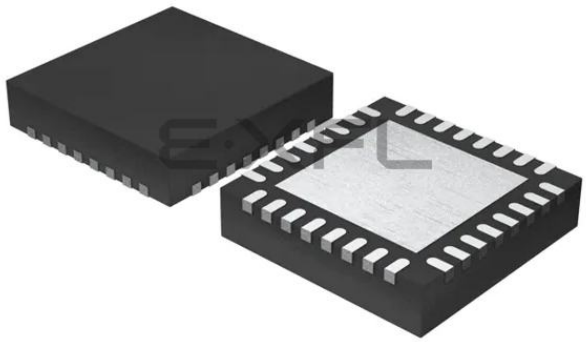

Welcome to E-XFL.COM

## What is "Embedded - Microcontrollers"?

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

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Active |
| Core Processor | 8051 |
| Core Size | 8-Bit |
| Speed | 48 MIPS |
| Connectivity | I²C, SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT |
| Number of I/O | 25 |
| 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 ~ 5.25V |
| Data Converters | A/D 21x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 32-VFQFN Exposed Pad |
| Supplier Device Package | 32-QFN (5x5) |
| Purchase URL | https://www.e-xfl.com/product-detail/silicon-labs/c8051f389-b-gmr |

## List of Tables

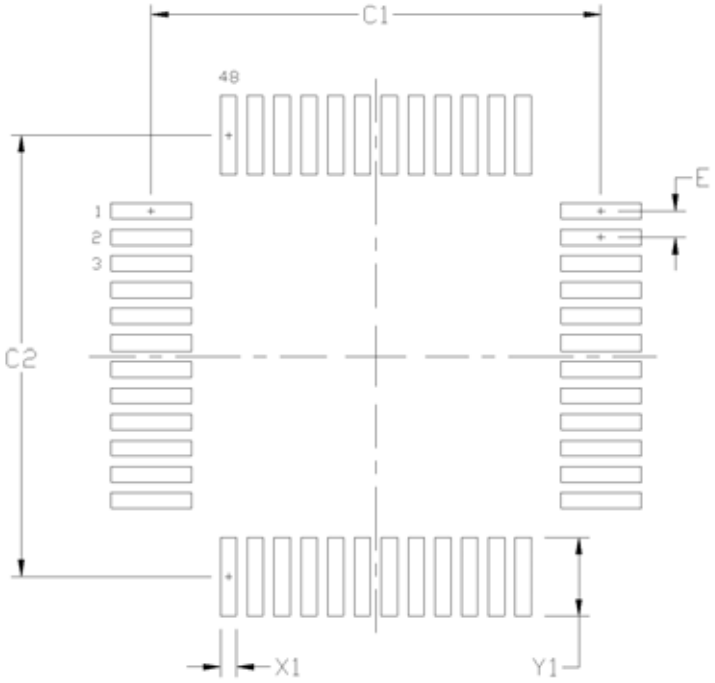


**Figure 3.3. TQFP-48 Recommended PCB Land Pattern**

**Table 3.3. TQFP-48 PCB Land Pattern Dimensions**

| Dimension | Min | Max |
|---|---|---|
| C1 | 8.30 | 8.40 |
| C2 | 8.30 | 8.40 |
| E | 0.50 BSC | |
| X1 | 0.20 | 0.30 |
| Y1 | 1.40 | 1.50 |

**Notes:**

**General:**

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. This Land Pattern Design is based on the IPC-7351 guidelines.

**Solder Mask Design:**

3. All metal pads are to be non-solder mask defined (NSMD). Clearance between the solder mask and the metal pad is to be 60 µm minimum, all the way around the pad.

**Stencil Design:**

4. A stainless steel, laser-cut and electro-polished stencil with trapezoidal walls should be used to assure good solder paste release.
5. The stencil thickness should be 0.125 mm (5 mils).
6. The ratio of stencil aperture to land pad size should be 1:1 for all pads.

**Card Assembly:**

7. A No-Clean, Type-3 solder paste is recommended.
8. The recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

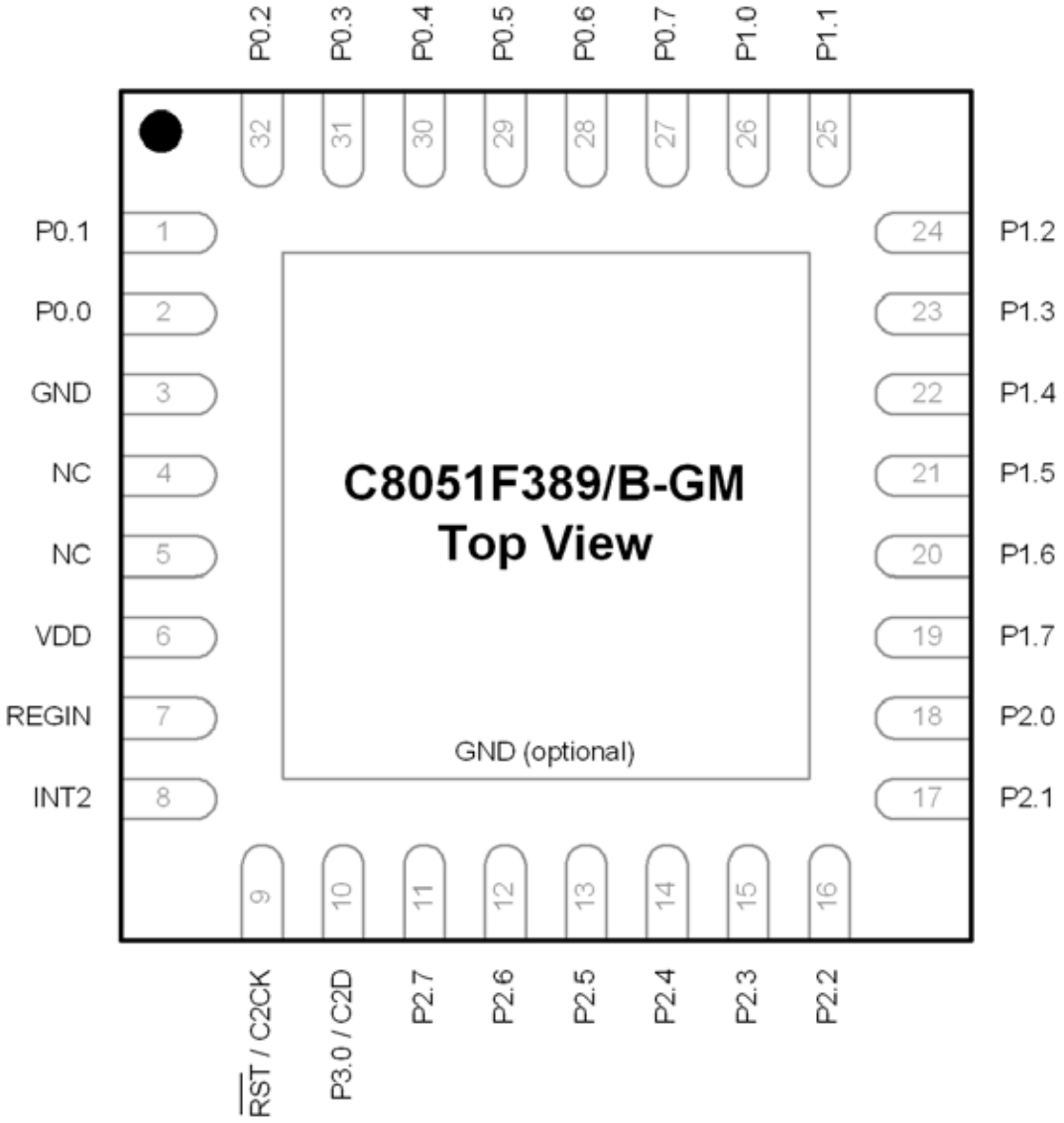


**Figure 3.7. QFN-32 (C8051F389/B) Pinout Diagram (Top View)**

### Table 5.11. Temperature Sensor Electrical Characteristics

$V_{DD}$ = 3.0 V, –40 to +85 °C unless otherwise specified.

| Parameter | Test Condition | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Linearity | | — | ± 0.5 | — | °C |
| Slope | | — | 2.87 | — | mV/°C |
| Slope Error* | | — | ±120 | — | µV/°C |
| Offset | Temp = 0 °C | — | 764 | — | mV |
| Offset Error* | Temp = 0 °C | — | ±15 | — | mV |
| **Note:** Represents one standard deviation from the mean. | | | | | |

### Table 5.12. Voltage Reference Electrical Characteristics

$V_{DD}$ = 3.0 V; –40 to +85 °C unless otherwise specified.

| Parameter | Test Condition | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| **Internal Reference (REFBE = 1)** | | | | | |
| Output Voltage | 25 °C ambient | 2.38 | 2.42 | 2.46 | V |
| VREF Short-Circuit Current | | — | — | 8 | mA |
| VREF Temperature Coefficient | | — | 35 | — | ppm/°C |
| Load Regulation | Load = 0 to 200 µA to GND | — | 1.5 | — | ppm/µA |
| VREF Turn-on Time 1 | 4.7 µF tantalum, 0.1 µF ceramic bypass | — | 3 | — | ms |
| VREF Turn-on Time 2 | 0.1 µF ceramic bypass | — | 100 | — | µs |
| Power Supply Rejection | | — | 140 | — | ppm/V |
| **External Reference (REFBE = 0)** | | | | | |
| Input Voltage Range | | 1 | — | $V_{DD}$ | V |
| Input Current | Sample Rate = 500 ksps; VREF = 3.0 V | — | 9 | — | µA |
| **Power Specifications** | | | | | |
| Supply Current | | — | 75 | — | µA |

## 6.3. Modes of Operation

ADC0 has a maximum conversion speed of 500 ksps. The ADC0 conversion clock is a divided version of the system clock, determined by the AD0SC bits in the ADC0CF register.

### 6.3.1. Starting a Conversion

A conversion can be initiated in one of several ways, depending on the programmed states of the ADC0 Start of Conversion Mode bits (AD0CM2–0) in register ADC0CN. Conversions may be initiated by one of the following:

1. Writing a 1 to the AD0BUSY bit of register ADC0CN
2. A Timer 0 overflow (i.e., timed continuous conversions)
3. A Timer 2 overflow
4. A Timer 1 overflow
5. A rising edge on the CNVSTR input signal
6. A Timer 3 overflow
7. A Timer 4 overflow
8. A Timer 5 overflow

Writing a 1 to AD0BUSY provides software control of ADC0 whereby conversions are performed "on-demand". During conversion, the AD0BUSY bit is set to logic 1 and reset to logic 0 when the conversion is complete. The falling edge of AD0BUSY triggers an interrupt (when enabled) and sets the ADC0 interrupt flag (AD0INT). Note: When polling for ADC conversion completions, the ADC0 interrupt flag (AD0INT) should be used. Converted data is available in the ADC0 data registers, ADC0H:ADC0L, when bit AD0INT is logic 1. Note that when Timer 2, 3, 4, or 5 overflows are used as the conversion source, Low Byte overflows are used if the timer is in 8-bit mode; High byte overflows are used if the timer is in 16-bit mode. See Section “25. Timers” on page 224 for timer configuration.

**Important Note About Using CNVSTR:** The CNVSTR input pin also functions as a Port I/O pin. When the CNVSTR input is used as the ADC0 conversion source, the associated pin should be skipped by the Digital Crossbar. See Section “20. Port Input/Output” on page 147 for details on Port I/O configuration.

## SFR Definition 6.7. ADC0LTH: ADC0 Less-Than Data High Byte

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | ADC0LTH[7:0] | | | | | | | |
| Type | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Address = 0xC6; SFR Page = All Pages

| Bit | Name | Function |
|---|---|---|
| 7:0 | ADC0LTH[7:0] | **ADC0 Less-Than Data Word High-Order Bits.** |

## SFR Definition 6.8. ADC0LTL: ADC0 Less-Than Data Low Byte

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | ADC0LTL[7:0] | | | | | | | |
| Type | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Address = 0xC5; SFR Page = All Pages

| Bit | Name | Function |
|---|---|---|
| 7:0 | ADC0LTL[7:0] | **ADC0 Less-Than Data Word Low-Order Bits.** |

## 8.1. Comparator Multiplexers

C8051F388/9/A/B devices include an analog input multiplexer to connect Port I/O pins to the comparator inputs. The Comparator inputs are selected in the CPTnMX registers (SFR Definition 8.5 and SFR Definition 8.6). The CMXnP2–CMXnP0 bits select the Comparator positive input; the CMXnN2–CMXnN0 bits select the Comparator negative input.

**Important Note About Comparator Inputs:** The Port pins selected as comparator inputs should be configured as analog inputs in their associated Port configuration register, and configured to be skipped by the Crossbar (for details on Port configuration, see Section “20.3. General Purpose Port I/O” on page 155).

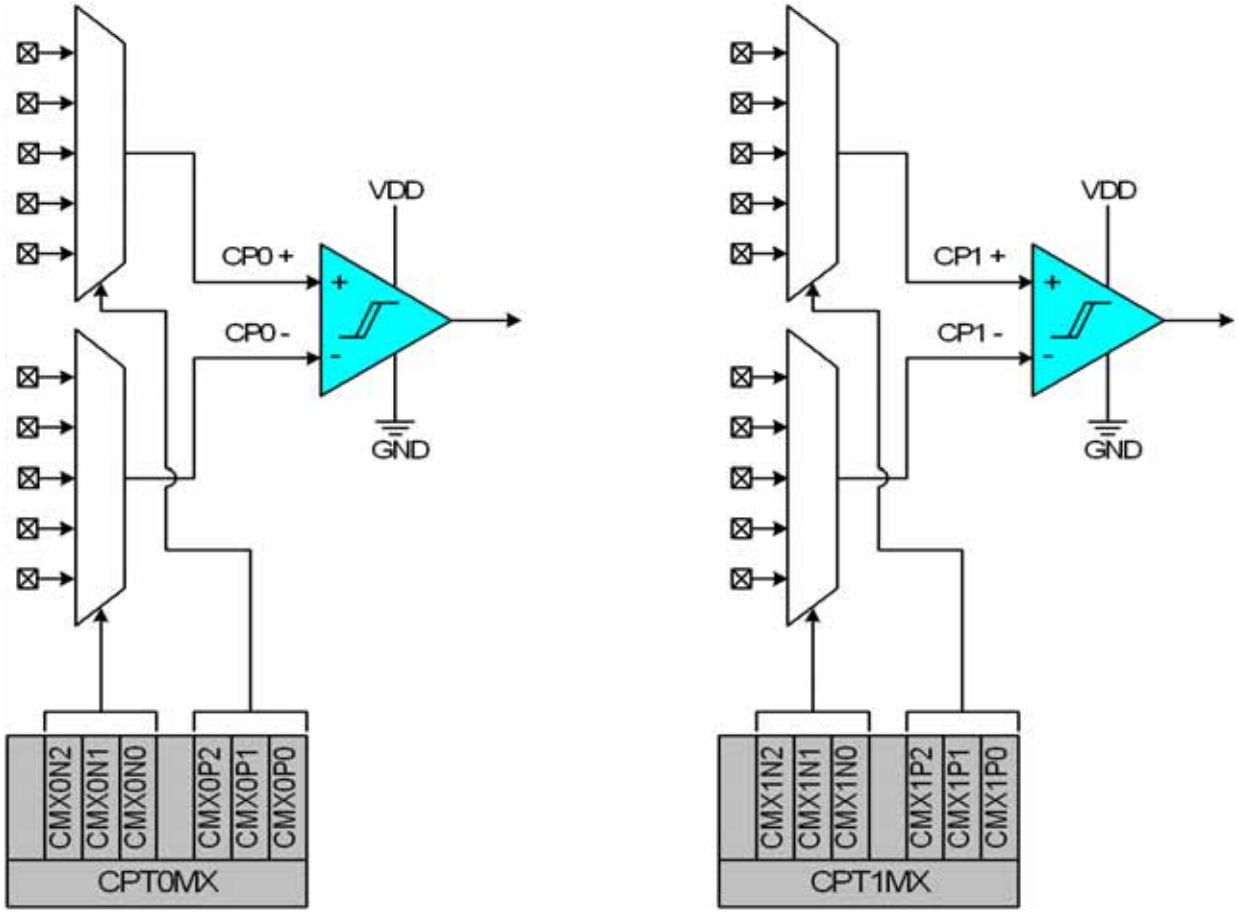


**Figure 8.4. Comparator Input Multiplexer Block Diagram**

Reset” on page 127 for more information on the use and configuration of the WDT.

## 10.2. Stop Mode

Setting the stop mode Select bit (PCON.1) causes the controller core to enter stop mode as soon as the instruction that sets the bit completes execution. In stop mode the internal oscillator, CPU, and all digital peripherals are stopped; the state of the external oscillator circuit is not affected. Each analog peripheral (including the external oscillator circuit) may be shut down individually prior to entering stop mode. Stop mode can only be terminated by an internal or external reset. On reset, the device performs the normal reset sequence and begins program execution at address 0x0000.

If enabled, the Missing Clock Detector will cause an internal reset and thereby terminate the stop mode. The Missing Clock Detector should be disabled if the CPU is to be put to in STOP mode for longer than the MCD timeout.

By default, when in stop mode the internal regulator is still active. However, the regulator can be configured to shut down while in stop mode to save power. To shut down the regulator in stop mode, the STOPCF bit in register REG01CN should be set to 1 prior to setting the STOP bit (see SFR Definition 9.1). If the regulator is shut down using the STOPCF bit, only the $\overline{\text{RST}}$ pin or a full power cycle are capable of resetting the device.

## 10.3. Suspend Mode

Setting the SUSPEND bit (OSCICN.5) causes the hardware to halt the high-frequency internal oscillator and go into suspend mode as soon as the instruction that sets the bit completes execution. All internal registers and memory maintain their original data. The CPU is not halted in Suspend, so code can still be executed using an oscillator other than the internal high-frequency oscillator.

Suspend mode can be terminated by a rising or falling edge on the INT2 pin or a device reset event. When suspend mode is terminated, if the oscillator source is the internal high-frequency oscillator, the device will continue execution on the instruction following the one that set the SUSPEND bit. If the wake event was configured to generate an interrupt, the interrupt will be serviced upon waking the device. If suspend mode is terminated by an internal or external reset, the CIP-51 performs a normal reset sequence and begins program execution at address 0x0000.

### 14.6.1.3. 8-bit MOVX with Bank Select: EMI0CF[4:2] = 110

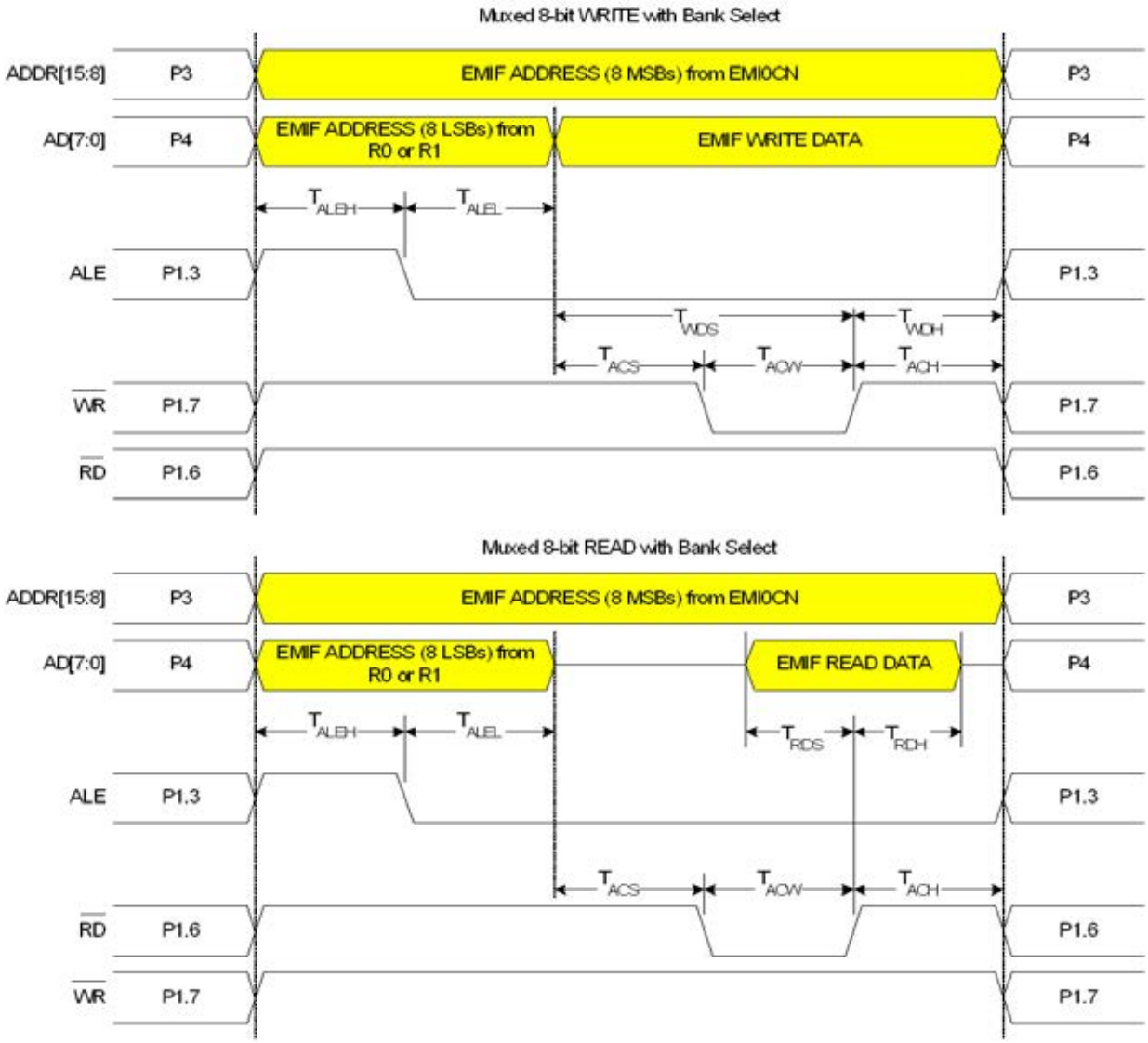


**Figure 14.6. Non-Multiplexed 8-bit MOVX with Bank Select Timing**

## SFR Definition 16.6. EIP2: Extended Interrupt Priority 2

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | | | PT5 | PT4 | PSMB1 | | PS1 | PINT2 |
| Type | R | R | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Address = 0xF7; SFR Page = All Pages

| Bit | Name | Function |
|---|---|---|
| :6 | Unused | Read = 00b, Write = Don't Care. |
| 5 | PT5 | **Timer 5 Interrupt Priority Control.**<br>This bit sets the priority of the Timer 5 interrupt.<br>0: Timer 5 interrupt set to low priority level.<br>1: Timer 5 interrupt set to high priority level. |
| 4 | PT4 | **Timer 4 Interrupt Priority Control.**<br>This bit sets the priority of the Timer 4 interrupt.<br>0: Timer 4 interrupt set to low priority level.<br>1: Timer 4 interrupt set to high priority level. |
| 3 | PSMB1 | **SMBus1 Interrupt Priority Control.**<br>This bit sets the priority of the SMB1 interrupt.<br>0: SMB1 interrupt set to low priority level.<br>1: SMB1 interrupt set to high priority level. |
| 2 | Reserved | Must Write 0b. |
| 1 | PS1 | **UART1 Interrupt Priority Control.**<br>This bit sets the priority of the UART1 interrupt.<br>0: UART1 interrupt set to low priority level.<br>1: UART1 interrupt set to high priority level. |
| 0 | PINT2 | **INT2 Level Interrupt Priority Control.**<br>This bit sets the masking of the INT2 interrupt.<br>0: INT2 interrupt set to low priority level.<br>1: INT2 interrupt set to high priority level. |

## SFR Definition 18.3. FLSCL: Flash Scale

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | FOSE | Reserved | | FLRT | Reserved | | | |
| Type | R/W | R/W | | R/W | R/W | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Address = 0xB6; SFR Page = All Pages

| Bit | Name | Function |
|---|---|---|
| 7 | FOSE | **Flash One-shot Enable.**<br>This bit enables the Flash read one-shot. When the Flash one-shot disabled, the Flash sense amps are enabled for a full clock cycle during Flash reads. At system clock frequencies below 10 MHz, disabling the Flash one-shot will increase system power consumption.<br>0: Flash one-shot disabled.<br>1: Flash one-shot enabled. |
| 6:5 | Reserved | Must write 00b. |
| 4 | FLRT | **FLASH Read Time.**<br>This bit should be programmed to the smallest allowed value, according to the system clock speed.<br>0: SYSCLK <= 25 MHz.<br>1: SYSCLK <= 48 MHz. |
| 3:0 | Reserved | Must write 0000b. |

**Table 21.3. Sources for Hardware Changes to SMBnCN**

| Bit | Set by Hardware When: | Cleared by Hardware When: |
|---|---|---|
| MASTERn | ■ A START is generated. | ■ A STOP is generated.<br>■ Arbitration is lost. |
| TXMODEn | ■ START is generated.<br>■ SMBnDAT is written before the start of an SMBus frame. | ■ A START is detected.<br>■ Arbitration is lost.<br>■ SMBnDAT is not written before the start of an SMBus frame. |
| STAn | ■ A START followed by an address byte is received. | ■ Must be cleared by software. |
| STOn | ■ A STOP is detected while addressed as a slave.<br>■ Arbitration is lost due to a detected STOP. | ■ A pending STOP is generated. |
| ACKRQn | ■ A byte has been received and an ACK response value is needed (only when hardware ACK is not enabled). | ■ After each ACK cycle. |
| ARBLOSTn | ■ A repeated START is detected as a MASTER when STAn is low (unwanted repeated START).<br>■ SCLn is sensed low while attempting to generate a STOP or repeated START condition.<br>■ SDAn is sensed low while transmitting a 1 (excluding ACK bits). | ■ Each time SIn is cleared. |
| ACKn | ■ The incoming ACK value is low (ACKNOWLEDGE). | ■ The incoming ACK value is high (NOT ACKNOWLEDGE). |
| SIn | ■ A START has been generated.<br>■ Lost arbitration.<br>■ A byte has been transmitted and an ACK/NACK received.<br>■ A byte has been received.<br>■ A START or repeated START followed by a slave address + R/W has been received.<br>■ A STOP has been received. | ■ Must be cleared by software. |

### 21.4.4. Hardware Slave Address Recognition

The SMBus hardware has the capability to automatically recognize incoming slave addresses and send an ACK without software intervention. Automatic slave address recognition is enabled by setting the EHACK bit in register SMB0ADM to 1. This will enable both automatic slave address recognition and automatic hardware ACK generation for received bytes (as a master or slave). More detail on automatic hardware ACK generation can be found in Section 21.4.3.2.

The registers used to define which address(es) are recognized by the hardware are the SMBus Slave Address register and the SMBus Slave Address Mask register. A single address or range of addresses (including the General Call Address 0x00) can be specified using these two registers. The most-significant seven bits of the two registers are used to define which addresses will be ACKed. A 1 in bit positions of the slave address mask SLVM[6:0] enable a comparison between the received slave address and the hardware's slave address SLV[6:0] for those bits. A 0 in a bit of the slave address mask means that bit will be treated as a "don't care" for comparison purposes. In this case, either a 1 or a 0 value are acceptable on

### 21.5.2. Read Sequence (Master)

During a read sequence, an SMBus master reads data from a slave device. The master in this transfer will be a transmitter during the address byte, and a receiver during all data bytes. The SMBus interface generates the START condition and transmits the first byte containing the address of the target slave and the data direction bit. In this case the data direction bit (R/W) will be logic 1 (READ). Serial data is then received from the slave on SDA while the SMBus outputs the serial clock. The slave transmits one or more bytes of serial data.

If hardware ACK generation is disabled, the ACKRQ is set to 1 and an interrupt is generated after each received byte. Software must write the ACK bit at that time to ACK or NACK the received byte.

With hardware ACK generation enabled, the SMBus hardware will automatically generate the ACK/NACK, and then post the interrupt. **It is important to note that the appropriate ACK or NACK value should be set up by the software prior to receiving the byte when hardware ACK generation is enabled.**

Writing a 1 to the ACK bit generates an ACK; writing a 0 generates a NACK. Software should write a 0 to the ACK bit for the last data transfer, to transmit a NACK. The interface exits Master Receiver Mode after the STO bit is set and a STOP is generated. The interface will switch to Master Transmitter Mode if SMB0DAT is written while an active Master Receiver. Figure 21.6 shows a typical master read sequence. Two received data bytes are shown, though any number of bytes may be received. Notice that the ‘data byte transferred’ interrupts occur at different places in the sequence, depending on whether hardware ACK generation is enabled. The interrupt occurs **before** the ACK with hardware ACK generation disabled, and **after** the ACK when hardware ACK generation is enabled.

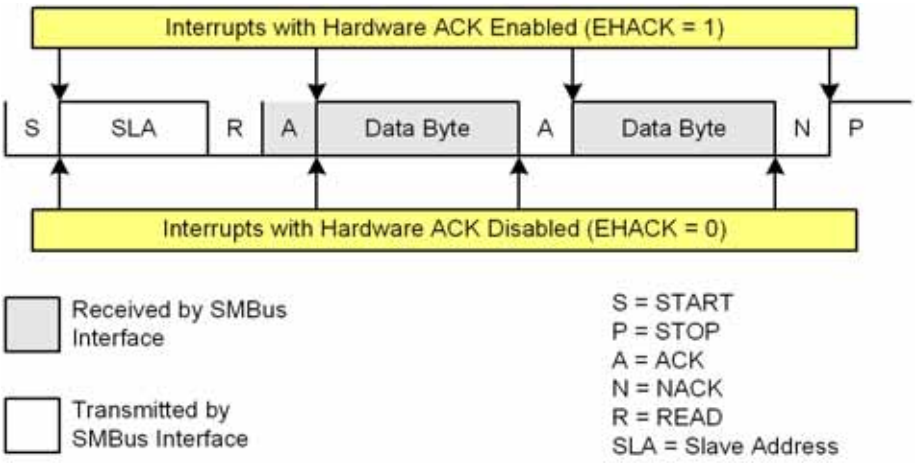


**Figure 21.6. Typical Master Read Sequence**

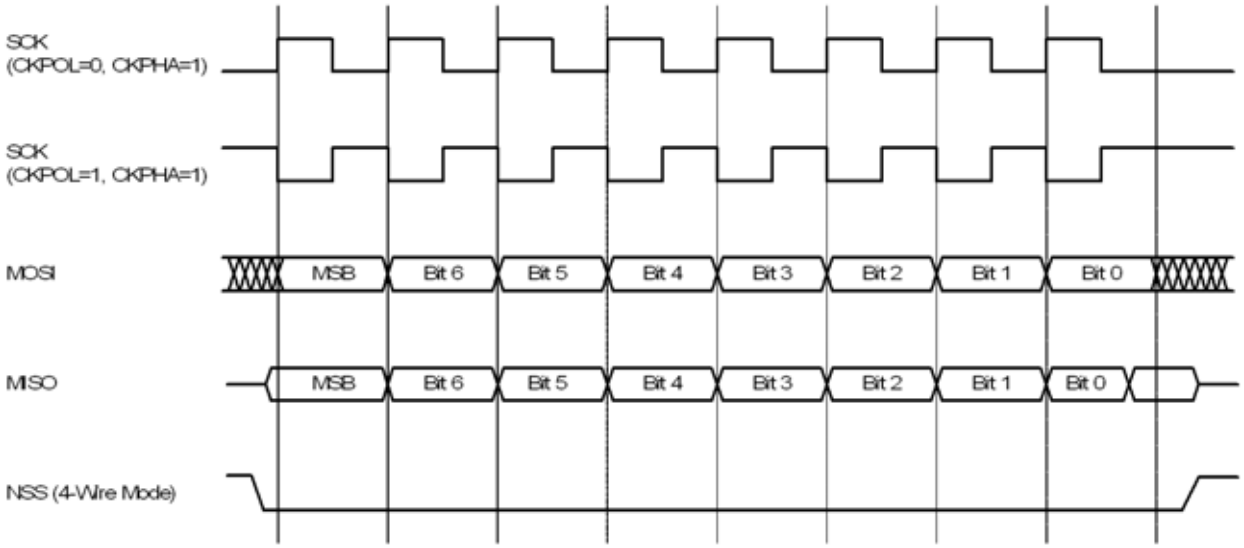


**Figure 24.7. Slave Mode Data/Clock Timing (CKPHA = 1)**

## 24.6. SPI Special Function Registers

SPI0 is accessed and controlled through four special function registers in the system controller: SPI0CN Control Register, SPI0DAT Data Register, SPI0CFG Configuration Register, and SPI0CKR Clock Rate Register. The four special function registers related to the operation of the SPI0 Bus are described in the following figures.

## 25.1. Timer 0 and Timer 1

Each timer is implemented as a 16-bit register accessed as two separate bytes: a low byte (TL0 or TL1) and a high byte (TH0 or TH1). The Counter/Timer Control register (TCON) is used to enable Timer 0 and Timer 1 as well as indicate status. Timer 0 interrupts can be enabled by setting the ET0 bit in the IE register; Timer 1 interrupts can be enabled by setting the ET1 bit in the IE register. Both counter/timers operate in one of four primary modes selected by setting the Mode Select bits T1M1–T0M0 in the Counter/Timer Mode register (TMOD). Each timer can be configured independently. Each operating mode is described below.

### 25.1.1. Mode 0: 13-bit Counter/Timer

Timer 0 and Timer 1 operate as 13-bit counter/timers in Mode 0. The following describes the configuration and operation of Timer 0. However, both timers operate identically, and Timer 1 is configured in the same manner as described for Timer 0.

The TH0 register holds the eight MSBs of the 13-bit counter/timer. TL0 holds the five LSBs in bit positions TL0.4–TL0.0. The three upper bits of TL0 (TL0.7–TL0.5) are indeterminate and should be masked out or ignored when reading. As the 13-bit timer register increments and overflows from 0x1FFF (all ones) to 0x0000, the timer overflow flag TF0 in TCON is set and an interrupt will occur if Timer 0 interrupts are enabled.

The C/T0 bit in the TMOD register selects the counter/timer's clock source. When C/T0 is set to logic 1, high-to-low transitions at the selected Timer 0 input pin (T0) increment the timer register (Refer to Section “20.1. Priority Crossbar Decoder” on page 148 for information on selecting and configuring external I/O pins). Clearing C/T selects the clock defined by the T0M bit in register CKCON. When T0M is set, Timer 0 is clocked by the system clock. When T0M is cleared, Timer 0 is clocked by the source selected by the Clock Scale bits in CKCON (see SFR Definition 25.1).

Setting the TR0 bit (TCON.4) enables the timer when either GATE0 in the TMOD register is logic 0 or the input signal $\overline{INT0}$ is active as defined by bit IN0PL in register IT01CF. Setting GATE0 to 1 allows the timer to be controlled by the external input signal $\overline{INT0}$, facilitating pulse width measurements

| TR0 | GATE0 | $\overline{INT0}$ | Counter/Timer |
|---|---|---|---|
| 0 | X | X | Disabled |
| 1 | 0 | X | Enabled |
| 1 | 1 | 0 | Disabled |
| 1 | 1 | 1 | Enabled |
| **Note:** X = Don't Care | | | |

Setting TR0 does not force the timer to reset. The timer registers should be loaded with the desired initial value before the timer is enabled.

TL1 and TH1 form the 13-bit register for Timer 1 in the same manner as described above for TL0 and TH0. Timer 1 is configured and controlled using the relevant TCON and TMOD bits just as with Timer 0. The input signal $\overline{INT1}$ is used with Timer 1; the $\overline{INT1}$ polarity is defined by bit IN1PL in register IT01CF.

## SFR Definition 25.3. TCON: Timer Control

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |
| 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 Address = 0x88; SFR Page = All Pages; Bit-Addressable

| Bit | Name | Function |
|---|---|---|
| 7 | TF1 | **Timer 1 Overflow Flag.**<br>Set to 1 by hardware when Timer 1 overflows. This flag can be cleared by software but is automatically cleared when the CPU vectors to the Timer 1 interrupt service routine. |
| 6 | TR1 | **Timer 1 Run Control.**<br>Timer 1 is enabled by setting this bit to 1. |
| 5 | TF0 | **Timer 0 Overflow Flag.**<br>Set to 1 by hardware when Timer 0 overflows. This flag can be cleared by software but is automatically cleared when the CPU vectors to the Timer 0 interrupt service routine. |
| 4 | TR0 | **Timer 0 Run Control.**<br>Timer 0 is enabled by setting this bit to 1. |
| 3 | IE1 | **External Interrupt 1.**<br>This flag is set by hardware when an edge/level of type defined by IT1 is detected. It can be cleared by software but is automatically cleared when the CPU vectors to the External Interrupt 1 service routine in edge-triggered mode. |
| 2 | IT1 | **Interrupt 1 Type Select.**<br>This bit selects whether the configured $\overline{\text{INT1}}$ interrupt will be edge or level sensitive. $\overline{\text{INT1}}$ is configured active low or high by the IN1PL bit in the IT01CF register (see SFR Definition 16.7).<br>0: $\overline{\text{INT1}}$ is level triggered.<br>1: $\overline{\text{INT1}}$ is edge triggered. |
| 1 | IE0 | **External Interrupt 0.**<br>This flag is set by hardware when an edge/level of type defined by IT1 is detected. It can be cleared by software but is automatically cleared when the CPU vectors to the External Interrupt 0 service routine in edge-triggered mode. |
| 0 | IT0 | **Interrupt 0 Type Select.**<br>This bit selects whether the configured $\overline{\text{INT0}}$ interrupt will be edge or level sensitive. $\overline{\text{INT0}}$ is configured active low or high by the IN0PL bit in register IT01CF (see SFR Definition 16.7).<br>0: $\overline{\text{INT0}}$ is level triggered.<br>1: $\overline{\text{INT0}}$ is edge triggered. |

### 25.2.2. 8-bit Timers with Auto-Reload

When T2SPLIT is set, Timer 2 operates as two 8-bit timers (TMR2H and TMR2L). Both 8-bit timers operate in auto-reload mode as shown in Figure 25.5. TMR2RLL holds the reload value for TMR2L; TMR2RLH holds the reload value for TMR2H. The TR2 bit in TMR2CN handles the run control for TMR2H. TMR2L is always running when configured for 8-bit Mode.

Each 8-bit timer may be configured to use SYSCLK, SYSCLK divided by 12, or the external oscillator clock source divided by 8. The Timer 2 Clock Select bits (T2MH and T2ML in CKCON) select either SYSCLK or the clock defined by the Timer 2 External Clock Select bit (T2XCLK in TMR2CN), as follows:

| T2MH | T2XCLK | TMR2H Clock Source |
|---|---|---|
| 0 | 0 | SYSCLK / 12 |
| 0 | 1 | External Clock / 8 |
| 1 | X | SYSCLK |

| T2ML | T2XCLK | TMR2L Clock Source |
|---|---|---|
| 0 | 0 | SYSCLK / 12 |
| 0 | 1 | External Clock / 8 |
| 1 | X | SYSCLK |

The TF2H bit is set when TMR2H overflows from 0xFF to 0x00; the TF2L bit is set when TMR2L overflows from 0xFF to 0x00. When Timer 2 interrupts are enabled (IE.5), an interrupt is generated each time TMR2H overflows. If Timer 2 interrupts are enabled and TF2LEN (TMR2CN.5) is set, an interrupt is generated each time either TMR2L or TMR2H overflows. When TF2LEN is enabled, software must check the TF2H and TF2L flags to determine the source of the Timer 2 interrupt. The TF2H and TF2L interrupt flags are not cleared by hardware and must be manually cleared by software.

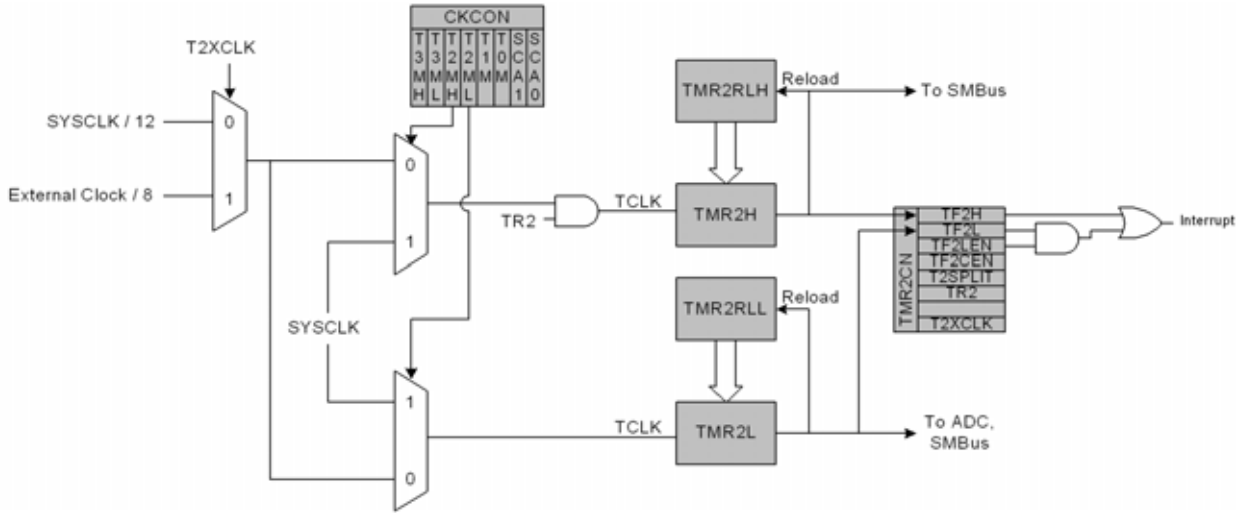


**Figure 25.5. Timer 2 8-Bit Mode Block Diagram**

### 25.2.3. Timer 2 Capture Modes: LFO Falling Edge

When T2CE = 1, Timer 2 will operate in a special capture mode with the LFO (T2CSS is set to 1). The LFO falling-edge capture mode can be used to calibrate the internal Low-Frequency Oscillator against the internal High-Frequency Oscillator or an external clock source. When T2SPLIT = 0, Timer 2 counts up and overflows from 0xFFFF to 0x0000. Each time a capture event is received, the contents of the Timer 2 registers (TMR2H:TMR2L) are latched into the Timer 2 Reload registers (TMR2RLH:TMR2RLL). A Timer 2 interrupt is generated if enabled.

## 26.2. PCA0 Interrupt Sources

Figure 26.3 shows a diagram of the PCA interrupt tree. There are six independent event flags that can be used to generate a PCA0 interrupt. They are: the main PCA counter overflow flag (CF), which is set upon a 16-bit overflow of the PCA0 counter and the individual flags for each PCA channel (CCF0, CCF1, CCF2, CCF3, and CCF4), which are set according to the operation mode of that module. These event flags are always set when the trigger condition occurs. Each of these flags can be individually selected to generate a PCA0 interrupt, using the corresponding interrupt enable flag (ECF for CF, and ECCFn for each CCFn). PCA0 interrupts must be globally enabled before any individual interrupt sources are recognized by the processor. PCA0 interrupts are globally enabled by setting the EA bit and the EPCA0 bit to logic 1.

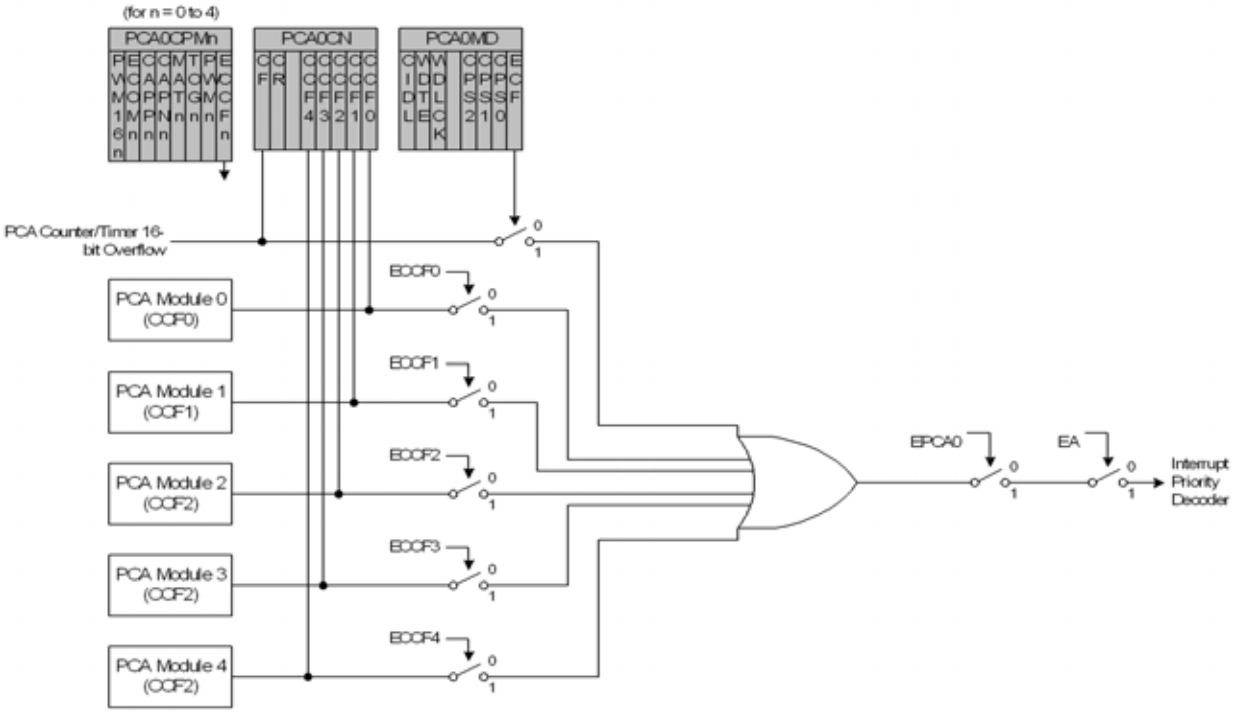


**Figure 26.3. PCA Interrupt Block Diagram**

## C2 Register Definition 27.2. DEVICEID: C2 Device ID

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

C2 Address: 0x00

| Bit | Name | Function |
|---|---|---|
| 7:0 | DEVICEID[7:0] | **Device ID.**<br>This read-only register returns the 8-bit device ID: 0x28 (C8051F388/9/A/B). |

## C2 Register Definition 27.3. REVID: C2 Revision ID

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

C2 Address: 0x01

| Bit | Name | Function |
|---|---|---|
| 7:0 | REVID[7:0] | **Revision ID.**<br>This read-only register returns the 8-bit revision ID. For example: 0x00 = Revision A or Revision B. |