | R | R | R | R | R | Reset Value |
|---|---|---|---|---|---|---|---|---|
| VDMEN | VDDSTAT | Reserved | Reserved | Reserved | Reserved | Reserved | Reserved | Variable |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

SFR Address: 0xFF

Bit7: VDMEN: $V_{DD}$ Monitor Enable.
This bit is turns the $V_{DD}$ monitor circuit on/off. The $V_{DD}$ Monitor cannot generate system resets until it is also selected as a reset source in register RSTSRC (Figure 9.2). The $V_{DD}$ Monitor must be allowed to stabilize before it is selected as a reset source. **Selecting the $V_{DD}$ monitor as a reset source before it has stabilized may generate a system reset.** See Table 9.1 for the minimum $V_{DD}$ Monitor turn-on time.
0: $V_{DD}$ Monitor Disabled.
1: $V_{DD}$ Monitor Enabled.

Bit6: $V_{DD}$ STAT: $V_{DD}$ Status.
This bit indicates the current power supply status ($V_{DD}$ Monitor output).
0: $V_{DD}$ is at or below the $V_{DD}$ monitor threshold.
1: $V_{DD}$ is above the $V_{DD}$ monitor threshold.

Bits5–0: Reserved. Read = Variable. Write = don't care.

## 9.3. External Reset

The external $\overline{\text{RST}}$ pin provides a means for external circuitry to force the device into a reset state. Asserting an active-low signal on the $\overline{\text{RST}}$ pin generates a reset; an external pullup and/or decoupling of the $\overline{\text{RST}}$ pin may be necessary to avoid erroneous noise-induced resets. See Table 9.1 for complete $\overline{\text{RST}}$ pin specifications. The PINRSF flag (RSTSRC.0) is set on exit from an external reset.

## NOTES:

**SFR Definition 13.2. XBR1: Port I/O Crossbar Register 1**

| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| WEAKPUD | XBARE | T1E | T0E | ECIE | PCA0ME | | | 00000000 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | SFR Address: 0xE2 |

Bit7: WEAKPUD: Port I/O Weak Pullup Disable.
0: Weak Pullups enabled (except for Ports whose I/O are configured as analog input).
1: Weak Pullups disabled.
Bit6: XBARE: Crossbar Enable.
0: Crossbar disabled.
1: Crossbar enabled.
Bit5: T1E: T1 Enable
0: T1 unavailable at Port pin.
1: T1 routed to Port pin.
Bit4: T0E: T0 Enable
0: T0 unavailable at Port pin.
1: T0 routed to Port pin.
Bit3: ECIE: PCA0 External Counter Input Enable
0: ECI unavailable at Port pin.
1: ECI routed to Port pin.
Bits2–0: PCA0ME: PCA Module I/O Enable Bits.
000: All PCA I/O unavailable at Port pins.
001: CEX0 routed to Port pin.
010: CEX0, CEX1 routed to Port pins.
011: CEX0, CEX1, CEX2 routed to Port pins.
100: CEX0, CEX1, CEX2, CEX3 routed to Port pins.
101: CEX0, CEX1, CEX2, CEX3, CEX4 routed to Port pins.

## 13.3. General Purpose Port I/O

Port pins that remain unassigned by the Crossbar and are not used by analog peripherals can be used for general purpose I/O. Ports3-0 are accessed through corresponding special function registers (SFRs) that are both byte addressable and bit addressable. When writing to a Port, the value written to the SFR is latched to maintain the output data value at each pin. When reading, the logic levels of the Port's input pins are returned regardless of the XBRn settings (i.e., even when the pin is assigned to another signal by the Crossbar, the Port register can always read its corresponding Port I/O pin). The exception to this is the execution of the read-modify-write instructions. The read-modify-write instructions when operating on a Port SFR are the following: ANL, ORL, XRL, JBC, CPL, INC, DEC, DJNZ and MOV, CLR or SET, when the destination is an individual bit in a Port SFR. For these instructions, the value of the register (not the pin) is read, modified, and written back to the SFR.

### SFR Definition 13.3. P0: Port0

| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| P0.7 | P0.6 | P0.5 | P0.4 | P0.3 | P0.2 | P0.1 | P0.0 | 11111111 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | SFR Address: |
| | | | | | | | (bit addressable) | 0x80 |

Bits7–0: P0.[7:0]
Write - Output appears on I/O pins per Crossbar Registers.
0: Logic Low Output.
1: Logic High Output (high impedance if corresponding P0MDOUT.n bit = 0).
Read - Always reads '1' if selected as analog input in register P0MDIN. Directly reads Port pin when configured as digital input.
0: P0.n pin is logic low.
1: P0.n pin is logic high.

### SFR Definition 13.4. P0MDIN: Port0 Input Mode

| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | 11111111 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | SFR Address: |
| | | | | | | | | 0xF1 |

Bits7–0: Analog Input Configuration Bits for P0.7–P0.0 (respectively).
Port pins configured as analog inputs have their weak pullup, digital driver, and digital receiver disabled.
0: Corresponding P0.n pin is configured as an analog input.
1: Corresponding P0.n pin is not configured as an analog input.

The direction bit (R/W) occupies the least-significant bit position of the address byte. The direction bit is set to logic 1 to indicate a "READ" operation and cleared to logic 0 to indicate a "WRITE" operation.

All transactions are initiated by a master, with one or more addressed slave devices as the target. The master generates the START condition and then transmits the slave address and direction bit. If the transaction is a WRITE operation from the master to the slave, the master transmits the data a byte at a time waiting for an ACK from the slave at the end of each byte. For READ operations, the slave transmits the data waiting for an ACK from the master at the end of each byte. At the end of the data transfer, the master generates a STOP condition to terminate the transaction and free the bus. Figure 14.3 illustrates a typical SMBus transaction.

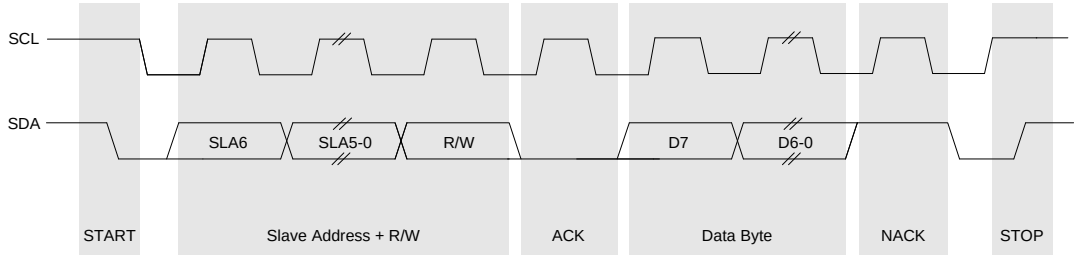


**Figure 14.3. SMBus Transaction**

### 14.3.1. Arbitration

A master may start a transfer only if the bus is free. The bus is free after a STOP condition or after the SCL and SDA lines remain high for a specified time (see **Section “14.3.4. SCL High (SMBus Free) Timeout” on page 148**). In the event that two or more devices attempt to begin a transfer at the same time, an arbitration scheme is employed to force one master to give up the bus. The master devices continue transmitting until one attempts a HIGH while the other transmits a LOW. Since the bus is open-drain, the bus will be pulled LOW. The master attempting the HIGH will detect a LOW SDA and lose the arbitration. The winning master continues its transmission without interruption; the losing master becomes a slave and receives the rest of the transfer if addressed. This arbitration scheme is non-destructive: one device always wins, and no data is lost.

## SFR Definition 15.1. SCON0: Serial Port 0 Control

| R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| S0MODE | | MCE0 | REN0 | TB80 | RB80 | TI0 | RI0 | 01000000 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | Bit Addressable |

SFR Address: 0x98

Bit7: S0MODE: Serial Port 0 Operation Mode.
This bit selects the UART0 Operation Mode.
0: 8-bit UART with Variable Baud Rate.
1: 9-bit UART with Variable Baud Rate.

Bit6: UNUSED. Read = 1b. Write = don't care.

Bit5: MCE0: Multiprocessor Communication Enable.
The function of this bit is dependent on the Serial Port 0 Operation Mode.
S0MODE = 0: Checks for valid stop bit.
0: Logic level of stop bit is ignored.
1: RI0 will only be activated if stop bit is logic level 1.
S0MODE = 1: Multiprocessor Communications Enable.
0: Logic level of ninth bit is ignored.
1: RI0 is set and an interrupt is generated only when the ninth bit is logic 1.

Bit4: REN0: Receive Enable.
This bit enables/disables the UART receiver.
0: UART0 reception disabled.
1: UART0 reception enabled.

Bit3: TB80: Ninth Transmission Bit.
The logic level of this bit will be assigned to the ninth transmission bit in 9-bit UART Mode. It is not used in 8-bit UART Mode. Set or cleared by software as required.

Bit2: RB80: Ninth Receive Bit.
RB80 is assigned the value of the STOP bit in Mode 0; it is assigned the value of the 9th data bit in Mode 1.

Bit1: TI0: Transmit Interrupt Flag.
Set by hardware when a byte of data has been transmitted by UART0 (after the 8th bit in 8-bit UART Mode, or at the beginning of the STOP bit in 9-bit UART Mode). When the UART0 interrupt is enabled, setting this bit causes the CPU to vector to the UART0 interrupt service routine. This bit must be cleared manually by software.

Bit0: RI0: Receive Interrupt Flag.
Set to '1' by hardware when a byte of data has been received by UART0 (set at the STOP bit sampling time). When the UART0 interrupt is enabled, setting this bit to '1' causes the CPU to vector to the UART0 interrupt service routine. This bit must be cleared manually by software.

## 16.5. 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 an active-high or active-low clock. 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 for master mode are shown in Figure 16.5. For slave mode, the clock and data relationships are shown in Figure 16.6 and Figure 16.7. CKPHA must be set to '0' on both the master and slave SPI when communicating between two of the following devices: C8051F04x, C8051F06x, C8051F12x, C8051F31x, C8051F32x, and C8051F33x

The SPI0 Clock Rate Register (SPI0CKR) as shown in SFR Definition 16.3 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, whichever is slower. When the SPI is configured as a slave, the maximum data transfer rate (bits/sec) for full-duplex operation is 1/10 the system clock frequency, provided that the master issues SCK, NSS (in 4-wire slave mode), and the serial input data synchronously with the slave's system clock. If the master issues SCK, NSS, and the serial input data asynchronously, the maximum data transfer rate (bits/sec) must be less than 1/10 the system clock frequency. In the special case where the master only wants to transmit data to the slave and does not need to receive data from the slave (i.e. half-duplex operation), the SPI slave can receive data at a maximum data transfer rate (bits/sec) of 1/4 the system clock frequency. This is provided that the master issues SCK, NSS, and the serial input data synchronously with the slave's system clock.

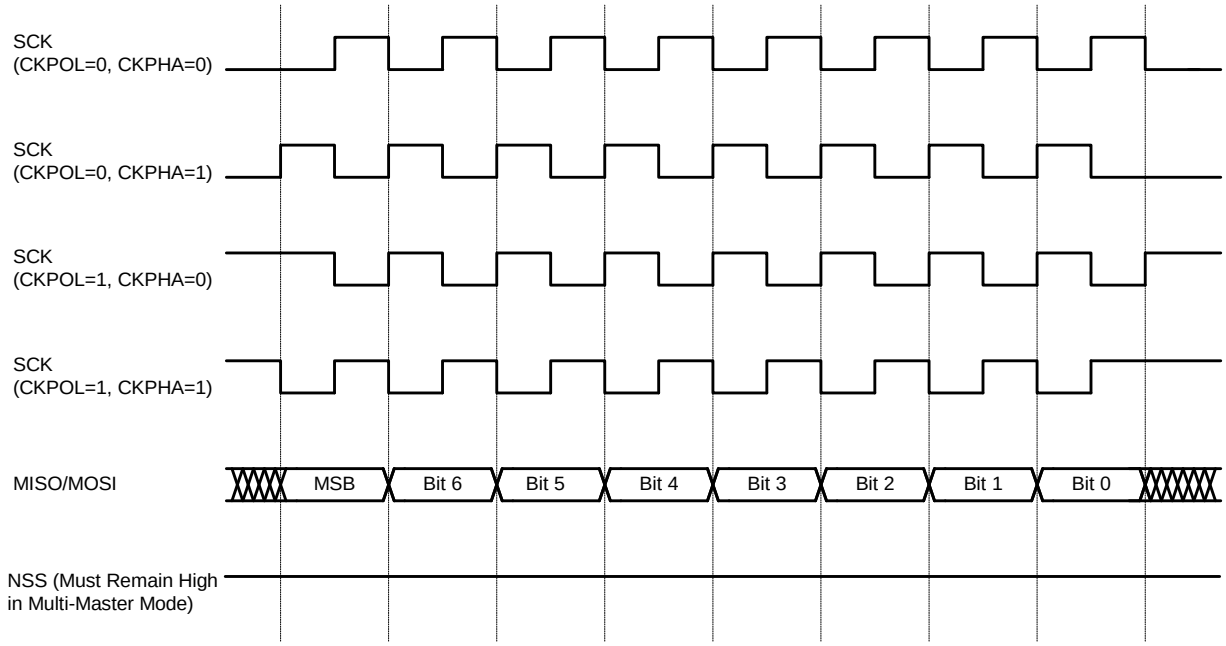


**Figure 16.5. Master Mode Data/Clock Timing**

## 18.1. 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 18.1. **Note that in 'External oscillator source divided by 8' mode, the external oscillator source is synchronized with the system clock, and must have a frequency less than or equal to the system clock.**

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 18.1. 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 oscillator source divided by 8 is synchronized with the system clock. | | | |

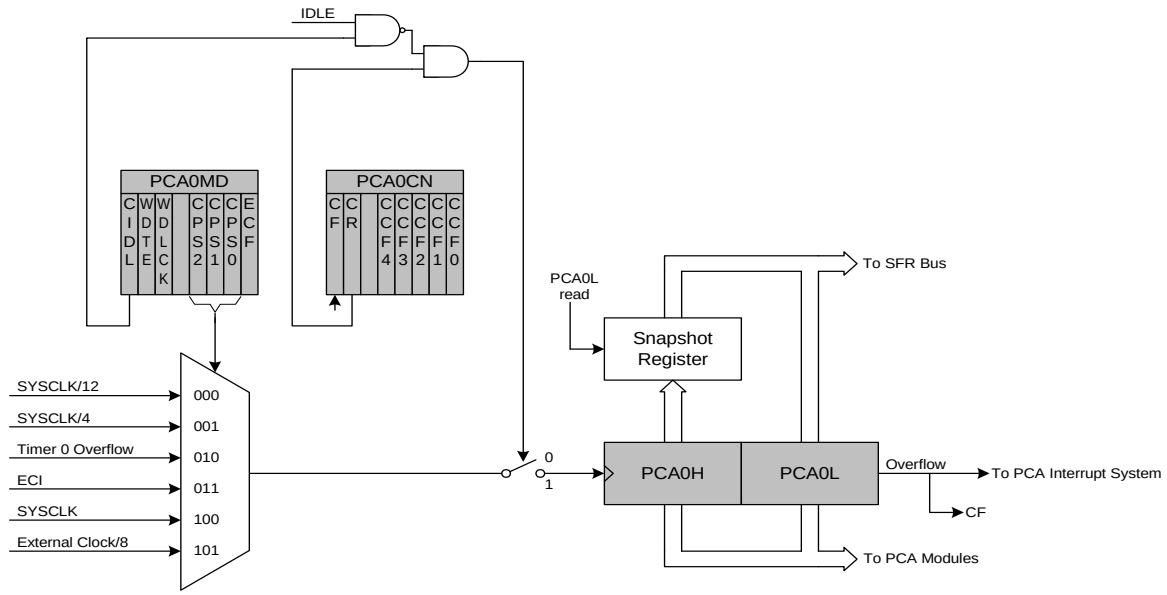


**Figure 18.2. PCA Counter/Timer Block Diagram**

### 18.2.3. High-Speed Output Mode

In High Speed Output mode, a module's associated CEXn pin is toggled each time a match occurs between the PCA Counter and the module's 16-bit capture/compare register (PCA0CPHn and PCA0CPLn) Setting the TOGn, MATn, and ECOMn bits in the PCA0CPMn register enables the High-Speed Output mode.

**Important Note About Capture/Compare Registers**: When writing a 16-bit value to the PCA0 Capture/ Compare registers, the low byte should always be written first. Writing to PCA0CPLn clears the ECOMn bit to '0'; writing to PCA0CPHn sets ECOMn to '1'.

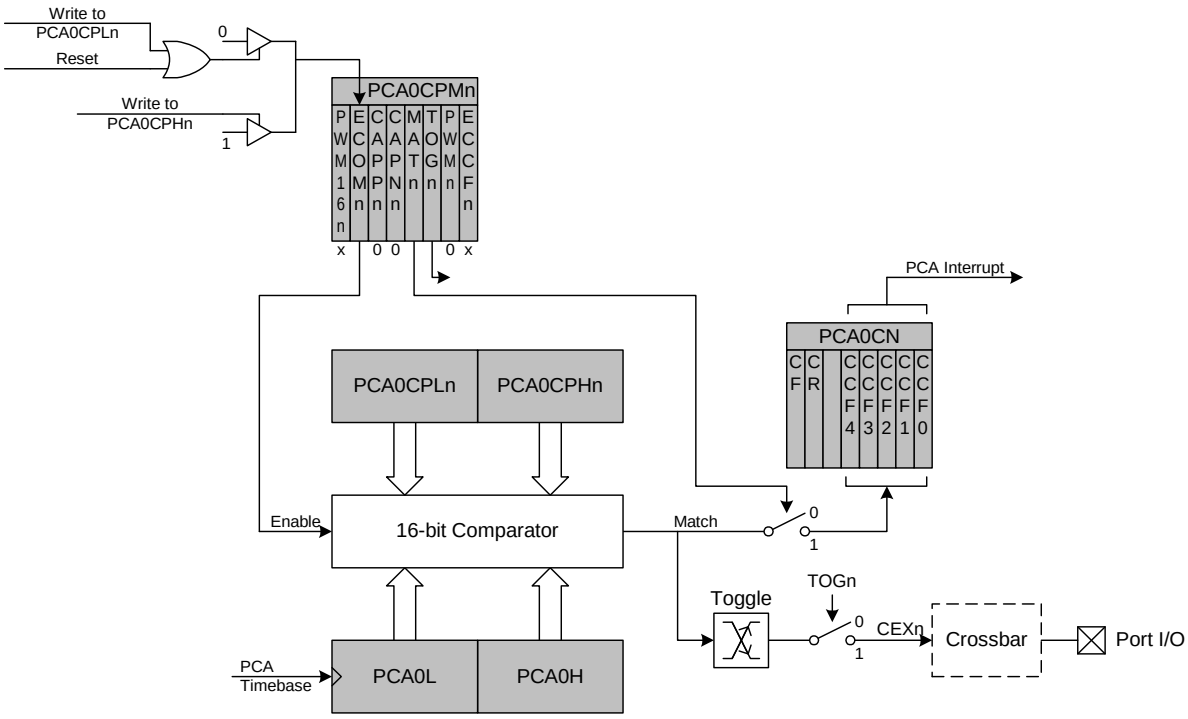


**Figure 18.6. PCA High Speed Output Mode Diagram**

## 18.4. Register Descriptions for PCA

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

### SFR Definition 18.1. PCA0CN: PCA Control

| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| CF | CR | | CCF4 | CCF3 | CCF2 | CCF1 | CCF0 | 00000000 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | SFR Address: |
| | | | | | | | (bit addressable) | 0xD8 |

Bit7: CF: PCA Counter/Timer Overflow Flag.
Set by hardware when the PCA 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 PCA interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

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

Bit5: UNUSED. Read = 0b, Write = don't care.

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

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

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

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

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

## NOTES: