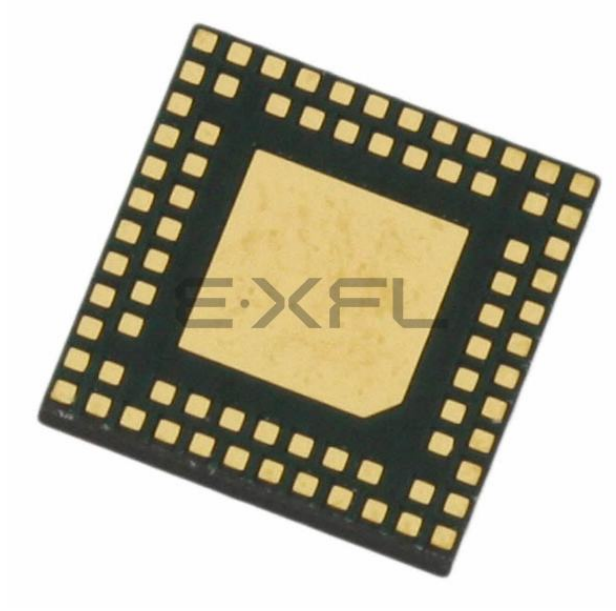

**Welcome to E-XFL.COM**

## 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 | I²C, SPI, UART/USART |
| Peripherals | DMA, POR, PWM, WDT |
| Number of I/O | 57 |
| Program Memory Size | 128KB (128K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 8.25K x 8 |
| Voltage - Supply (Vcc/Vdd) | 1.8V ~ 3.8V |
| Data Converters | A/D 16x10b/12b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 76-VFQFN Dual Rows, Exposed Pad |
| Supplier Device Package | 76-DQFN (6x6) |
| Purchase URL | https://www.e-xfl.com/product-detail/silicon-labs/c8051f962-a-gmr |

**Table 3.1. Pin Definitions for the C8051F96x (Continued)**

| Name | Pin Numbers | | | Type | Description |
|---|---|---|---|---|---|
| | DQFN76 | TQFP80 | QFN40 | | |
| P0.6<br><br>CNVSTR | A38 | 76 | 38 | D I/O or A In<br><br>D In | Port 0.6. See Port I/O Section for a complete description.<br><br>External Convert Start Input for ADC0. See ADC0 section for a complete description. |
| P0.7<br><br>IREF0 | A37 | 74 | 37 | D I/O or A In<br>A Out | Port 0.7. See Port I/O Section for a complete description.<br><br>IREF0 Output. See IREF Section for complete description. |
| P1.0<br><br>PC0 | A36 | 72 | 36 | D I/O or A In<br><br>D I/O | Port 1.0. See Port I/O Section for a complete description.<br><br>Pulse Counter 0. |
| P1.1<br><br>PC1 | A35 | 70 | 35 | D I/O or A In<br><br>D I/O | Port 1.1. See Port I/O Section for a complete description.<br><br>Pulse Counter 1. |
| P1.2<br><br>XTAL3 | A34 | 67 | 34 | D I/O or A In<br><br>A In | Port 1.2. See Port I/O Section for a complete description.<br><br>SmaRTClock Oscillator Crystal Input. |
| P1.3<br><br>XTAL4 | A33 | 65 | 33 | D I/O or A In<br><br>A Out | Port 1.3. See Port I/O Section for a complete description.<br><br>SmaRTClock Oscillator Crystal Output. |
| P1.4 | A31 | 60 | 31 | D I/O or A In | Port 1.4. See Port I/O Section for a complete description. |
| P1.5 | A30 | 57 | 30 | D I/O or A In | Port 1.5. See Port I/O Section for a complete description. VIORF supply. |
| P1.6 | A29 | 56 | 29 | D I/O or A In | Port 1.6. See Port I/O Section for a complete description. VIORF supply. May also be used as INT0 or INT1. |
| P1.7 | A28 | 54 | 28 | D I/O or A In | Port 1.7. See Port I/O Section for a complete description. VIORF supply. May also be used as INT0 or INT1. |
| P2.0 | A27 | 53 | 27 | D I/O or A In | Port 2.0. See Port I/O Section for a complete description. VIORF supply. May also be used as SCK for SPI1. |

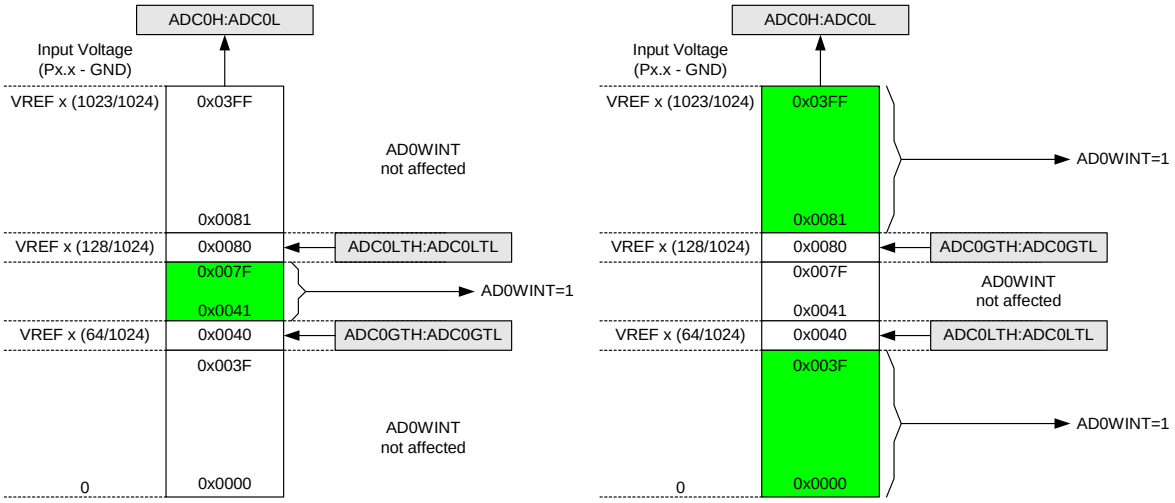


**Figure 5.5. ADC Window Compare Example: Right-Justified Single-Ended Data**

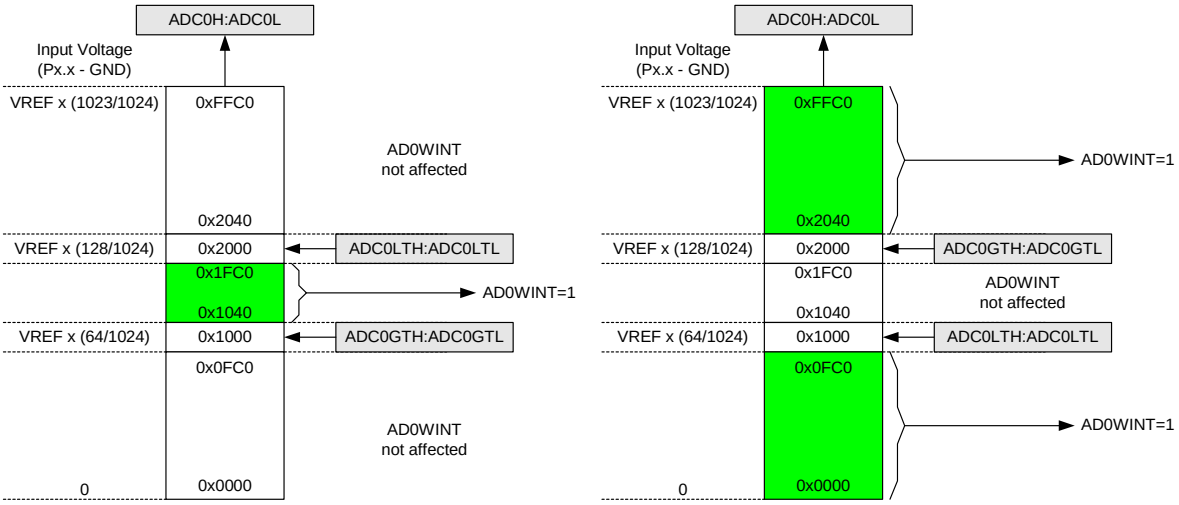


**Figure 5.6. ADC Window Compare Example: Left-Justified Single-Ended Data**

### 5.6.2. ADC0 Specifications

See “4. Electrical Characteristics” on page 56 for a detailed listing of ADC0 specifications.

## SFR Definition 10.2. EMI0CF: External Memory Configuration

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | | | | EMD2 | EMD[1:0] | | EALE[1:0] | |
| Type | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

SFR Page = 0x0; SFR Address = 0xAB

| Bit | Name | Function |
|---|---|---|
| 7:5 | Unused | Read = 000b; Write = Don't Care. |
| 4 | EMD2 | **EMIF Multiplex Mode Select Bit.**<br>0: EMIF operates in multiplexed address/data mode<br>1: EMIF operates in non-multiplexed mode (separate address and data pins) |
| 3:2 | EMD[1:0] | **EMIF Operating Mode Select Bits.**<br>00: Internal Only: MOVX accesses on-chip XRAM only. All effective addresses alias to on-chip memory space<br>01: Split Mode without Bank Select: Accesses below the 8 kB boundary are directed on-chip. Accesses above the 8 kB boundary are directed off-chip. 8-bit off-chip MOVX operations use current contents of the Address high port latches to resolve the upper address byte. To access off chip space, EMI0CN must be set to a page that is not contained in the on-chip address space.<br>10: Split Mode with Bank Select: Accesses below the 8 kB boundary are directed on-chip. Accesses above the 8 kB boundary are directed off-chip. 8-bit off-chip MOVX operations uses the contents of EMI0CN to determine the high-byte of the address.<br>11: External Only: MOVX accesses off-chip XRAM only. On-chip XRAM is not visible to the CPU. |
| 1:0 | EALE[1:0] | **ALE Pulse-Width Select Bits.**<br>These bits only have an effect when EMD2 = 0.<br>00: ALE high and ALE low pulse width = 1 SYSCLK cycle.<br>01: ALE high and ALE low pulse width = 2 SYSCLK cycles.<br>10: ALE high and ALE low pulse width = 3 SYSCLK cycles.<br>11: ALE high and ALE low pulse width = 4 SYSCLK cycles. |

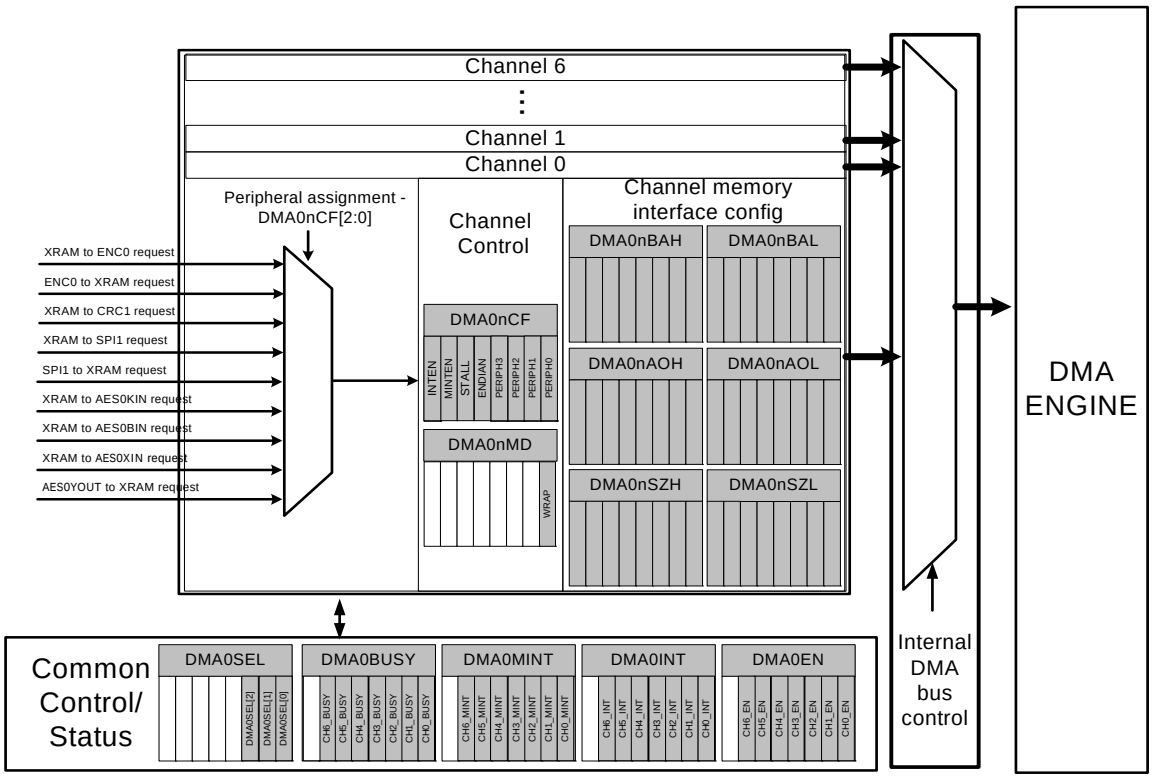


**Figure 11.1. DMA0 Block Diagram**

## 11.1. DMA0 Architecture

The first step in configuring a DMA0 channel is to select the desired channel for data transfer using DMA0-SEL[2:0] bits (DMA0SEL). After setting the DMA0 channel, firmware can address channel-specific registers such as DMA0NCF, DMA0NBAH/L, DMA0NAOH/L, and DMA0NSZH/L. Once firmware selects a channel, the subsequent SFR configuration applies to the DMA0 transfer of that selected channel.

Each DMA0 channel consists of an SFR assigning the channel to a peripheral, a channel control register and a set of SFRs that describe XRAM and SFR addresses to be used during data transfer (See Figure 11.1). The peripheral assignment bits of DMA0nCF select one of the eight data transfer functions. The selected channel can choose the desired function by writing to the PERIPH[2:0] bits (DMA0NCF[2:0]).

The control register DMA0NCF of each channel configures the endian-ness of the data in XRAM, stall enable, full-length interrupt enable and mid-point interrupt enable. When a channel is stalled by setting the STALL bit (DMA0NCF.5), DMA0 transfers in progress will not be aborted, but new DMA0 transfers will be blocked until the stall status of the channel is reset. After the stall bit is set, software should poll the corresponding DMA0BUSY to verify that there are no more DMA transfers for that channel.

The memory interface configuration SFRs of a channel define the linear region of XRAM involved in the transfer through a 12-bit base address register DMA0NBAH:L, a 10-bit address offset register DMA0NAOH:L and a 10-bit data transfer size DMA0NSZH:L. The effective memory address is the address involved in the current DMA0 transaction.

Effective Memory Address = Base Address + Address Offset

The address offset serves as byte counter. The address offset should be always less than data transfer length. The address offset increments by one after each byte transferred. For DMA0 configuration of any channel, address offsets of active channels should be reset to 0 before DMA0 transfers occur.

## SFR Definition 13.5. CRC1OUTL: CRC1 Output LSB

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | CRC1OUTL[7:0] | | | | | | | |
| Type | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Page = 0x2; SFR Address = 0xBA; Not Bit-Addressable

| Bit | Name | Function |
|---|---|---|
| 7:0 | CRC1OUTL[7:0] | **CRC1 Output LSB** |

## SFR Definition 13.6. CRC1OUTH: CRC1 Output MSB

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | CRC1OUTH[7:0] | | | | | | | |
| Type | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Page = 0x2; SFR Address = 0xBB; Not Bit-Addressable

| Bit | Name | Function |
|---|---|---|
| 7:0 | CRC1OUTH[7:0] | **CRC1 Output MSB.** |

## SFR Definition 15.1. ENC0CN: Encoder Decoder 0 Control

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | READY | ERROR | ENC | DEC | | DMA | ENDIAN | MODE |
| Type | R | R | R/W | R/W | R | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Address = 0xC5; SFR page = 0x2; Not bit-Addressable

| Bit | Name | Function |
|---|---|---|
| 7 | READY | **Ready Flag.** |
| 6 | ERROR | **Error Flag.** |
| 5 | ENC | **Encode.**<br>Setting this bit will initiate an Encode operation. |
| 4 | DEC | **Decode.**<br>Setting this bit will initiate a Decode operation. |
| 2 | DMA | **DMA Mode Enable.**<br>This bit should be set when using the encoder/decoder with the DMA. |
| 1 | ENDIAN | **Big-Endian DMA Mode Select.**<br>This bit should be set when using the DMA with big-endian multiple byte DMA transfers. The DMA must also be configured for the same endian mode. |
| 0 | MODE | **Mode.**<br>0: Select Manchester encoding or decoding.<br>1:Select Three-out-of-Six encoding or decoding. |

## SFR Definition 15.2. ENC0L: ENC0 Data Low Byte

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | ENC0L[7:0] | | | | | | | |
| Type | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Page = 0x2; SFR Address = 0xC2; Bit-Addressable

| Bit | Name | Function |
|---|---|---|
| 7:0 | ENC0L[7:0] | **ENC0 Data Low Byte.** |

## SFR Definition 15.3. ENC0M: ENC0 Data Middle Byte

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | ENC0M[7:0] | | | | | | | |
| Type | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Page = 0x2; SFR Address = 0xC3; Bit-Addressable

| Bit | Name | Function |
|---|---|---|
| 7:0 | ENC0M[7:0] | **ENC0 Data Middle Byte.** |

## SFR Definition 15.4. ENC0H: ENC0 Data High Byte

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | ENC0H[7:0] | | | | | | | |
| Type | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Page = 0x2; SFR Address = 0xC4; Bit-Addressable

| Bit | Name | Function |
|---|---|---|
| 7:0 | ENC0H[7:0] | **ENC0 Data High Byte.** |

**Table 16.3. Special Function Registers (Continued)**

SFRs are listed in alphabetical order. All undefined SFR locations are reserved

| Register | Address | SFR Page | Description | Page |
|---|---|---|---|---|
| TL0 | 0x8A | 0x0 | Timer/Counter 0 Low | 452 |
| TL1 | 0x8B | 0x0 | Timer/Counter 1 Low | 452 |
| TMOD | 0x89 | 0x0 | Timer/Counter Mode | 451 |
| TMR2CN | 0xC8 | All Pages | Timer/Counter 2 Control | 457 |
| TMR2H | 0xCD | 0x0 | Timer/Counter 2 High | 459 |
| TMR2L | 0xCC | 0x0 | Timer/Counter 2 Low | 459 |
| TMR2RLH | 0xCB | 0x0 | Timer/Counter 2 Reload High | 458 |
| TMR2RLL | 0xCA | 0x0 | Timer/Counter 2 Reload Low | 458 |
| TMR3CN | 0x91 | 0x0 | Timer/Counter 3 Control | 463 |
| TMR3H | 0x95 | 0x0 | Timer/Counter 3 High | 465 |
| TMR3L | 0x94 | 0x0 | Timer/Counter 3 Low | 465 |
| TMR3RLH | 0x93 | 0x0 | Timer/Counter 3 Reload High | 464 |
| TMR3RLL | 0x92 | 0x0 | Timer/Counter 3 Reload Low | 464 |
| TOFFH | 0xBB | 0xF | Temperature Offset High | 99 |
| TOFFL | 0xBD | 0xF | Temperature Offset Low | 99 |
| VDM0CN | 0xFF | All Pages | VDD Monitor Control | 282 |
| XBR0 | 0xE1 | 0x0 and 0xF | Port I/O Crossbar Control 0 | 358 |
| XBR1 | 0xE2 | 0x0 and 0xF | Port I/O Crossbar Control 1 | 359 |
| XBR2 | 0xE3 | 0x0 and 0xF | Port I/O Crossbar Control 2 | 360 |

## 22.2. Power-Fail Reset

C8051F96x devices have two Active Mode Supply Monitors that can hold the system in reset if the supply voltage drops below $V_{RST}$. The first of the two identical supply monitors is connected to the output of the supply select switch (which chooses the VBAT or VDC pin as the source of the digital supply voltage) and is enabled and selected as a reset source after each power-on or power-fail reset. This supply monitor will be referred to as the digital supply monitor. The second supply monitor is connected directly to the VBAT pin and is disabled after each power-on or power-fail reset. This supply monitor will be referred to as the analog supply monitor. The analog supply monitor should be enabled any time the supply select switch is set to the VDC pin to ensure that the VBAT supply does not drop below $V_{RST}$ .

When enabled and selected as a reset source, any power down transition or power irregularity that causes the monitored supply voltage to drop below $V_{RST}$ will cause the RST pin to be driven low and the CIP-51 will be held in a reset state (see Figure 22.2). When the supply voltage returns to a level above $V_{RST}$, the CIP-51 will be released from the reset state.

After a power-fail reset, the PORSF flag reads 1, the contents of RAM are invalid, and the digital supply monitor is enabled and selected as a reset source. The enable state of either supply monitor and its selection as a reset source is only altered by power-on and power-fail resets. For example, if the supply monitor is de-selected as a reset source and disabled by software, then a software reset is performed, the supply monitor will remain disabled and de-selected after the reset.

In battery-operated systems, the contents of RAM can be preserved near the end of the battery's usable life if the device is placed in Sleep Mode prior to a power-fail reset occurring. When the device is in Sleep Mode, the power-fail reset is automatically disabled, both active mode supply monitors are turned off, and the contents of RAM are preserved as long as the supply does not fall below $V_{POR}$. A large capacitor can be used to hold the power supply voltage above $V_{POR}$ while the user is replacing the battery. Upon waking from Sleep mode, the enable and reset source select state of the $V_{DD}$ supply monitor are restored to the value last set by the user.

To allow software early notification that a power failure is about to occur, the VDDOK bit is cleared when the supply falls below the $V_{WARN}$ threshold. The VDDOK bit can be configured to generate an interrupt. Each of the active mode supply montiors have their independent VDDOK and $V_{WARN}$ flags. See Section "17. Interrupt Handler" on page 232 for more details.

**Important Note:** To protect the integrity of Flash contents, **the active mode supply monitor(s) must be enabled and selected as a reset source if software contains routines which erase or write Flash memory.** If the digital supply monitor is not enabled, any erase or write performed on Flash memory will cause a Flash Error device reset.

## 24.2. SmaRTClock Clocking Sources

The SmaRTClock peripheral is clocked from its own timebase, independent of the system clock. The SmaRTClock timebase can be derived from an external CMOS clock, the internal LFO, or the SmaRTClock oscillator circuit, which has two modes of operation: Crystal Mode, and Self-Oscillate Mode. The oscillation frequency is 32.768 kHz in Crystal Mode and can be programmed in the range of 10 kHz to 40 kHz in Self-Oscillate Mode. The internal LFO frequency is 16.4 kHz ±20%. The frequency of the SmaRTClock oscillator can be measured with respect to another oscillator using an on-chip timer. See Section “32. Timers” on page 444 for more information on how this can be accomplished.

**Note:** The SmaRTClock timebase can be selected as the system clock and routed to a port pin. See Section “23. Clocking Sources” on page 286 for information on selecting the system clock source and Section “27. Port Input/Output” on page 351 for information on how to route the system clock to a port pin. The SmaRTClock timebase can also be routed to a port pin while the device is in its ultra low power sleep mode. See the PMU0MD register description for details.

### 24.2.1. Using the SmaRTClock Oscillator with a Crystal or External CMOS Clock

When using Crystal Mode, a 32.768 kHz crystal should be connected between XTAL3 and XTAL4. No other external components are required. The following steps show how to start the SmaRTClock crystal oscillator in software:

1. Configure the XTAL3 and XTAL4 pins for Analog I/O.
2. Set SmaRTClock to Crystal Mode (XMODE = 1).
3. Disable Automatic Gain Control (AGCEN) and enable Bias Doubling (BIASX2) for fast crystal startup.
4. Set the desired loading capacitance (RTC0XCF).
5. Enable power to the SmaRTClock oscillator circuit (RTC0EN = 1).
6. Wait 20 ms.
7. Poll the SmaRTClock Clock Valid Bit (CLKVLD) until the crystal oscillator stabilizes.
8. Poll the SmaRTClock Load Capacitance Ready Bit (LOADRDY) until the load capacitance reaches its programmed value.
9. Enable Automatic Gain Control (AGCEN) and disable Bias Doubling (BIASX2) for maximum power savings.
10. Enable the SmaRTClock missing clock detector.
11. Wait 2 ms.
12. Clear the PMU0CF wake-up source flags.

In Crystal Mode, the SmaRTClock oscillator may be driven by an external CMOS clock. The CMOS clock should be applied to XTAL3. XTAL34 should be left floating. In this mode, the external CMOS clock is ac coupled into the SmaRTClock and should have a minimum voltage swing of 400 mV. The CMOS clock signal voltage should not exceed VDD or drop below GND. Bias levels closer to VDD will result in lower I/O power consumption because the XTAL3 pin has a built-in weak pull-up. The SmaRTClock oscillator should be configured to its lowest bias setting with AGC disabled. The CLKVLD bit is indeterminate when using a CMOS clock, however, the OSCFAIL bit may be checked 2 ms after SmaRTClock oscillator is powered on to ensure that there is a valid clock on XTAL3. The CLKVLD bit is forced low when BIASX2 is disabled.

## Internal Register Definition 24.5. RTC0XCN: SmaRTClock Oscillator Control

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | AGCEN | XMODE | BIASX2 | CLKVLD | LFOEN | | | |
| Type | R/W | R/W | R/W | R | R/W | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SmaRTClock Address = 0x05

| Bit | Name | Function |
|---|---|---|
| 7 | AGCEN | **SmaRTClock Oscillator Automatic Gain Control (AGC) Enable.**<br>0: AGC disabled.<br>1: AGC enabled. |
| 6 | XMODE | **SmaRTClock Oscillator Mode.**<br>Selects Crystal or Self Oscillate Mode.<br>0: Self-Oscillate Mode selected.<br>1: Crystal Mode selected. |
| 5 | BIASX2 | **SmaRTClock Oscillator Bias Double Enable.**<br>**Enables/disables the Bias Double feature.**<br>0: Bias Double disabled.<br>1: Bias Double enabled. |
| 4 | CLKVLD | **SmaRTClock Oscillator Crystal Valid Indicator.**<br>Indicates if oscillation amplitude is sufficient for maintaining oscillation. This bit always reads 0 when BIASX2 is disabled.<br>0: Oscillation has not started or oscillation amplitude is too low to maintain oscillation.<br>1: Sufficient oscillation amplitude detected. |
| 3 | LFOEN | **Low Frequency Oscillator Enable and Select.**<br>Overrides XMODE and selects the internal low frequency oscillator (LFO) as the SmaRTClock oscillator source.<br>0: XMODE determines SmaRTClock oscillator source.<br>1: LFO enabled and selected as SmaRTClock oscillator source. |
| 2:0 | Unused | Read = 000b; Write = Don't Care. |

## Internal Register Definition 24.7. RTC0CF: SmaRTClock Configuration

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | | ALRM2 | ALRM1 | ALRM0 | AUTORST | RTC2EN | RTC1EN | RTC0EN |
| Type | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SmaRTClock Address = 0x07

| Bit | Name | Function |
|---|---|---|
| 7 | Reserved | Read = 0b; Must write 0b. |
| 6 | ALRM2 | **Event Flag for Alarm 2.**<br>This bit must be cleared by software. Writing a '1' to this bit has no effect.<br>0: An Alarm 2 event has not occured since the last time the flag was cleared.<br>1: An Alarm 2 event has occured. |
| 5 | ALRM1 | **Event Flag for Alarm 1.**<br>This bit must be cleared by software. Writing a '1' to this bit has no effect.<br>0: An Alarm 1 event has not occured since the last time the flag was cleared.<br>1: An Alarm 1 event has occured. |
| 4 | ALRM0 | **Event Flag for Alarm 0.**<br>This bit must be cleared by software. Writing a '1' to this bit has no effect.<br>0: An Alarm 0 event has not occured since the last time the flag was cleared.<br>1: An Alarm 0 event has occured. |
| 3 | AUTORST | **Auto Reset Enable.**<br>Enables the Auto Reset function to clear the counter when an Alarm 0 event occurs.<br>0: Auto Reset is disabled<br>1: Auto Reset is enabled. |
| 2 | RTC2EN | **Alarm 2 Enable.**<br>0: Alarm 2 is disabled.<br>1: Alarm 2 is enabled. |
| 1 | RTC1EN | **Alarm 1 Enable.**<br>0: Alarm 1 is disabled.<br>1: Alarm 1 is enabled. |
| 0 | RTC0EN | **Alarm 0 Enable.**<br>0: Alarm 0 is disabled.<br>1: Alarm 0 is enabled. |

## Internal Register Definition 24.8. CAPTUREn: SmaRTClock Timer Capture

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | CAPTURE[31:0] | | | | | | | |
| Type | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SmaRTClock Addresses: CAPTURE0 = 0x00; CAPTURE1 = 0x01; CAPTURE2 =0x02; CAPTURE3: 0x03.

| Bit | Name | Function |
|---|---|---|
| 7:0 | CAPTURE[31:0] | **SmaRTClock Timer Capture.**<br>These 4 registers (CAPTURE3–CAPTURE0) are used to read or set the 32-bit SmaRTClock timer. Data is transferred to or from the SmaRTClock timer when the RTC0SET or RTC0CAP bits are set. |
| **Note:** The least significant bit of the timer capture value is CAPTURE0.0. | | |

## Internal Register Definition 24.9. ALARM0Bn: SmaRTClock Alarm 0 Match Value

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | ALARM0[31:0] | | | | | | | |
| Type | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SmaRTClock Address: ALARM0B0 = 0x08; ALARM0B1 = 0x09; ALARM0B2 = 0x0A; ALARM0B3 = 0x0B

| Bit | Name | Function |
|---|---|---|
| 7:0 | ALARM0[31:0] | **SmaRTClock Alarm 0 Programmed Value.**<br>These 4 registers (ALARM0B3–ALARM0B0) are used to set an alarm event for the SmaRTClock timer. The SmaRTClock alarm should be disabled (ALRM0EN=0) when updating these registers. |
| **Note:** The least significant bit of the alarm programmed value is ALARM0B0.0. | | |

## 27.1. Port I/O Modes of Operation

Port pins P0.0–P6.7 use the Port I/O cell shown in Figure 27.2. The supply pin for P1.5–P2.3 is VIORF and the supply for all other GPIOs is VIO. Each Port I/O cell can be configured by software for analog I/O or digital I/O using the PnMDIN registers. P7.0 can only be used for digital functtons and is shared with the C2D signal. On reset, all Port I/O cells default to a digital high impedance state with weak pull-ups enabled.

### 27.1.1. Port Pins Configured for Analog I/O

Any pins to be used as Comparator or ADC input, external oscillator input/output, or AGND, VREF, or Current Reference output should be configured for analog I/O (PnMDIN.n = 0). When a pin is configured for analog I/O, its weak pullup and digital receiver are disabled. In most cases, software should also disable the digital output drivers. Port pins configured for analog I/O will always read back a value of 0 regardless of the actual voltage on the pin.

Configuring pins as analog I/O saves power and isolates the Port pin from digital interference. Port pins configured as digital inputs may still be used by analog peripherals; however, this practice is not recommended and may result in measurement errors.

### 27.1.2. Port Pins Configured For Digital I/O

Any pins to be used by digital peripherals (UART, SPI, SMBus, etc.), external digital event capture functions, or as GPIO should be configured as digital I/O (PnMDIN.n = 1). For digital I/O pins, one of two output modes (push-pull or open-drain) must be selected using the PnMDOUT registers.

Push-pull outputs (PnMDOUT.n = 1) drive the Port pad to the supply or GND rails based on the output logic value of the Port pin. Open-drain outputs have the high side driver disabled; therefore, they only drive the Port pad to GND when the output logic value is 0 and become high impedance inputs (both high and low drivers turned off) when the output logic value is 1.

When a digital I/O cell is placed in the high impedance state, a weak pull-up transistor pulls the Port pad to the supply voltage to ensure the digital input is at a defined logic state. Weak pull-ups are disabled when the I/O cell is driven to GND to minimize power consumption and may be globally disabled by setting WEAKPUD to 1. The user must ensure that digital I/O are always internally or externally pulled or driven to a valid logic state. Port pins configured for digital I/O always read back the logic state of the Port pad, regardless of the output logic value of the Port pin.

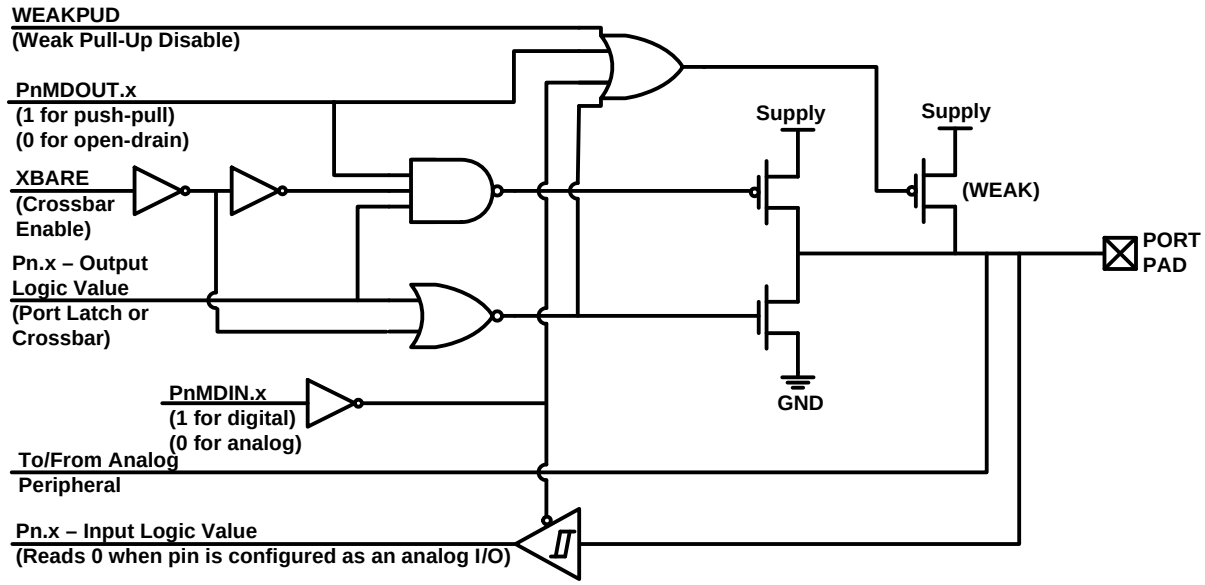


**Figure 27.2. Port I/O Cell Block Diagram**

## 31.3. SPI1 Slave Mode Operation

When SPI1 is enabled and not configured as a master, it will operate as a SPI slave. As a slave, bytes are shifted in through the MOSI pin and out through the MISO pin by a master device controlling the SCK signal. A bit counter in the SPI1 logic counts SCK edges. When 8 bits have been shifted through the shift register, the SPIF flag is set to logic 1, and the byte is copied into the receive buffer. Data is read from the receive buffer by reading SPI1DAT. A slave device cannot initiate transfers. Data to be transferred to the master device is pre-loaded into the shift register by writing to SPI1DAT. Writes to SPI1DAT are double-buffered, and are placed in the transmit buffer first. If the shift register is empty, the contents of the transmit buffer will immediately be transferred into the shift register. When the shift register already contains data, the SPI will load the shift register with the transmit buffer's contents after the last SCK edge of the next (or current) SPI transfer.

When configured as a slave, SPI1 can be configured for 4-wire or 3-wire operation. The default, 4-wire slave mode, is active when NSSMD1 (SPI1CN.3) = 0 and NSSMD0 (SPI1CN.2) = 1. In 4-wire mode, the NSS signal is routed to a port pin and configured as a digital input. SPI1 is enabled when NSS is logic 0, and disabled when NSS is logic 1. The bit counter is reset on a falling edge of NSS. Note that the NSS signal must be driven low at least 2 system clocks before the first active edge of SCK for each byte transfer. Figure 31.4 shows a connection diagram between two slave devices in 4-wire slave mode and a master device.

3-wire slave mode is active when NSSMD1 (SPI1CN.3) = 0 and NSSMD0 (SPI1CN.2) = 0. NSS is not used in this mode, and is not mapped to an external port pin through the crossbar. Since there is no way of uniquely addressing the device in 3-wire slave mode, SPI1 must be the only slave device present on the bus. It is important to note that in 3-wire slave mode there is no external means of resetting the bit counter that determines when a full byte has been received. The bit counter can only be reset by disabling and re-enabling SPI1 with the SPIEN bit. Figure 31.3 shows a connection diagram between a slave device in 3-wire slave mode and a master device.

## 31.4. SPI1 Interrupt Sources

When SPI1 interrupts are enabled, the following four flags will generate an interrupt when they are set to logic 1:

All of the following bits must be cleared by software.

- The SPI Interrupt Flag, SPIF (SPI1CN.7) is set to logic 1 at the end of each byte transfer. This flag can occur in all SPI1 modes.
- The Write Collision Flag, WCOL (SPI1CN.6) is set to logic 1 if a write to SPI1DAT is attempted when the transmit buffer has not been emptied to the SPI shift register. When this occurs, the write to SPI1DAT will be ignored, and the transmit buffer will not be written.This flag can occur in all SPI1 modes.
- The Mode Fault Flag MODF (SPI1CN.5) is set to logic 1 when SPI1 is configured as a master, and for multi-master mode and the NSS pin is pulled low. When a Mode Fault occurs, the MSTEN and SPIEN bits in SPI1CN are set to logic 0 to disable SPI1 and allow another master device to access the bus.
- The Receive Overrun Flag RXOVRN (SPI1CN.4) is set to logic 1 when configured as a slave, and a transfer is completed and the receive buffer still holds an unread byte from a previous transfer. The new byte is not transferred to the receive buffer, allowing the previously received data byte to be read. The data byte which caused the overrun is lost.

## 31.5. Serial Clock Phase and Polarity

Four combinations of serial clock phase and polarity can be selected using the clock control bits in the SPI1 Configuration Register (SPI1CFG). The CKPHA bit (SPI1CFG.5) selects one of two clock phases (edge used to latch the data). The CKPOL bit (SPI1CFG.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. SPI1 should be disabled (by clearing the SPIEN bit, SPI1CN.0) when changing the clock phase or polarity. The clock and data line relationships for master mode are shown in Figure 31.5. For slave mode, the clock and data relationships are shown in Figure 31.6 and Figure 31.7. Note that CKPHA should be set to 0 on both the master and slave SPI when communicating between two Silicon Labs C8051 devices.

The SPI1 Clock Rate Register (SPI1CKR) as shown in SFR Definition 31.3 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.

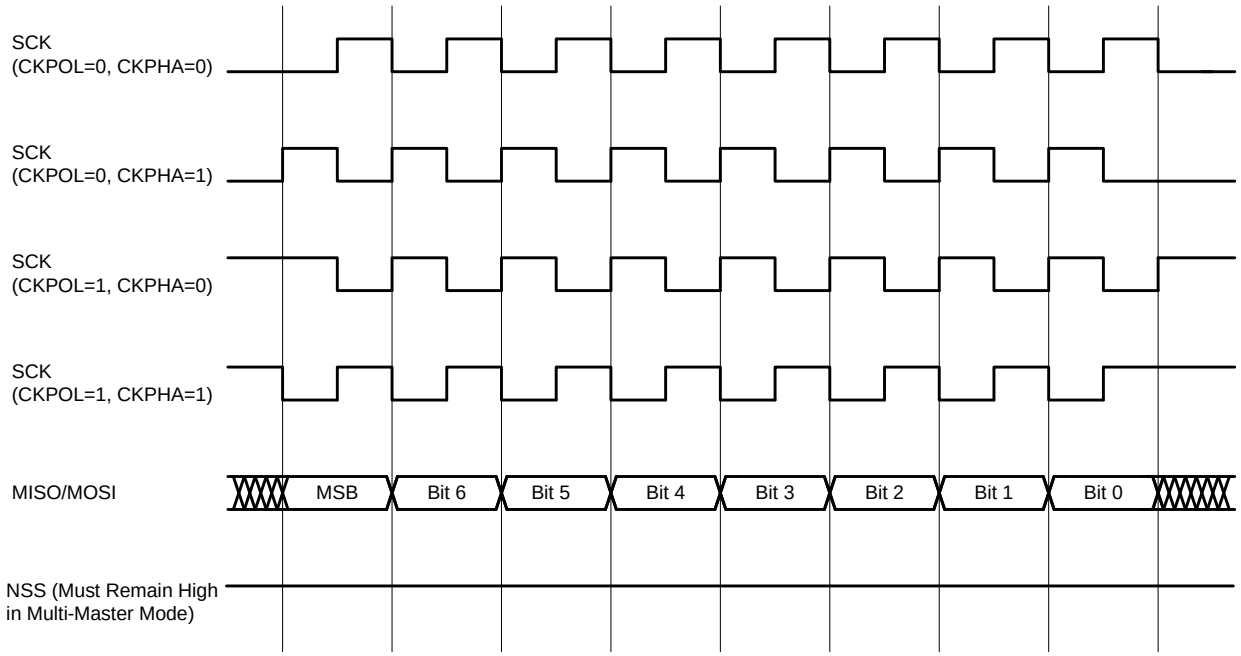


**Figure 31.5. Master Mode Data/Clock Timing**

## SFR Definition 31.1. SPI1CFG: SPI1 Configuration

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | SPIBSY | MSTEN | CKPHA | CKPOL | SLVSEL | NSSIN | SRMT | RXBMT |
| Type | R | R/W | R/W | R/W | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |

SFR Page = 0x0; SFR Address = 0x84

| Bit | Name | Function |
|---|---|---|
| 7 | SPIBSY | **SPI Busy.**<br>This bit is set to logic 1 when a SPI transfer is in progress (master or slave mode). |
| 6 | MSTEN | **Master Mode Enable.**<br>0: Disable master mode. Operate in slave mode.<br>1: Enable master mode. Operate as a master. |
| 5 | CKPHA | **SPI1 Clock Phase.**<br>0: Data centered on first edge of SCK period.*<br>1: Data centered on second edge of SCK period.* |
| 4 | CKPOL | **SPI1 Clock Polarity.**<br>0: SCK line low in idle state.<br>1: SCK line high in idle state. |
| 3 | SLVSEL | **Slave Selected Flag.**<br>This bit is set to logic 1 whenever the NSS pin is low indicating SPI1 is the selected slave. It is cleared to logic 0 when NSS is high (slave not selected). This bit does not indicate the instantaneous value at the NSS pin, but rather a de-glitched version of the pin input. |
| 2 | NSSIN | **NSS Instantaneous Pin Input.**<br>This bit mimics the instantaneous value that is present on the NSS port pin at the time that the register is read. This input is not de-glitched. |
| 1 | SRMT | **Shift Register Empty (valid in slave mode only).**<br>This bit will be set to logic 1 when all data has been transferred in/out of the shift register, and there is no new information available to read from the transmit buffer or write to the receive buffer. It returns to logic 0 when a data byte is transferred to the shift register from the transmit buffer or by a transition on SCK. SRMT = 1 when in Master Mode. |
| 0 | RXBMT | **Receive Buffer Empty (valid in slave mode only).**<br>This bit will be set to logic 1 when the receive buffer has been read and contains no new information. If there is new information available in the receive buffer that has not been read, this bit will return to logic 0. RXBMT = 1 when in Master Mode. |

**Note:** In slave mode, data on MOSI is sampled in the center of each data bit. In master mode, data on MISO is sampled one SYSCLK before the end of each data bit, to provide maximum settling time for the slave device. See Table 31.1 for timing parameters.

## 33.1. 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 33.1.

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. Clearing the CIDL bit in the PCA0MD register allows the PCA to continue normal operation while the CPU is in Idle mode.

**Table 33.1. 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[1] |
| 1 | 1 | 0 | SmaRTClock oscillator source divided by 8[2] |
| 1 | 1 | 1 | Reserved |

**Notes:**
1. External oscillator source divided by 8 is synchronized with the system clock.
2. SmaRTClock oscillator source divided by 8 is synchronized with the system clock.

## SFR Definition 33.3. PCA0PWM: PCA PWM Configuration

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | ARSEL | ECOV | COVF | | | | CLSEL[1:0] | |
| Type | R/W | R/W | R/W | R | R | R | R/W | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Page = 0x0; SFR Address = 0xDF

| Bit | Name | Function |
|---|---|---|
| 7 | ARSEL | **Auto-Reload Register Select.**<br>This bit selects whether to read and write the normal PCA capture/compare registers (PCA0CPn), or the Auto-Reload registers at the same SFR addresses. This function is used to define the reload value for 9, 10, and 11-bit PWM modes. In all other modes, the Auto-Reload registers have no function.<br>0: Read/Write Capture/Compare Registers at PCA0CPHn and PCA0CPLn.<br>1: Read/Write Auto-Reload Registers at PCA0CPHn and PCA0CPLn. |
| 6 | ECOV | **Cycle Overflow Interrupt Enable.**<br>This bit sets the masking of the Cycle Overflow Flag (COVF) interrupt.<br>0: COVF will not generate PCA interrupts.<br>1: A PCA interrupt will be generated when COVF is set. |
| 5 | COVF | **Cycle Overflow Flag.**<br>This bit indicates an overflow of the 8th, 9th, 10th, or 11th bit of the main PCA counter (PCA0). The specific bit used for this flag depends on the setting of the Cycle Length Select bits. The bit can be set by hardware or software, but must be cleared by software.<br>0: No overflow has occurred since the last time this bit was cleared.<br>1: An overflow has occurred since the last time this bit was cleared. |
| 4:2 | Unused | Read = 000b; Write = don't care. |
| 1:0 | CLSEL[1:0] | **Cycle Length Select.**<br>When 16-bit PWM mode is not selected, these bits select the length of the PWM cycle, between 8, 9, 10, or 11 bits. This affects all channels configured for PWM which are not using 16-bit PWM mode. These bits are ignored for individual channels configured to16-bit PWM mode.<br>00: 8 bits.<br>01: 9 bits.<br>10: 10 bits.<br>11: 11 bits. |