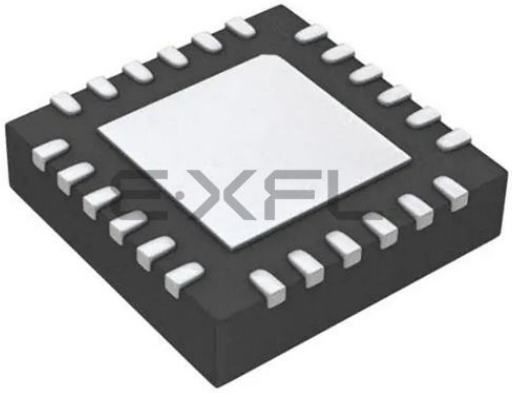

**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

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

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Not For New Designs |
| Core Processor | 8051 |
| Core Size | 8-Bit |
| Speed | 25MHz |
| Connectivity | SMBus (2-Wire/I²C), SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT |
| Number of I/O | 16 |
| Program Memory Size | 8KB (8K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 768 x 8 |
| Voltage - Supply (Vcc/Vdd) | 0.9V ~ 3.6V |
| Data Converters | A/D 15x10/12b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 24-WFQFN Exposed Pad |
| Supplier Device Package | 24-QFN (4x4) |
| Purchase URL | https://www.e-xfl.com/product-detail/silicon-labs/c8051f902-d-gm |

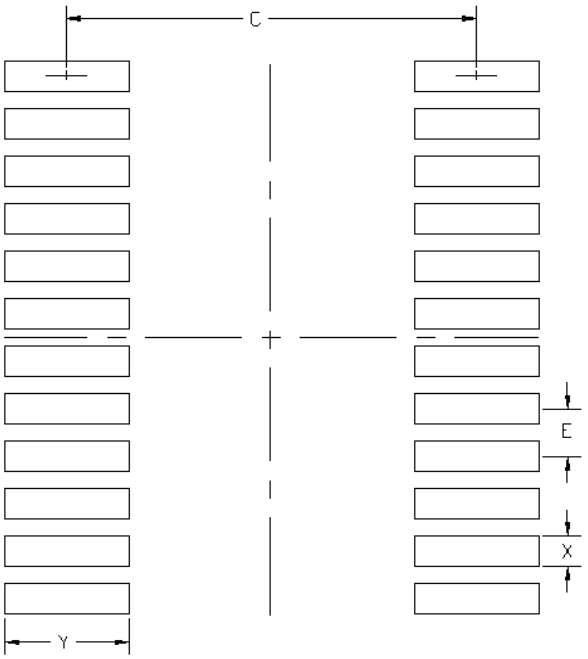


**Figure 3.6. QSOP-24 Landing Diagram**

**Table 3.5. PCB Land Pattern**

| Dimension | MIN | MAX |
|---|---|---|
| C | 5.20 | 5.30 |
| E | 0.635 BSC | |
| X | 0.30 | 0.40 |
| Y | 1.50 | 1.60 |

**Notes:**

**General**

**1.** All dimensions shown are in millimeters (mm) unless otherwise noted.

**2.** This land pattern is based on the IPC-7351 guidelines.

**Solder Mask Design**

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

**Stencil Design**

**1.** A stainless steel, laser-cut and electro-polished stencil with trapezoidal walls should be used to assure good solder paste release.

**2.** The stencil thickness should be 0.125 mm (5 mils).

**3.** The ratio of stencil aperture to land pad size should be 1:1 for all perimeter pads.

**Card Assembly**

**1.** A No-Clean, Type 3 solder paste is recommended.

**2.** The recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

## Table 4.2. Global Electrical Characteristics (Continued)

–40 to +85 °C, 25 MHz system clock unless otherwise specified. See "AN358: Optimizing Low Power Operation of the ‘F9xx" for details on how to achieve the supply current specifications listed in this table.

| Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|

**Notes:**

1. Based on device characterization data; Not production tested.
2. SYSCLK must be at least 32 kHz to enable debugging.
3. Digital Supply Current depends upon the particular code being executed. The values in this table are obtained with the CPU executing an “sjmp $” loop, which is the compiled form of a while(1) loop in C. One iteration requires 3 CPU clock cycles, and the Flash memory is read on each cycle. The supply current will vary slightly based on the physical location of the sjmp instruction and the number of Flash address lines that toggle as a result. In the worst case, current can increase by up to 30% if the sjmp loop straddles a 64-byte Flash address boundary (e.g., 0x007F to 0x0080). Real-world code with larger loops and longer linear sequences will have few transitions across the 64-byte address boundaries.
4. Includes oscillator and regulator supply current.
5. IDD can be estimated for frequencies $\leq$14 MHz by simply multiplying the frequency of interest by the frequency sensitivity number for that range, then adding an offset of 90 µA. When using these numbers to estimate $I_{DD}$ for >14 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, $I_{DD}$ = 4 mA – (25 MHz – 20 MHz) x 0.102 mA/MHz = 3.5 mA assuming the same oscillator setting.
6. The supply current specifications in Table 4.2 are for two cell mode. The VBAT current in one-cell mode can be estimated using the following equation:

$$\text{VBAT Current (one-cell mode)} = \frac{\text{Supply Voltage} \times \text{Supply Current (two-cell mode)}}{\text{DC-DC Converter Efficiency} \times \text{VBAT Voltage}}$$

   The VBAT Voltage is the voltage at the VBAT pin, typically 0.9 to 1.8 V.
   The Supply Current (two-cell mode) is the data sheet specification for supply current.
   The Supply Voltage is the voltage at the VDD/DC+ pin, typically 1.8 to 3.3 V (default = 1.9 V).
   The DC-DC Converter Efficiency can be estimated using Figure 4.3–Figure 4.5.
7. Idle IDD can be estimated by taking 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 $I_{DD}$ = 2.1 mA – (25 MHz – 5 MHz) x 0.079 mA/MHz = 0.52 mA.
8. Internal LFO only available on ‘F912 and ‘F902 devices.
9. Ability to disable VBAT supply monitor only available on ‘F912 and ‘F902 devices.

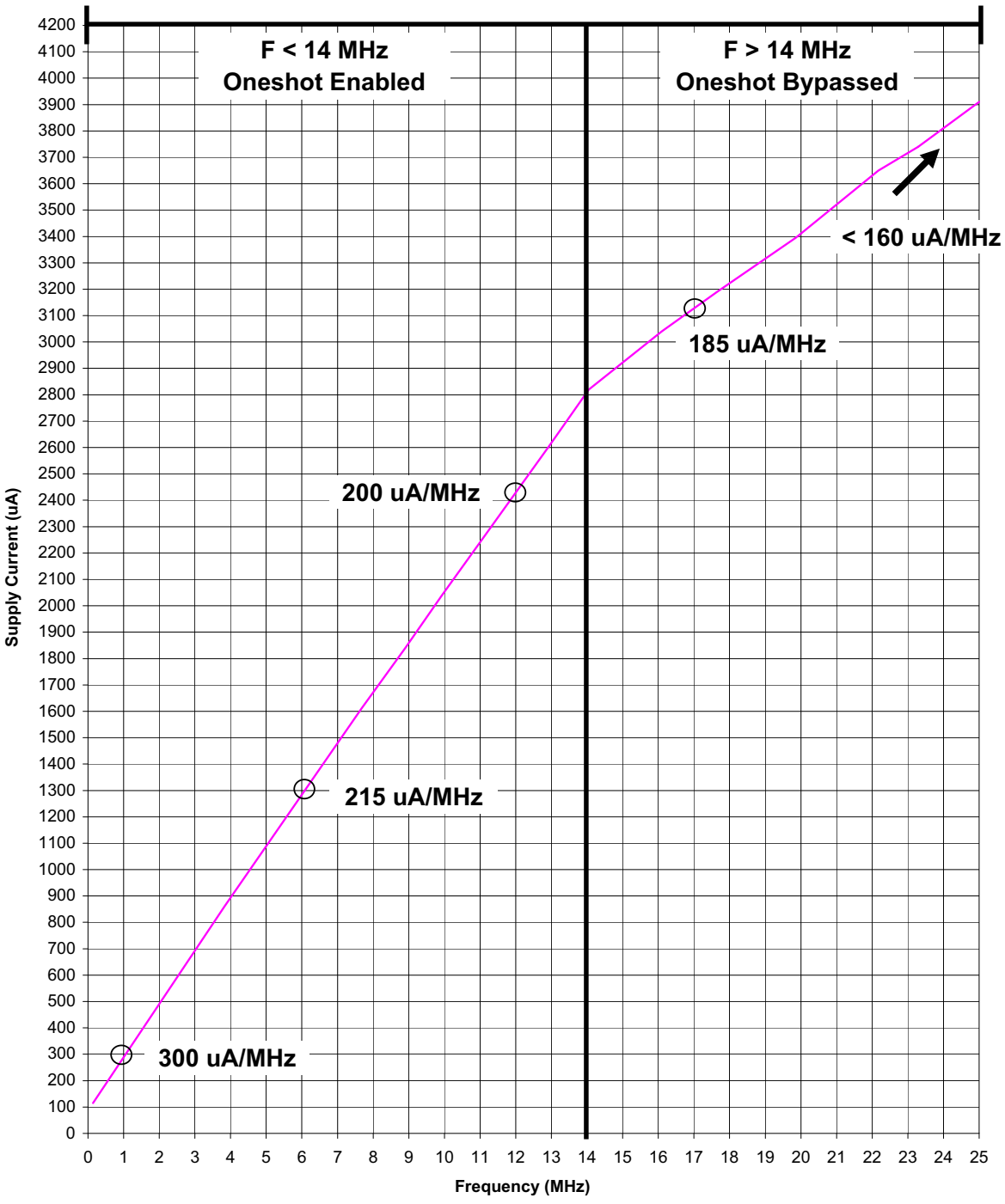


**Figure 4.1. Active Mode Current (External CMOS Clock)**

### Table 4.9. SmaRTClock Characteristics

$V_{DD}$ = 1.8 to 3.6 V; $T_A$ = –40 to +85 °C unless otherwise specified; Using factory-calibrated settings.

| Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| **Oscillator Frequency (LFO)** | | **13.1** | **16.4** | **19.7** | kHz |

**Note:** **Blue** indicates a feature only available on ‘F912 and ‘F902 devices.

### Table 4.10. ADC0 Electrical Characteristics

$V_{DD}$ = 1.8 to 3.6V V, VREF = 1.65 V (REFSL[1:0] = 11), –40 to +85 °C unless otherwise specified.

| Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| **DC Accuracy** | | | | | |
| Resolution | **12-bit mode**<br>10-bit mode | | **12**<br>10 | | bits |
| Integral Nonlinearity | **12-bit mode** [2]<br>10-bit mode | —<br>— | **±1**<br>±0.5 | **±1.5**<br>±1 | LSB |
| Differential Nonlinearity (Guaranteed Monotonic) | **12-bit mode**[2]<br>10-bit mode | —<br>— | **±0.8**<br>±0.5 | **±1**<br>±1 | LSB |
| Offset Error | **12-bit mode**<br>10-bit mode | —<br>— | **±<1**<br>±<1 | **±2**<br>±2 | LSB |
| Full Scale Error | **12-bit mode**[3]<br>10-bit mode | —<br>— | **±1**<br>±1 | **±4**<br>±2.5 | LSB |
| **Dynamic performance (10 kHz sine-wave single-ended input, 1 dB below Full Scale, maximum sampling rate)** | | | | | |
| Signal-to-Noise Plus Distortion[4] | **12-bit mode**<br>10-bit mode | **62**<br>54 | **65**<br>58 | —<br>— | dB |
| Signal-to-Distortion[4] | **12-bit mode**<br>10-bit mode | —<br>— | **76**<br>73 | —<br>— | dB |
| Spurious-Free Dynamic Range[4] | **12-bit mode**<br>10-bit mode | —<br>— | **82**<br>75 | —<br>— | dB |
| **Conversion Rate** | | | | | |
| SAR Conversion Clock | Normal Mode<br>**Low Power Mode** | —<br>— | —<br>— | 8.33<br>**4.4** | MHz |
| Conversion Time in SAR Clocks | 10-bit Mode<br>8-bit Mode | 13<br>11 | —<br>— | —<br>— | clocks |
| Track/Hold Acquisition Time | Initial Acquisition<br>Subsequent Acquisitions (DC input, burst mode) | 1.5<br>1.1 | — | — | us |
| Throughput Rate | **12-bit mode**<br>10-bit mode | —<br>— | —<br>— | **75**<br>300 | ksps |

**Notes:**

1. **Blue** indicates a feature only available on ‘F912 and ‘F902 devices.
2. INL and DNL specifications for 12-bit mode do not include the first or last four ADC codes.
3. The maximum code in 12-bit mode is 0xFFFC. The Full Scale Error is referenced from the maximum code.
4. Performance in 8-bit mode is similar to 10-bit mode.

## 5.8. Temperature Sensor

An on-chip temperature sensor is included on the C8051F91x-C8051F90x which can be directly accessed via the ADC multiplexer in single-ended configuration. To use the ADC to measure the temperature sensor, the ADC mux channel should select the temperature sensor. The temperature sensor transfer function is shown in Figure 5.8. The output voltage ($V_{TEMP}$) is the positive ADC input when the ADC multiplexer is set correctly. The TEMPE bit in register REF0CN enables/disables the temperature sensor, as described in SFR Definition 5.15. While disabled, the temperature sensor defaults to a high impedance state and any ADC measurements performed on the sensor will result in meaningless data. Refer to Table 4.11 for the slope and offset parameters of the temperature sensor.

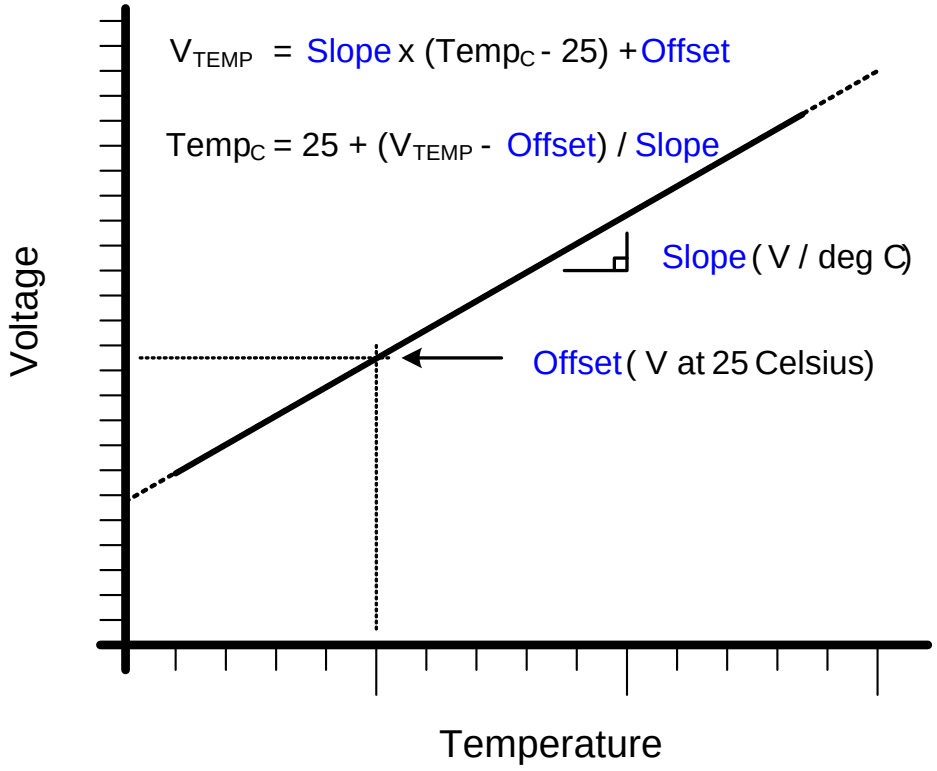


**Figure 5.8. Temperature Sensor Transfer Function**

**Table 8.1. CIP-51 Instruction Set Summary (Continued)**

| Mnemonic | Description | Bytes | Clock Cycles |
|---|---|---|---|
| ANL C, /bit | AND complement of direct bit to Carry | 2 | 2 |
| ORL C, bit | OR direct bit to carry | 2 | 2 |
| ORL C, /bit | OR complement of direct bit to Carry | 2 | 2 |
| MOV C, bit | Move direct bit to Carry | 2 | 2 |
| MOV bit, C | Move Carry to direct bit | 2 | 2 |
| JC rel | Jump if Carry is set | 2 | 2/3 |
| JNC rel | Jump if Carry is not set | 2 | 2/3 |
| JB bit, rel | Jump if direct bit is set | 3 | 3/4 |
| JNB bit, rel | Jump if direct bit is not set | 3 | 3/4 |
| JBC bit, rel | Jump if direct bit is set and clear bit | 3 | 3/4 |
| **Program Branching** | | | |
| ACALL addr11 | Absolute subroutine call | 2 | 3 |
| LCALL addr16 | Long subroutine call | 3 | 4 |
| RET | Return from subroutine | 1 | 5 |
| RETI | Return from interrupt | 1 | 5 |
| AJMP addr11 | Absolute jump | 2 | 3 |
| LJMP addr16 | Long jump | 3 | 4 |
| SJMP rel | Short jump (relative address) | 2 | 3 |
| JMP @A+DPTR | Jump indirect relative to DPTR | 1 | 3 |
| JZ rel | Jump if A equals zero | 2 | 2/3 |
| JNZ rel | Jump if A does not equal zero | 2 | 2/3 |
| CJNE A, direct, rel | Compare direct byte to A and jump if not equal | 3 | 3/4 |
| CJNE A, #data, rel | Compare immediate to A and jump if not equal | 3 | 3/4 |
| CJNE Rn, #data, rel | Compare immediate to Register and jump if not equal | 3 | 3/4 |
| CJNE @Ri, #data, rel | Compare immediate to indirect and jump if not equal | 3 | 4/5 |
| DJNZ Rn, rel | Decrement Register and jump if not zero | 2 | 2/3 |
| DJNZ direct, rel | Decrement direct byte and jump if not zero | 3 | 3/4 |
| NOP | No operation | 1 | 1 |

**Table 11.3. Special Function Registers (Continued)**

SFRs are listed in alphabetical order. All undefined SFR locations are reserved. SFRs highlighted in **blue** are only available on ‘F912 and ‘F902 devices.

| Register | Address | SFR Page | Description | Page |
|---|---|---|---|---|
| PCA0MD | 0xD9 | 0x0 | PCA0 Mode | 307 |
| PCA0PWM | 0xDF | 0x0 | PCA0 PWM Configuration | 308 |
| PCON | 0x87 | 0x0 | Power Control | 151 |
| PMU0CF | 0xB5 | 0x0 | PMU0 Configuration | 149 |
| **PMU0MD** | 0xB5 | 0xF | **PMU0 Mode** | 150 |
| PSCTL | 0x8F | 0x0 | Program Store R/W Control | 140 |
| PSW | 0xD0 | All | Program Status Word | 106 |
| REF0CN | 0xD1 | 0x0 | Voltage Reference Control | 85 |
| REG0CN | 0xC9 | 0x0 | Voltage Regulator (VREG0) Control | 170 |
| RSTSRC | 0xEF | 0x0 | Reset Source Configuration/Status | 178 |
| RTC0ADR | 0xAC | 0x0 | RTC0 Address | 193 |
| RTC0DAT | 0xAD | 0x0 | RTC0 Data | 193 |
| RTC0KEY | 0xAE | 0x0 | RTC0 Key | 192 |
| SBUF0 | 0x99 | 0x0 | UART0 Data Buffer | 253 |
| SCON0 | 0x98 | 0x0 | UART0 Control | 252 |
| SFRPAGE | 0xA7 | All | SFR Page | 115 |
| SMB0ADM | 0xF5 | 0x0 | SMBus Slave Address Mask | 237 |
| SMB0ADR | 0xF4 | 0x0 | SMBus Slave Address | 237 |
| SMB0CF | 0xC1 | 0x0 | SMBus Configuration | 232 |
| SMB0CN | 0xC0 | 0x0 | SMBus Control | 234 |
| SMB0DAT | 0xC2 | 0x0 | SMBus Data | 238 |
| SP | 0x81 | All | Stack Pointer | 105 |
| SPI0CFG | 0xA1 | 0x0 | SPI0 Configuration | 263 |
| SPI0CKR | 0xA2 | 0x0 | SPI0 Clock Rate Control | 265 |
| SPI0CN | 0xF8 | 0x0 | SPI0 Control | 264 |
| SPI0DAT | 0xA3 | 0x0 | SPI0 Data | 266 |
| SPI1CFG | 0x84 | 0x0 | SPI1 Configuration | 263 |
| SPI1CKR | 0x85 | 0x0 | SPI1 Clock Rate Control | 265 |
| SPI1CN | 0xB0 | 0x0 | SPI1 Control | 264 |
| SPI1DAT | 0x86 | 0x0 | SPI1 Data | 266 |
| TCON | 0x88 | 0x0 | Timer/Counter Control | 276 |
| TH0 | 0x8C | 0x0 | Timer/Counter 0 High | 279 |
| TH1 | 0x8D | 0x0 | Timer/Counter 1 High | 279 |

## 12.3. Interrupt Priorities

Each interrupt source can be individually programmed to one of two priority levels: low or high. A low priority interrupt service routine can be preempted by a high priority interrupt. A high priority interrupt cannot be preempted. If a high priority interrupt preempts a low priority interrupt, the low priority interrupt will finish execution after the high priority interrupt completes. Each interrupt has an associated interrupt priority bit in in the Interrupt Priority and Extended Interrupt Priority registers used to configure its priority level. Low priority is the default.

If two interrupts are recognized simultaneously, the interrupt with the higher priority is serviced first. If both interrupts have the same priority level, a fixed priority order is used to arbitrate. See Table 12.1 on page 122 to determine the fixed priority order used to arbitrate between simultaneously recognized interrupts.

## 12.4. Interrupt Latency

Interrupt response time depends on the state of the CPU when the interrupt occurs. Pending interrupts are sampled and priority decoded each system clock cycle. Therefore, the fastest possible response time is 7 system clock cycles: 1 clock cycle to detect the interrupt, 1 clock cycle to execute a single instruction, and 5 clock cycles to complete the LCALL to the ISR. If an interrupt is pending when a RETI is executed, a single instruction is executed before an LCALL is made to service the pending interrupt. Therefore, the maximum response time for an interrupt (when no other interrupt is currently being serviced or the new interrupt is of greater priority) occurs when the CPU is performing an RETI instruction followed by a DIV as the next instruction. In this case, the response time is 19 system clock cycles: 1 clock cycle to detect the interrupt, 5 clock cycles to execute the RETI, 8 clock cycles to complete the DIV instruction and 5 clock cycles to execute the LCALL to the ISR. If the CPU is executing an ISR for an interrupt with equal or higher priority, the new interrupt will not be serviced until the current ISR completes, including the RETI and following instruction.

## 18.2. Power-Fail (VDD/DC+ Supply Monitor) Reset

C8051F91x-C8051F90x devices have a VDD/DC+ Supply Monitor that is enabled and selected as a reset source after each power-on or power-fail reset. When enabled and selected as a reset source, any power down transition or power irregularity that causes VDD/DC+ to drop below $V_{RST}$ will cause the $\overline{RST}$ pin to be driven low and the CIP-51 will be held in a reset state (see Figure 18.3). When VDD/DC+ 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 invalid, and the VDD/DC+ supply monitor is enabled and selected as a reset source. The enable state of the VDD/DC+ supply monitor and its selection as a reset source is only altered by power-on and power-fail resets. For example, if the VDD/DC+ supply monitor is de-selected as a reset source and disabled by software, then a software reset is performed, the VDD/DC+ 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 and the contents of RAM are preserved as long as the VBAT 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 VDD/DC+ 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 VDD/DC+ supply falls below the $V_{WARN}$ threshold. The VDDOK bit can be configured to generate an interrupt. See Section "12. Interrupt Handler" on page 120 for more details.

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

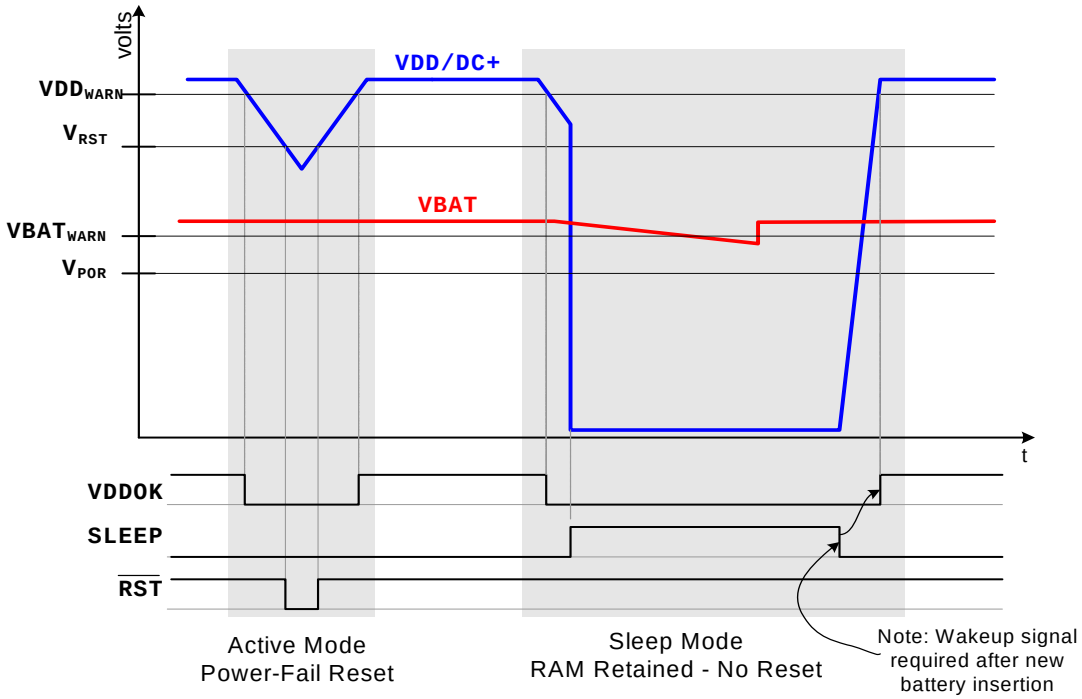


**Figure 18.3. Power-Fail Reset Timing Diagram**

## 20.2. 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 (‘F912 and ‘F902 devices only), 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 “25. Timers” on page 270 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 “19. Clocking Sources” on page 179 for information on selecting the system clock source and Section “21. Port Input/Output” on page 205 for information on how to route the system clock to a port pin. On ‘F912 and ‘F902 devices, the SmaRTClock timebase can be routed to a port pin while the device is in its ultra low power sleep mode. See the PMU0MD register description for details.

### 20.2.1. 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. Set SmaRTClock to Crystal Mode (XMODE = 1).
2. Disable Automatic Gain Control (AGCEN) and enable Bias Doubling (BIASX2) for fast crystal startup.
3. Set the desired loading capacitance (RTC0XCF).
4. Enable power to the SmaRTClock oscillator circuit (RTC0EN = 1).
5. Wait 20 ms.
6. Poll the SmaRTClock Clock Valid Bit (CLKVLD) until the crystal oscillator stabilizes.
7. Poll the SmaRTClock Load Capacitance Ready Bit (LOADRDY) until the load capacitance reaches its programmed value.
8. Enable Automatic Gain Control (AGCEN) and disable Bias Doubling (BIASX2) for maximum power savings.
9. Enable the SmaRTClock missing clock detector.
10. Wait 2 ms.
11. 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. XTAL4 should be left floating. The input low voltage (VIL) and input high voltage (VIH) for XTAL3 when used with an external CMOS clock are 0.1 and 0.8 V, respectively. 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.

### 21.1.3. Interfacing Port I/O to 5 V and 3.3 V Logic

All Port I/O configured for digital, open-drain operation are capable of interfacing to digital logic operating at a supply voltage higher than 4.5 V and less than 5.25 V. When the supply voltage is in the range of 1.8 to 2.2 V, the I/O may also interface to digital logic operating between 3.0 to 3.6 V if the input signal frequency is less than 12.5 MHz or less than 25 MHz if the signal rise time (10% to 90%) is less than 1.2 ns. When operating at a supply voltage above 2.2 V, the device should not interface to 3.3 V logic; however, interfacing to 5 V logic is permitted. An external pull-up resistor to the higher supply voltage is typically required for most systems.

**Important Note:**

- When interfacing to a signal that is between 4.5 and 5.25 V, the maximum clock frequency that may be input on a GPIO pin is 12.5 MHz. The exception to this rule is when routing an external CMOS clock to P0.3, in which case, a signal up to 25 MHz is valid as long as the rise time (10% to 90%) is shorter than 1.8 ns.
- When the supply voltage is less than 2.2 V and interfacing to a signal that is between 3.0 and 3.6 V, the maximum clock frequency that may be input on a GPIO pin is 3.125 MHz. The exception to this rule is when routing an external CMOS clock to P0.3, in which case, a signal up to 25 MHz is valid as long as the rise time (10% to 90%) is shorter than 1.2 ns.
- In a multi-voltage interface, the external pull-up resistor should be sized to allow a current of at least 150 µA to flow into the Port pin when the supply voltage is between (VDD/DC+ plus 0.4 V) and (VDD/DC+ plus 1.0 V). Once the Port pad voltage increases beyond this range, the current flowing into the Port pin is minimal.
- These guidelines only apply to multi-voltage interfaces. Port I/Os may always interface to digital logic operating at the same supply voltage.

### 21.1.4. Increasing Port I/O Drive Strength

Port I/O output drivers support a high and low drive strength; the default is low drive strength. The drive strength of a Port I/O can be configured using the PnDRV registers. See Section “4. Electrical Characteristics” on page 36 for the difference in output drive strength between the two modes.

## 21.2. Assigning Port I/O Pins to Analog and Digital Functions

Port I/O pins P0.0–P1.6 can be assigned to various analog, digital, and external interrupt functions. The Port pins assuaged to analog functions should be configured for analog I/O and Port pins assuaged to digital or external interrupt functions should be configured for digital I/O.

### 21.2.1. Assigning Port I/O Pins to Analog Functions

Table 21.1 shows all available analog functions that need Port I/O assignments. **Port pins selected for these analog functions should have their digital drivers disabled (PnMDOUT.n = 0 and Port Latch = 1) and their corresponding bit in PnSKIP set to 1.** This reserves the pin for use by the analog function and does not allow it to be claimed by the Crossbar. Table 21.1 shows the potential mapping of Port I/O to each analog function.

**Table 21.1. Port I/O Assignment for Analog Functions**

| Analog Function | Potentially Assignable Port Pins | SFR(s) used for Assignment |
|---|---|---|
| ADC Input | P0.0–P1.6 | ADC0MX, PnSKIP |
| Comparator0 Input | P0.0–P1.6 | CPT0MX, PnSKIP |
| Comparator1 Input | P0.0–P1.6 | CPT1MX, PnSKIP |

## SFR Definition 21.10. P0MDIN: Port0 Input Mode

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | P0MDIN[7:0] | | | | | | | |
| Type | R/W | | | | | | | |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

SFR Page= 0x0; SFR Address = 0xF1

| Bit | Name | Function |
|---|---|---|
| 7:0 | P0MDIN[7:0] | **Analog Configuration Bits for P0.7–P0.0 (respectively).**<br>Port pins configured for analog mode have their weak pullup, and digital receiver disabled. The digital driver is not explicitly disabled.<br>0: Corresponding P0.n pin is configured for analog mode.<br>1: Corresponding P0.n pin is not configured for analog mode. |

## SFR Definition 21.11. P0MDOUT: Port0 Output Mode

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | P0MDOUT[7:0] | | | | | | | |
| Type | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Page = 0x0; SFR Address = 0xA4

| Bit | Name | Function |
|---|---|---|
| 7:0 | P0MDOUT[7:0] | **Output Configuration Bits for P0.7–P0.0 (respectively).**<br>These bits control the digital driver even when the corresponding bit in register P0MDIN is logic 0.<br>0: Corresponding P0.n Output is open-drain.<br>1: Corresponding P0.n Output is push-pull. |

## SFR Definition 21.17. P1DRV: Port1 Drive Strength

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | | P1DRV[6:0] | | | | | | |
| Type | | R/W | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Page = 0xF; SFR Address = 0xA5

| Bit | Name | Function |
|---|---|---|
| 7 | Unused | **Unused.**<br>Read =0b; Write = Don't Care. |
| 6:0 | P1DRV[6:0] | **Drive Strength Configuration Bits for P1.6–P1.0 (respectively).**<br>Configures digital I/O Port cells to high or low output drive strength.<br>0: Corresponding P1.n Output has low output drive strength.<br>1: Corresponding P1.n Output has high output drive strength. |

## SFR Definition 21.18. P2: Port2

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | P2 | | | | | | | |
| Type | R/W | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SFR Page = All Pages; SFR Address = 0xA0; Bit-Addressable

| Bit | Name | Description | Read | Write |
|---|---|---|---|---|
| 7 | P2 | **Port 2 Data.**<br>Sets the Port latch logic value or reads the Port pin logic state in Port cells configured for digital I/O. | 0: Set output latch to logic LOW.<br>1: Set output latch to logic HIGH. | 0: P2.7 Port pin is logic LOW.<br>1: P2.7 Port pin is logic HIGH. |
| 6:0 | Unused | **Unused.**<br>Read = 0000000b; Write = Don't Care. | | |

### 22.5.3. Write Sequence (Slave)

During a write sequence, an SMBus master writes data to a slave device. The slave in this transfer will be a receiver during the address byte, and a receiver during all data bytes. When slave events are enabled (INH = 0), the interface enters Slave Receiver Mode when a START followed by a slave address and direction bit (WRITE in this case) is received. If hardware ACK generation is disabled, upon entering Slave Receiver Mode, an interrupt is generated and the ACKRQ bit is set. The software must respond to the received slave address with an ACK, or ignore the received slave address with a NACK. If hardware ACK generation is enabled, the hardware will apply the ACK for a slave address which matches the criteria set up by SMB0ADR and SMB0ADM. The interrupt will occur after the ACK cycle.

If the received slave address is ignored (by software or hardware), slave interrupts will be inhibited until the next START is detected. If the received slave address is acknowledged, zero or more data bytes are received.

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.**

The interface exits Slave Receiver Mode after receiving a STOP. Note that the interface will switch to Slave Transmitter Mode if SMB0DAT is written while an active Slave Receiver. Figure 22.7 shows a typical slave write 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.

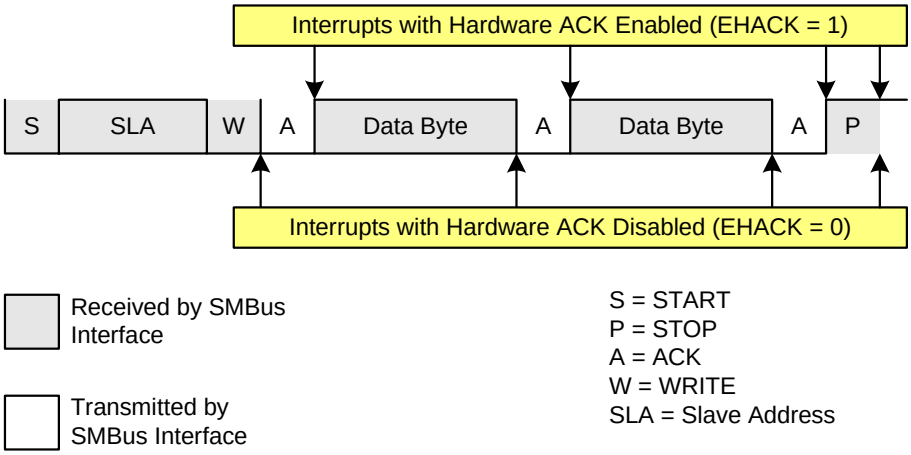


**Figure 22.7. Typical Slave Write Sequence**

### 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, SmaRTClock divided by 8 or Comparator 0 output. 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 bits (T2XCLK[1:0] in TMR2CN), as follows:

| T2MH | T2XCLK[1:0] | TMR2H Clock Source |
|---|---|---|
| 0 | 00 | SYSCLK / 12 |
| 0 | 01 | SmaRTClock / 8 |
| 0 | 10 | Reserved |
| 0 | 11 | Comparator 0 |
| 1 | X | SYSCLK |

| T2ML | T2XCLK[1:0] | TMR2L Clock Source |
|---|---|---|
| 0 | 00 | SYSCLK / 12 |
| 0 | 01 | SmaRTClock / 8 |
| 0 | 10 | Reserved |
| 0 | 11 | Comparator 0 |
| 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.

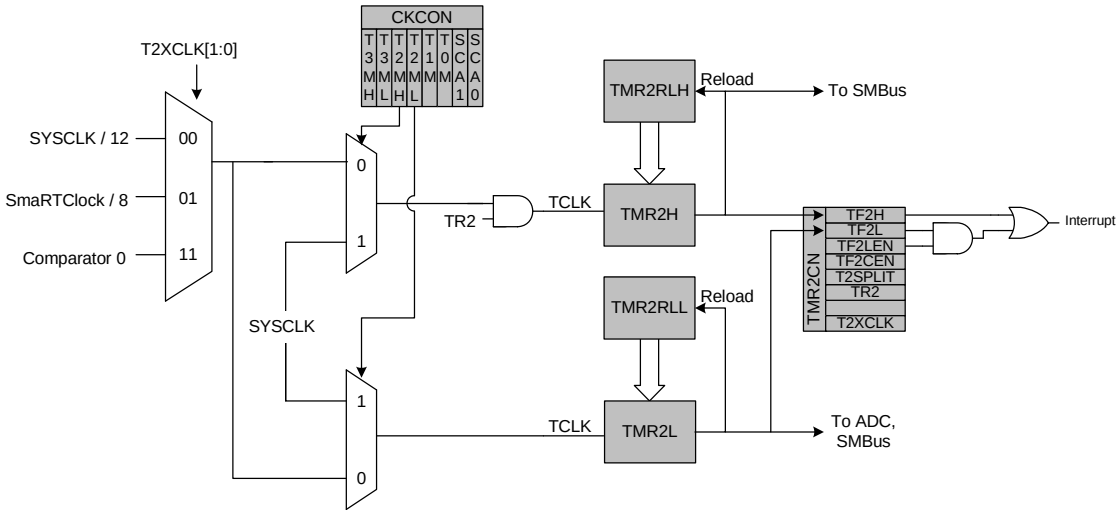


**Figure 25.5. Timer 2 8-Bit Mode Block Diagram**

### 25.3.2. 8-bit Timers with Auto-Reload

When T3SPLIT is set, Timer 3 operates as two 8-bit timers (TMR3H and TMR3L). Both 8-bit timers operate in auto-reload mode as shown in Figure 25.8. TMR3RLL holds the reload value for TMR3L; TMR3RLH holds the reload value for TMR3H. The TR3 bit in TMR3CN handles the run control for TMR3H. TMR3L is always running when configured for 8-bit Mode.

Each 8-bit timer may be configured to use SYSCLK, SYSCLK divided by 12, the external oscillator clock source divided by 8, or Comparator 1. The Timer 3 Clock Select bits (T3MH and T3ML in CKCON) select either SYSCLK or the clock defined by the Timer 3 External Clock Select bits (T3XCLK[1:0] in TMR3CN), as follows:

| T3MH | T3XCLK[1:0] | TMR3H Clock Source |
|---|---|---|
| 0 | 00 | SYSCLK / 12 |
| 0 | 01 | Comparator 1 |
| 0 | 10 | Reserved |
| 0 | 11 | External Clock / 8 |
| 1 | X | SYSCLK |

| T3ML | T3XCLK[1:0] | TMR3L Clock Source |
|---|---|---|
| 0 | 00 | SYSCLK / 12 |
| 0 | 01 | Comparator 1 |
| 0 | 10 | Reserved |
| 0 | 11 | External Clock / 8 |
| 1 | X | SYSCLK |

The TF3H bit is set when TMR3H overflows from 0xFF to 0x00; the TF3L bit is set when TMR3L overflows from 0xFF to 0x00. When Timer 3 interrupts are enabled, an interrupt is generated each time TMR3H overflows. If Timer 3 interrupts are enabled and TF3LEN (TMR3CN.5) is set, an interrupt is generated each time either TMR3L or TMR3H overflows. When TF3LEN is enabled, software must check the TF3H and TF3L flags to determine the source of the Timer 3 interrupt. The TF3H and TF3L interrupt flags are not cleared by hardware and must be manually cleared by software.

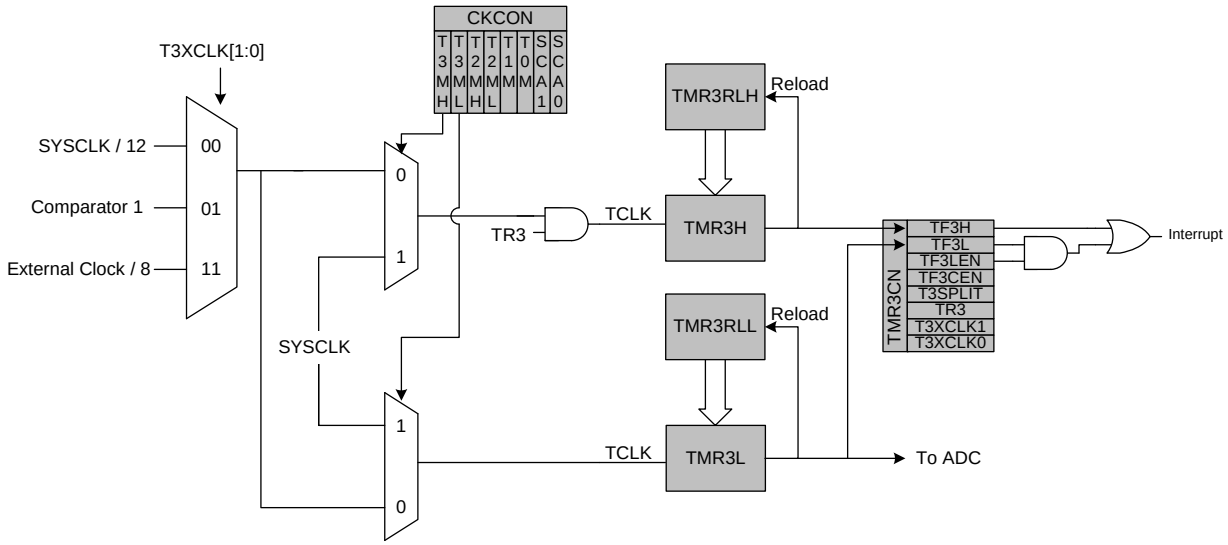


**Figure 25.8. Timer 3 8-Bit Mode Block Diagram.**

### 26.3.2. Software Timer (Compare) Mode

In Software Timer mode, the PCA counter/timer value is compared to the module's 16-bit capture/compare register (PCA0CPHn and PCA0CPLn). When a match occurs, the Capture/Compare Flag (CCFn) in PCA0CN is set to logic 1. An interrupt request is generated if the CCFn interrupt for that module is enabled. The CCFn bit is not automatically cleared by hardware when the CPU vectors to the interrupt service routine, and must be cleared by software. Setting the ECOMn and MATn bits in the PCA0CPMn register enables Software Timer 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.

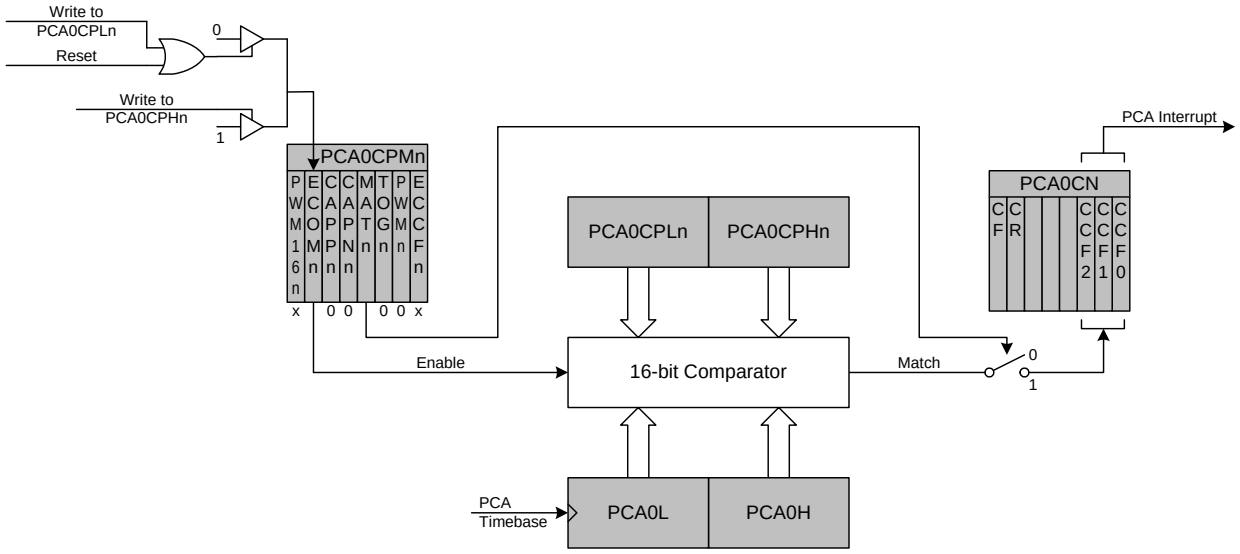


**Figure 26.5. PCA Software Timer Mode Diagram**

## C2 Register Definition 27.4. FPCTL: C2 Flash Programming Control

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

C2 Address: 0x02

| Bit | Name | Function |
|---|---|---|
| 7:0 | FPCTL[7:0] | **Flash Programming Control Register.**<br>This register is used to enable Flash programming via the C2 interface. To enable C2 Flash programming, the following codes must be written in order: 0x02, 0x01. Note that once C2 Flash programming is enabled, a system reset must be issued to resume normal operation. |

## C2 Register Definition 27.5. FPDAT: C2 Flash Programming Data

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

C2 Address: 0xB4

| Bit | Name | Function | |
|---|---|---|---|
| 7:0 | FPDAT[7:0] | **C2 Flash Programming Data Register.**<br>This register is used to pass Flash commands, addresses, and data during C2 Flash accesses. Valid commands are listed below. | |
| | | **Code** | **Command** |
| | | 0x06 | Flash Block Read |
| | | 0x07 | Flash Block Write |
| | | 0x08 | Flash Page Erase |
| | | 0x03 | Device Erase |