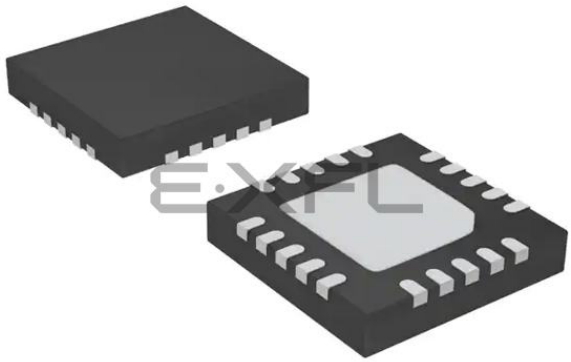


Welcome to E-XFL.COM

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	Cap Sense, POR, PWM, WDT
Number of I/O	17
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-VFQFN Exposed Pad
Supplier Device Package	20-QFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f806-gmr

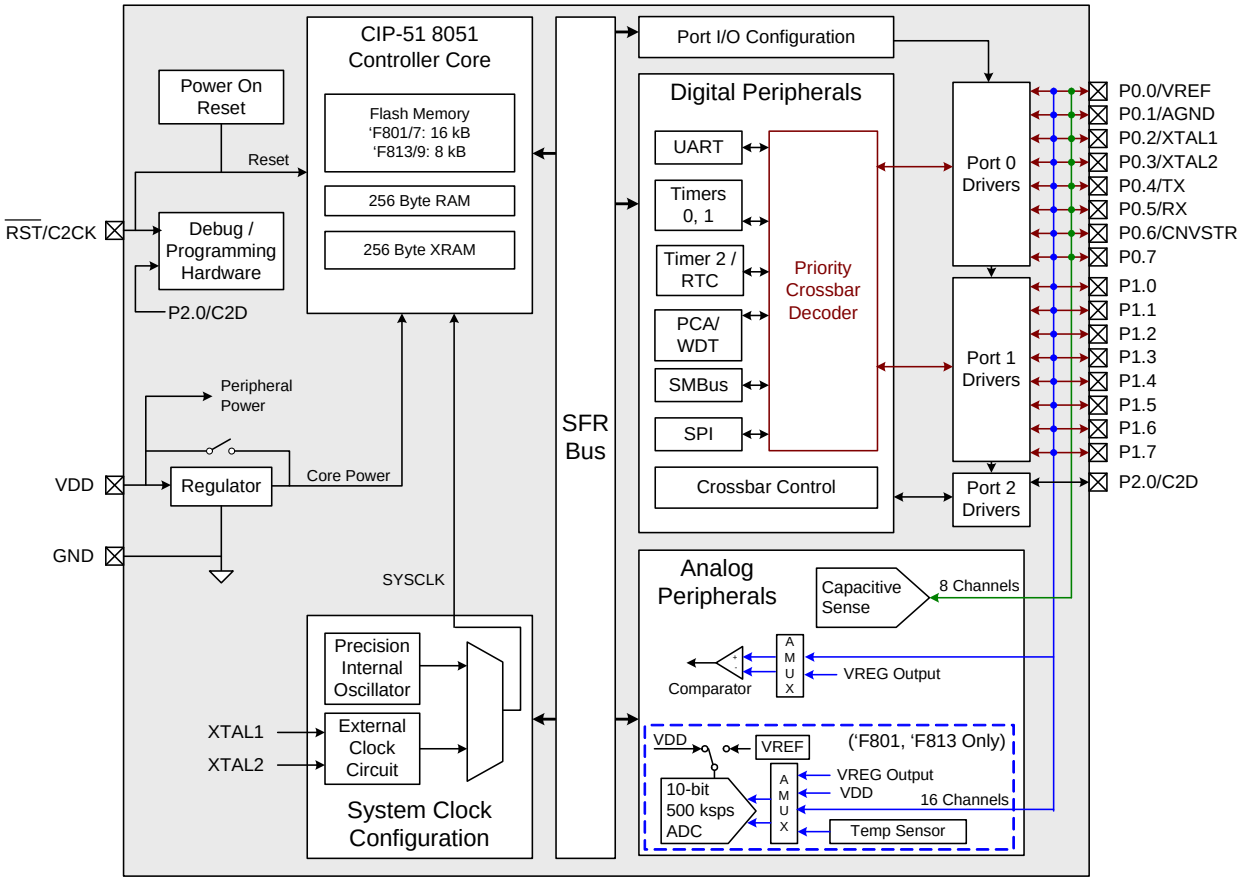


Figure 1.2. C8051F801, C8051F807, C8051F813, C8051F819 Block Diagram

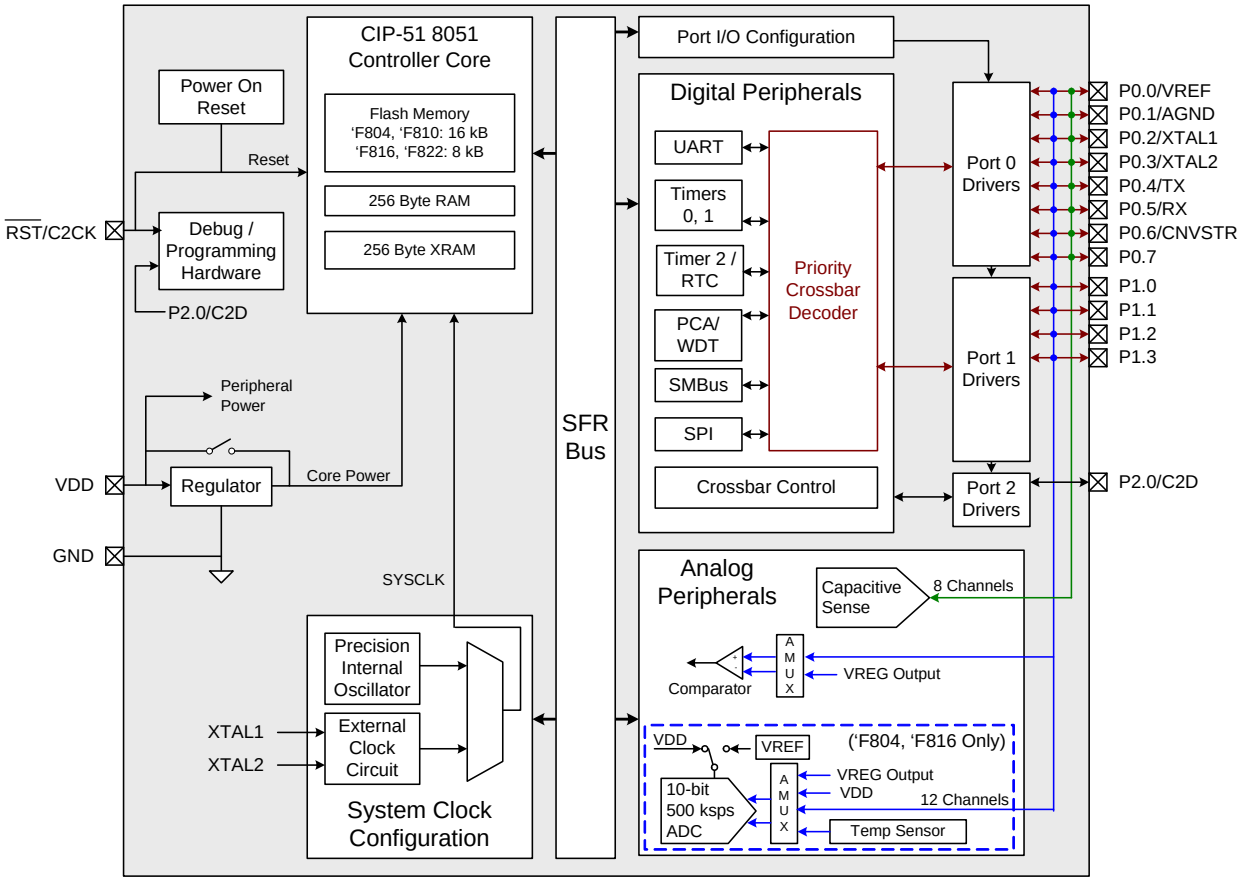


Figure 1.5. C8051F804, C8051F810, C8051F816, C8051F822 Block Diagram

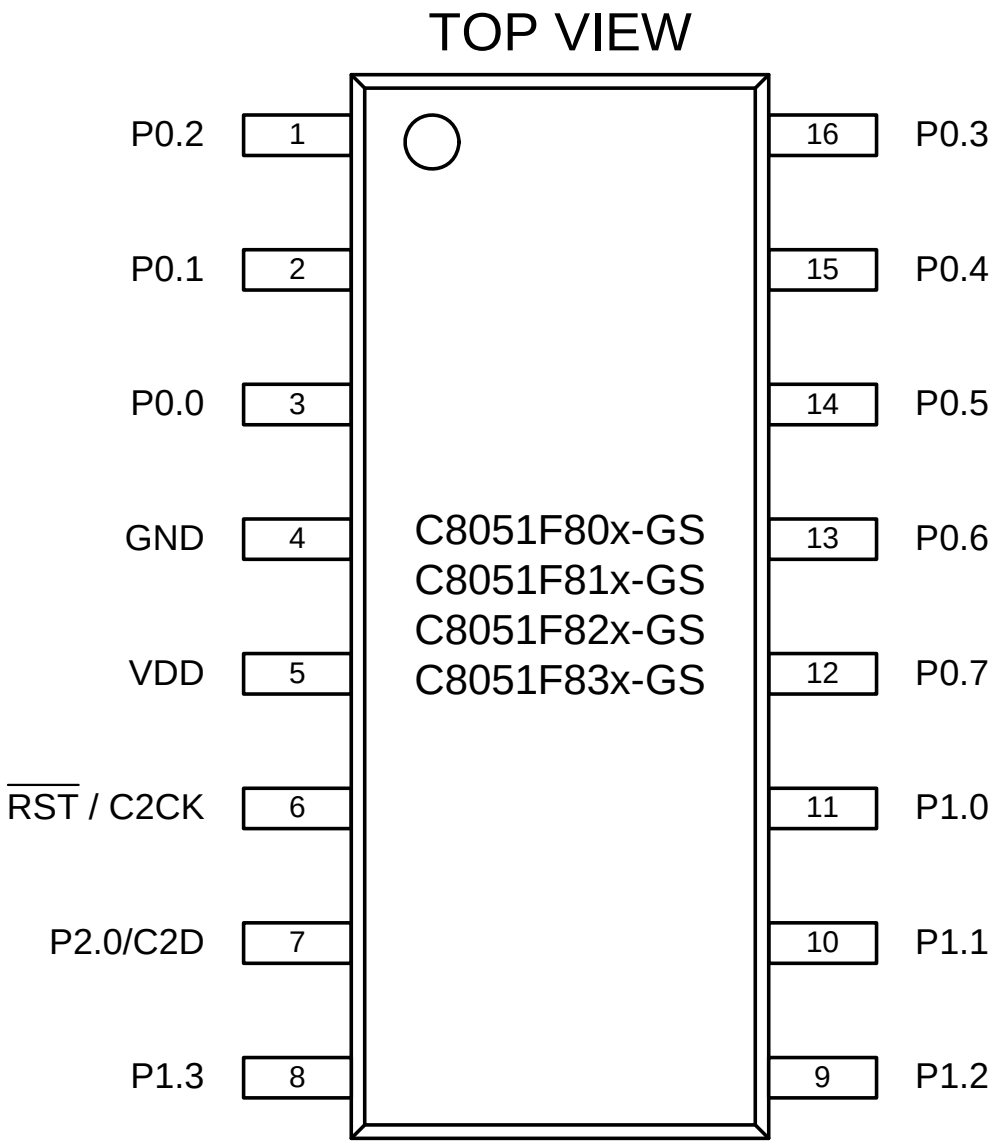


Figure 3.3. SOIC-16 Pinout Diagram (Top View)

4. QFN-20 Package Specifications

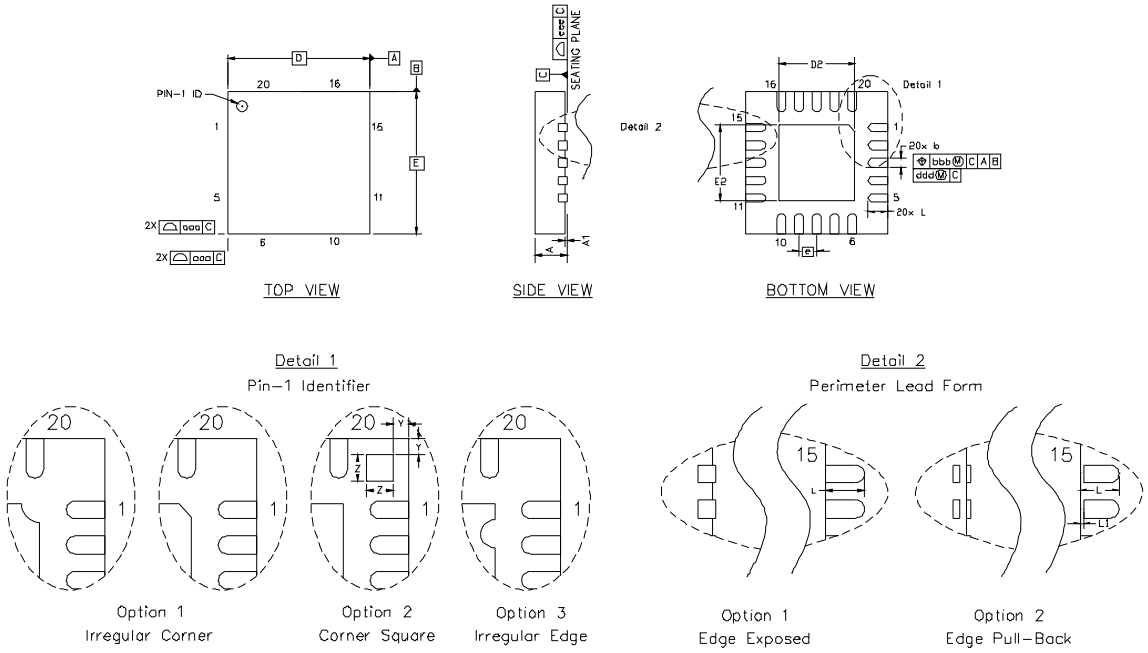


Figure 4.1. QFN-20 Package Drawing

Table 4.1. QFN-20 Package Dimensions

Dimension	Min	Typ	Max
A	0.80	0.90	1.00
A1	0.00	0.02	0.05
b	0.18	0.25	0.30
D		4.00 BSC.	
D2	2.00	2.15	2.25
e		0.50 BSC.	
E		4.00 BSC.	
E2	2.00	2.15	2.25

Dimension	Min	Typ	Max
L	0.45	0.55	0.65
L1	0.00	—	0.15
aaa	—	—	0.15
bbb	—	—	0.10
ddd	—	—	0.05
eee	—	—	0.08
Z	—	0.43	—
Y	—	0.18	—

Notes:

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. Dimensioning and Tolerancing per ANSI Y14.5M-1994.
3. This drawing conforms to the JEDEC Solid State Outline MO-220, variation VGGD except for custom features D2, E2, Z, Y, and L which are toleranced per supplier designation.
4. Recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

Table 7.13. Comparator Electrical Characteristics

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

Parameter	Conditions	Min	Typ	Max	Units
Response Time: Mode 0, Vcm* = 1.5 V	CP0+ – CP0– = 100 mV	—	220	—	ns
	CP0+ – CP0– = –100 mV	—	225	—	ns
Response Time: Mode 1, Vcm* = 1.5 V	CP0+ – CP0– = 100 mV	—	340	—	ns
	CP0+ – CP0– = –100 mV	—	380	—	ns
Response Time: Mode 2, Vcm* = 1.5 V	CP0+ – CP0– = 100 mV	—	510	—	ns
	CP0+ – CP0– = –100 mV	—	945	—	ns
Response Time: Mode 3, Vcm* = 1.5 V	CP0+ – CP0– = 100 mV	—	1500	—	ns
	CP0+ – CP0– = –100 mV	—	5000	—	ns
Common-Mode Rejection Ratio		—	1	4	mV/V
Positive Hysteresis 1	Mode 2, CP0HYP1–0 = 00b	—	0	1	mV
Positive Hysteresis 2	Mode 2, CP0HYP1–0 = 01b	2	5	10	mV
Positive Hysteresis 3	Mode 2, CP0HYP1–0 = 10b	7	10	20	mV
Positive Hysteresis 4	Mode 2, CP0HYP1–0 = 11b	10	20	30	mV
Negative Hysteresis 1	Mode 2, CP0HYN1–0 = 00b	—	0	1	mV
Negative Hysteresis 2	Mode 2, CP0HYN1–0 = 01b	2	5	10	mV
Negative Hysteresis 3	Mode 2, CP0HYN1–0 = 10b	7	10	20	mV
Negative Hysteresis 4	Mode 2, CP0HYN1–0 = 11b	10	20	30	mV
Inverting or Non-Inverting Input Voltage Range		–0.25	—	V_{DD} + 0.25	V
Input Offset Voltage		–7.5	—	7.5	mV
Power Specifications					
Power Supply Rejection		—	0.1	—	mV/V
Powerup Time		—	10	—	µs
Supply Current at DC	Mode 0	—	20	—	µA
	Mode 1	—	8	—	µA
	Mode 2	—	3	—	µA
	Mode 3	—	0.5	—	µA

Note: Vcm is the common-mode voltage on CP0+ and CP0–.

SFR Definition 8.4. ADC0CN: ADC0 Control

Bit	7	6	5	4	3	2	1	0
Name	AD0EN	AD0TM	AD0INT	AD0BUSY	AD0WINT	AD0CM[2:0]		
Type	R/W	R/W	R/W	R/W	R/W	R/W		
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xE8; Bit-Addressable

Bit	Name	Function		
7	AD0EN	**ADC0 Enable Bit.** 0: ADC0 Disabled. ADC0 is in low-power shutdown. 1: ADC0 Enabled. ADC0 is active and ready for data conversions.		
6	AD0TM	**ADC0 Track Mode Bit.** 0: Normal Track Mode: When ADC0 is enabled, tracking is continuous unless a conversion is in progress. Conversion begins immediately on start-of-conversion event, as defined by AD0CM[2:0]. 1: Delayed Track Mode: When ADC0 is enabled, input is tracked when a conversion is not in progress. A start-of-conversion signal initiates three SAR clocks of additional tracking, and then begins the conversion.		
5	AD0INT	**ADC0 Conversion Complete Interrupt Flag.** 0: ADC0 has not completed a data conversion since AD0INT was last cleared. 1: ADC0 has completed a data conversion.		
4	AD0BUSY	**ADC0 Busy Bit.**	**Read:** 0: ADC0 conversion is not in progress. 1: ADC0 conversion is in progress.	**Write:** 0: No Effect. 1: Initiates ADC0 Conversion if AD0CM[2:0] = 000b
3	AD0WINT	**ADC0 Window Compare Interrupt Flag.** 0: ADC0 Window Comparison Data match has not occurred since this flag was last cleared. 1: ADC0 Window Comparison Data match has occurred.		
2:0	AD0CM[2:0]	**ADC0 Start of Conversion Mode Select.** 000: ADC0 start-of-conversion source is write of 1 to AD0BUSY. 001: ADC0 start-of-conversion source is overflow of Timer 0. 010: ADC0 start-of-conversion source is overflow of Timer 2. 011: ADC0 start-of-conversion source is overflow of Timer 1. 100: ADC0 start-of-conversion source is rising edge of external CNVSTR. 101–111: Reserved.		

SFR Definition 8.7. 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

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

SFR Definition 8.8. 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

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

21.2. Power-Fail Reset / V_{DD} Monitor

When a power-down transition or power irregularity causes V_{DD} to drop below V_{RST}, the power supply monitor will drive the $\overline{RST}$ pin low and hold the CIP-51 in a reset state (see Figure 21.2). When V_{DD} returns to a level above V_{RST}, the CIP-51 will be released from the reset state. Even though internal data memory contents are not altered by the power-fail reset, it is impossible to determine if V_{DD} dropped below the level required for data retention. If the PORSF flag reads 1, the data may no longer be valid. The V_{DD} monitor is enabled and selected as a reset source after power-on resets. Its defined state (enabled/disabled) is not altered by any other reset source. For example, if the V_{DD} monitor is disabled by code and a software reset is performed, the V_{DD} monitor will still be disabled after the reset.

Important Note: If the V_{DD} monitor is being turned on from a disabled state, it should be enabled before it is selected as a reset source. Selecting the V_{DD} monitor as a reset source before it is enabled and stabilized may cause a system reset. In some applications, this reset may be undesirable. If this is not desirable in the application, a delay should be introduced between enabling the monitor and selecting it as a reset source. The procedure for enabling the V_{DD} monitor and configuring it as a reset source from a disabled state is shown below:

1. Enable the V_{DD} monitor (VDMEN bit in VDM0CN = 1).
2. If necessary, wait for the V_{DD} monitor to stabilize.
3. Select the V_{DD} monitor as a reset source (PORSF bit in RSTSRC = 1).

See Figure 21.2 for V_{DD} monitor timing; note that the power-on-reset delay is not incurred after a V_{DD} monitor reset. See Section “7. Electrical Characteristics” on page 39 for complete electrical characteristics of the V_{DD} monitor.

SFR Definition 22.3. OSCICN: Internal H-F Oscillator Control

Bit	7	6	5	4	3	2	1	0
Name	IOSCEN	IFRDY	SUSPEND	STSYNC	SSE		IFCN[1:0]	
Type	R/W	R	R/W	R	R/W	R	R/W	
Reset	1	1	0	0	0	0	0	0

SFR Address = 0xB2

Bit	Name	Function
7	IOSCEN	**Internal H-F Oscillator Enable Bit.** 0: Internal H-F Oscillator Disabled. 1: Internal H-F Oscillator Enabled.
6	IFRDY	**Internal H-F Oscillator Frequency Ready Flag.** 0: Internal H-F Oscillator is not running at programmed frequency. 1: Internal H-F Oscillator is running at programmed frequency.
5	SUSPEND	**Internal Oscillator Suspend Enable Bit.** Setting this bit to logic 1 places the internal oscillator in SUSPEND mode. The internal oscillator resumes operation when one of the SUSPEND mode awakening events occurs.
4	STSYNC	**Suspend Timer Synchronization Bit.** This bit is used to indicate when it is safe to read and write the registers associated with the suspend wake-up timer. If a suspend wake-up source other than Timer 2 has brought the oscillator out of suspend mode, it make take up to three timer clocks before the timer can be read or written. 0: Timer 2 registers can be read safely. 1: Timer 2 register reads and writes should not be performed.
3	SSE	**Spread Spectrum Enable.** Spread spectrum enable bit. 0: Spread Spectrum clock dithering disabled. 1: Spread Spectrum clock dithering enabled.
2	Unused	Read = 0b; Write = Don't Care
1:0	IFCN[1:0]	**Internal H-F Oscillator Frequency Divider Control Bits.** 00: SYSCLK derived from Internal H-F Oscillator divided by 8. 01: SYSCLK derived from Internal H-F Oscillator divided by 4. 10: SYSCLK derived from Internal H-F Oscillator divided by 2. 11: SYSCLK derived from Internal H-F Oscillator divided by 1.

22.3.1. External Crystal Example

If a crystal or ceramic resonator is used as an external oscillator source for the MCU, the circuit should be configured as shown in Figure 22.1, Option 1. The External Oscillator Frequency Control value (XFCN) should be chosen from the Crystal column of the table in SFR Definition 22.4 (OSCXCN register). For example, an 11.0592 MHz crystal requires an XFCN setting of 111b and a 32.768 kHz Watch Crystal requires an XFCN setting of 001b. After an external 32.768 kHz oscillator is stabilized, the XFCN setting can be switched to 000 to save power. It is recommended to enable the missing clock detector before switching the system clock to any external oscillator source.

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

1. Force XTAL1 and XTAL2 to a low state. This involves enabling the Crossbar and writing 0 to the port pins associated with XTAL1 and XTAL2.
2. Configure XTAL1 and XTAL2 as analog inputs.
3. Enable the external oscillator.
4. Wait at least 1 ms.
5. Poll for XTLVLD = 1.
6. If desired, enable the Missing Clock Detector.
7. Switch the system clock to the external oscillator.

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

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

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

For example, a tuning-fork crystal of 32.768 kHz with a recommended load capacitance of 12.5 pF should use the configuration shown in Figure 22.1, Option 1. The total value of the capacitors and the stray capacitance of the XTAL pins should equal 25 pF. With a stray capacitance of 3 pF per pin, the 22 pF capacitors yield an equivalent capacitance of 12.5 pF across the crystal, as shown in Figure 22.2.

SFR Definition 23.14. P1SKIP: Port 1 Skip

Bit	7	6	5	4	3	2	1	0
Name	P1SKIP[7:0]							
Type	R/W							
Reset	0*	0*	0*	0*	0	0	0	0

SFR Address = 0xD5

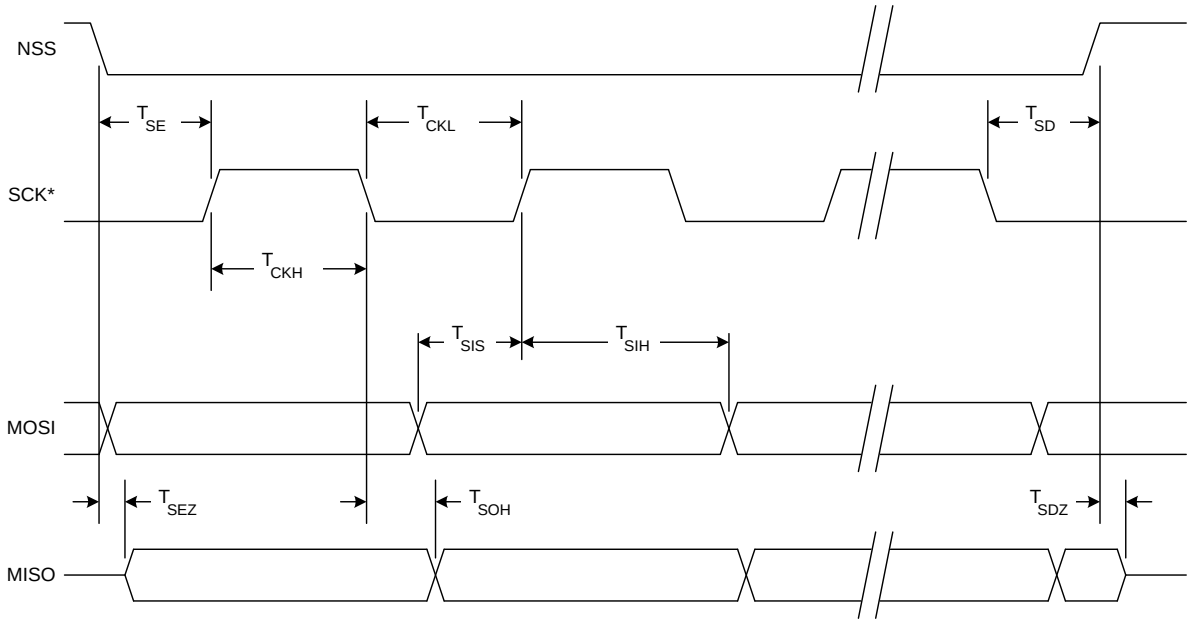
Bit	Name	Function
7:0	P1SKIP[7:0]	**Port 1 Crossbar Skip Enable Bits.** These bits select Port 1 pins to be skipped by the Crossbar Decoder. Port pins used for analog, special functions or GPIO should be skipped by the Crossbar. 0: Corresponding P1.n pin is not skipped by the Crossbar. 1: Corresponding P1.n pin is skipped by the Crossbar. **Note:** P1.4–P1.7 are not available on 16-pin packages, with the reset value of 1111b for P1SKIP[7:4].

SFR Definition 23.15. P2: Port 2

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

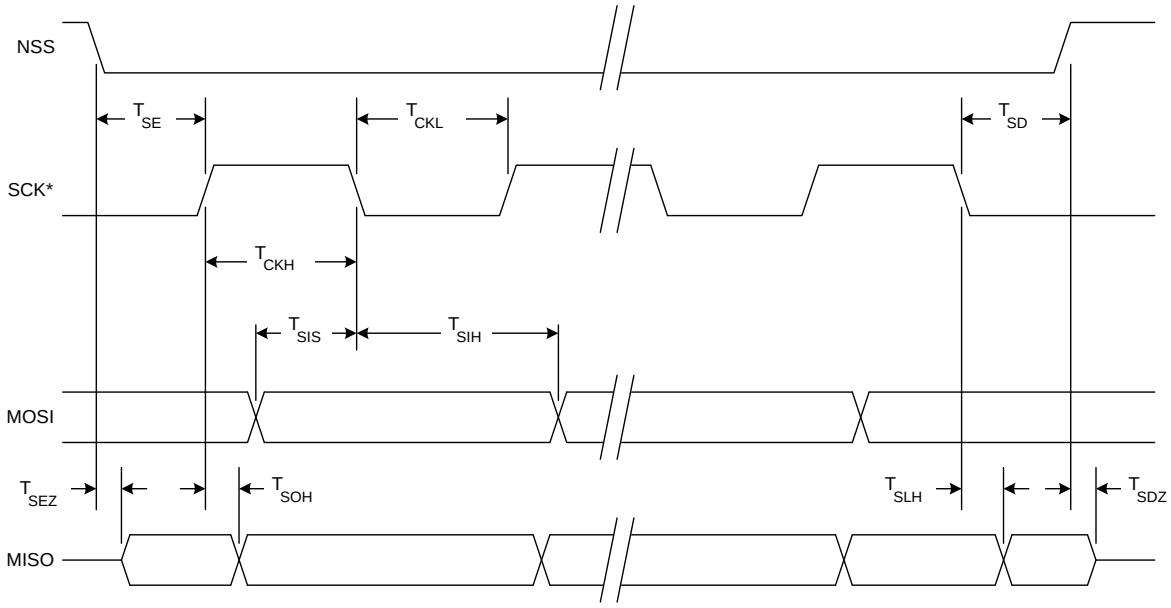
SFR Address = 0xA0; Bit-Addressable

Bit	Name	Description	Write	Read
7:1	Unused	**Unused.**	Don't Care	0000000b
0	P2[0]	**Port 2 Data.** 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. 1: Set output latch to logic HIGH.	0: P2.0 Port pin is logic LOW. 1: P2.0 Port pin is logic HIGH.



* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.

Figure 25.10. SPI Slave Timing (CKPHA = 0)



* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.

Figure 25.11. SPI Slave Timing (CKPHA = 1)

Table 26.3. Sources for Hardware Changes to SMB0CN

Bit	Set by Hardware When:	Cleared by Hardware When:
MASTER	■ A START is generated.	■ A STOP is generated. ■ Arbitration is lost.
TXMODE	■ START is generated. ■ SMB0DAT is written before the start of an SMBus frame.	■ A START is detected. ■ Arbitration is lost. ■ SMB0DAT is not written before the start of an SMBus frame.
STA	■ A START followed by an address byte is received.	■ Must be cleared by software.
STO	■ A STOP is detected while addressed as a slave. ■ Arbitration is lost due to a detected STOP.	■ A pending STOP is generated.
ACKRQ	■ A byte has been received and an ACK response value is needed (only when hardware ACK is not enabled).	■ After each ACK cycle.
ARBLOST	■ A repeated START is detected as a MASTER when STA is low (unwanted repeated START). ■ SCL is sensed low while attempting to generate a STOP or repeated START condition. ■ SDA is sensed low while transmitting a 1 (excluding ACK bits).	■ Each time SI is cleared.
ACK	■ The incoming ACK value is low (ACKNOWLEDGE).	■ The incoming ACK value is high (NOT ACKNOWLEDGE).
SI	■ A START has been generated. ■ Lost arbitration. ■ A byte has been transmitted and an ACK/NACK received. ■ A byte has been received. ■ A START or repeated START followed by a slave address + R/W has been received. ■ A STOP has been received.	■ Must be cleared by software.

26.4.3. 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 26.4.2.2.

The registers used to define which address(es) are recognized by the hardware are the SMBus Slave Address register (SFR Definition 26.3) and the SMBus Slave Address Mask register (SFR Definition 26.4). 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

26.5.2. 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 26.6 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.

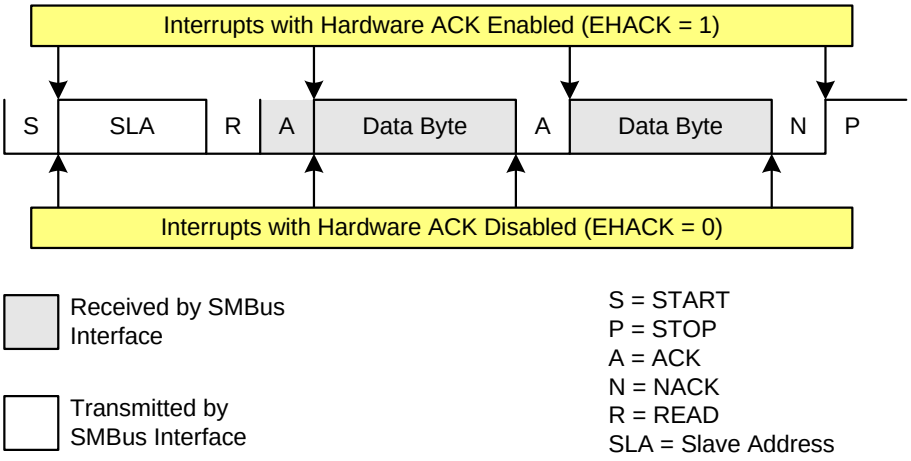


Figure 26.6. Typical Master Read Sequence

27. UART0

UART0 is an asynchronous, full duplex serial port offering modes 1 and 3 of the standard 8051 UART. Enhanced baud rate support allows a wide range of clock sources to generate standard baud rates (details in Section “27.1. Enhanced Baud Rate Generation” on page 202). Received data buffering allows UART0 to start reception of a second incoming data byte before software has finished reading the previous data byte.

UART0 has two associated SFRs: Serial Control Register 0 (SCON0) and Serial Data Buffer 0 (SBUF0). The single SBUF0 location provides access to both transmit and receive registers. **Writes to SBUF0 always access the Transmit register. Reads of SBUF0 always access the buffered Receive register; it is not possible to read data from the Transmit register.**

With UART0 interrupts enabled, an interrupt is generated each time a transmit is completed (TI0 is set in SCON0), or a data byte has been received (RI0 is set in SCON0). The UART0 interrupt flags are not cleared by hardware when the CPU vectors to the interrupt service routine. They must be cleared manually by software, allowing software to determine the cause of the UART0 interrupt (transmit complete or receive complete).

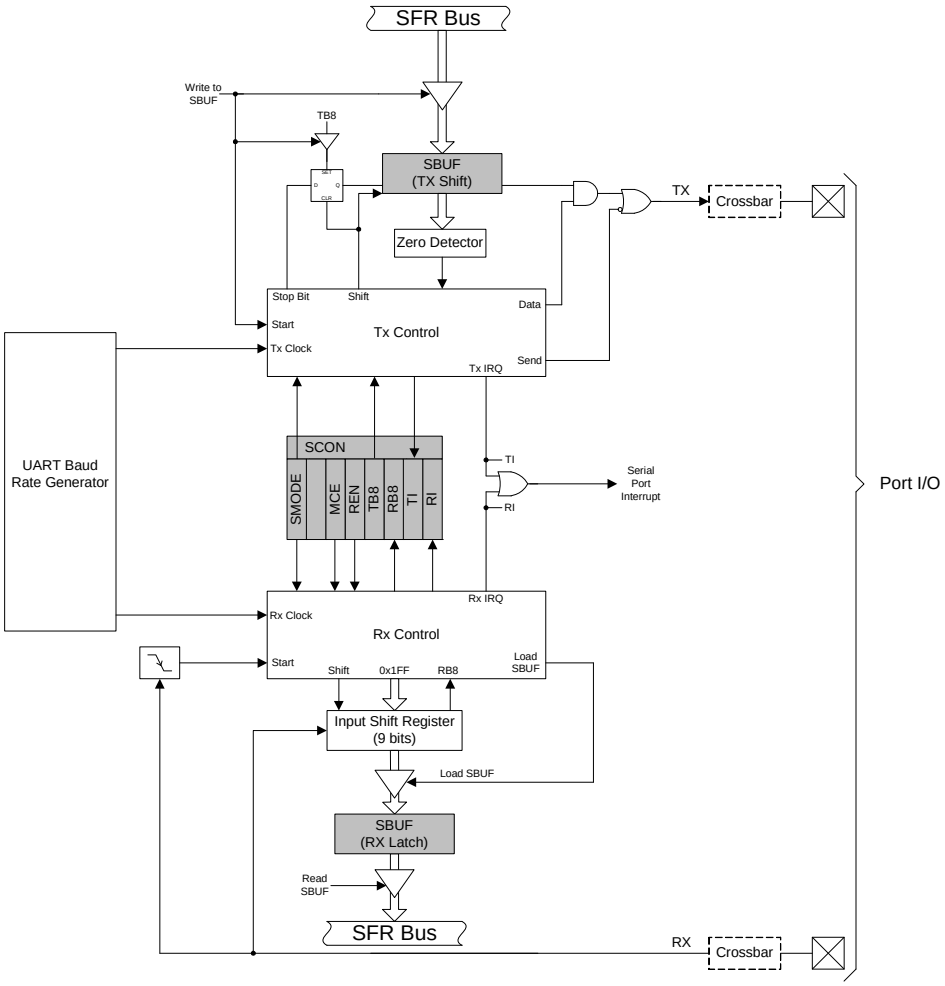


Figure 27.1. UART0 Block Diagram

27.2.2. 9-Bit UART

9-bit UART mode uses a total of eleven bits per data byte: a start bit, 8 data bits (LSB first), a programmable ninth data bit, and a stop bit. The state of the ninth transmit data bit is determined by the value in TB80 (SCON0.3), which is assigned by user software. It can be assigned the value of the parity flag (bit P in register PSW) for error detection, or used in multiprocessor communications. On receive, the ninth data bit goes into RB80 (SCON0.2) and the stop bit is ignored.

Data transmission begins when an instruction writes a data byte to the SBUF0 register. The TI0 Transmit Interrupt Flag (SCON0.1) is set at the end of the transmission (the beginning of the stop-bit time). Data reception can begin any time after the REN0 Receive Enable bit (SCON0.4) is set to 1. After the stop bit is received, the data byte will be loaded into the SBUF0 receive register if the following conditions are met: (1) RI0 must be logic 0, and (2) if MCE0 is logic 1, the 9th bit must be logic 1 (when MCE0 is logic 0, the state of the ninth data bit is unimportant). If these conditions are met, the eight bits of data are stored in SBUF0, the ninth bit is stored in RB80, and the RI0 flag is set to 1. If the above conditions are not met, SBUF0 and RB80 will not be loaded and the RI0 flag will not be set to 1. A UART0 interrupt will occur if enabled when either TI0 or RI0 is set to 1.

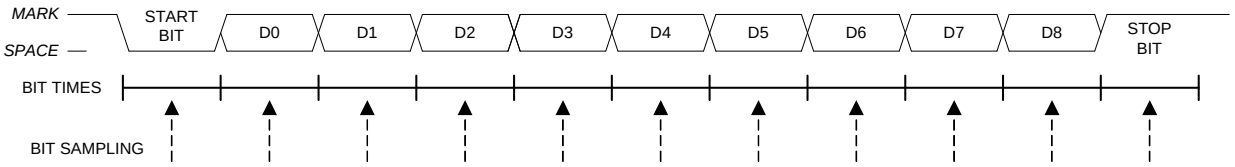


Figure 27.5. 9-Bit UART Timing Diagram

SFR Definition 28.9. TMR2RLL: Timer 2 Reload Register Low Byte

Bit	7	6	5	4	3	2	1	0
Name	TMR2RLL[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCA

Bit	Name	Function
7:0	TMR2RLL[7:0]	**Timer 2 Reload Register Low Byte.** TMR2RLL holds the low byte of the reload value for Timer 2.

SFR Definition 28.10. TMR2RLH: Timer 2 Reload Register High Byte

Bit	7	6	5	4	3	2	1	0
Name	TMR2RLH[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCB

Bit	Name	Function
7:0	TMR2RLH[7:0]	**Timer 2 Reload Register High Byte.** TMR2RLH holds the high byte of the reload value for Timer 2.

29.3.1. Edge-Triggered Capture Mode

In this mode, a valid transition on the CEXn pin causes the PCA to capture the value of the PCA counter/timer and load it into the corresponding module's 16-bit capture/compare register (PCA0CPLn and PCA0CPHn). The CAPPn and CAPNn bits in the PCA0CPMn register are used to select the type of transition that triggers the capture: low-to-high transition (positive edge), high-to-low transition (negative edge), or either transition (positive or negative edge). When a capture 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. If both CAPPn and CAPNn bits are set to logic 1, then the state of the Port pin associated with CEXn can be read directly to determine whether a rising-edge or falling-edge caused the capture.

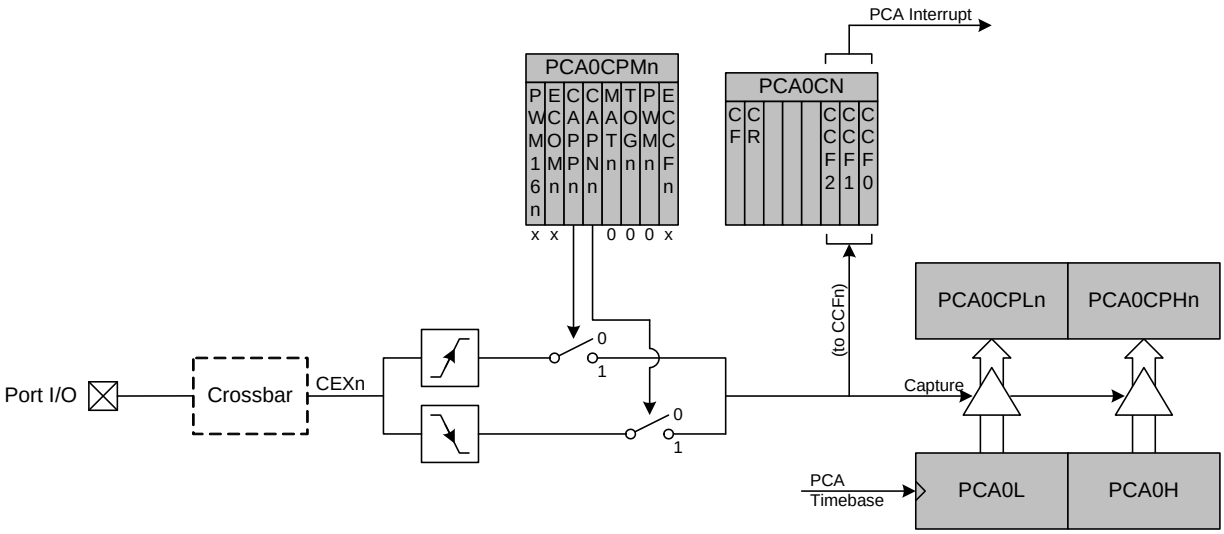


Figure 29.4. PCA Capture Mode Diagram

Note: The CEXn input signal must remain high or low for at least 2 system clock cycles to be recognized by the hardware.

29.3.6. 16-Bit Pulse Width Modulator Mode

A PCA module may be operated in 16-Bit PWM mode. 16-bit PWM mode is independent of the other (8-bit through 15-bit) PWM modes. In this mode, the 16-bit capture/compare module defines the number of PCA clocks for the low time of the PWM signal. When the PCA counter matches the module contents, the output on CEXn is asserted high; when the 16-bit counter overflows, CEXn is asserted low. 16-Bit PWM Mode is enabled by setting the ECOMn, PWMn, and PWM16n bits in the PCA0CPMn register.

The duty cycle of the PWM output signal can be varied by writing to an “Auto-Reload” Register, which is dual-mapped into the PCA0CPHn and PCA0CPLn register locations. The auto-reload registers are accessed (read or written) when the bit ARSEL in PCA0PWM is set to 1. The capture/compare registers are accessed when ARSEL is set to 0. This synchronous update feature allows software to asynchronously write a new PWM high time, which will then take effect on the following PWM period.

For backwards-compatibility with the 16-bit PWM mode available on other devices, the PWM duty cycle can also be changed without using the “Auto-Reload” register. To output a varying duty cycle without using the “Auto-Reload” register, new value writes should be synchronized with PCA CCFn match interrupts. Match interrupts should be enabled (ECCFn = 1 AND MATn = 1) to help synchronize the capture/compare register writes. If the MATn bit is set to 1, the CCFn flag for the module will be set each time a 16-bit comparator match (rising edge) occurs. The CF flag in PCA0CN can be used to detect the overflow (falling edge). The duty cycle for 16-Bit PWM Mode is given by Equation 29.4.

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.

$$\text{Duty Cycle} = \frac{(65536 - PCA0CPn)}{65536}$$

Equation 29.4. 16-Bit PWM Duty Cycle

Using Equation 29.4, the largest duty cycle is 100% (PCA0CPn = 0), and the smallest duty cycle is 0.0015% (PCA0CPn = 0xFFFF). A 0% duty cycle may be generated by clearing the ECOMn bit to 0.

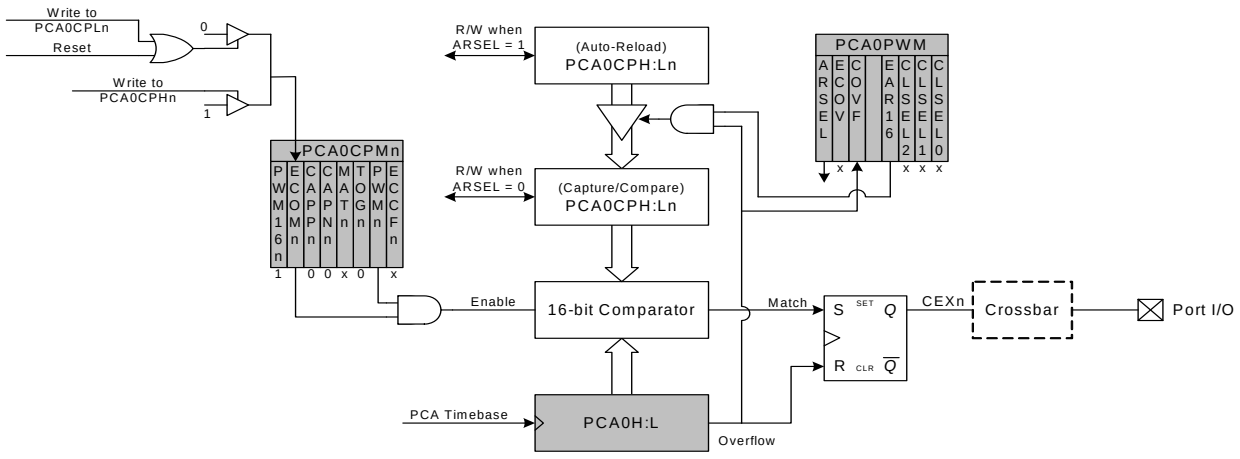


Figure 29.10. PCA 16-Bit PWM Mode

30. C2 Interface

C8051F80x-83x devices include an on-chip Silicon Labs 2-Wire (C2) debug interface to allow Flash programming and in-system debugging with the production part installed in the end application. The C2 interface operates using only two pins: a bi-directional data signal (C2D), and a clock input (C2CK). See the C2 Interface Specification for details on the C2 protocol.

30.1. C2 Interface Registers

The following describes the C2 registers necessary to perform Flash programming functions through the C2 interface. All C2 registers are accessed through the C2 interface as described in the C2 Interface Specification.

C2 Register Definition 30.1. C2ADD: C2 Address

Bit	7	6	5	4	3	2	1	0
Name	C2ADD[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function		
7:0	C2ADD[7:0]	**C2 Address.** The C2ADD register is accessed via the C2 interface to select the target Data register for C2 Data Read and Data Write commands.		
		Address	**Name**	**Description**
		0x00	DEVICEID	Selects the Device ID Register (read only)
		0x01	REVID	Selects the Revision ID Register (read only)
		0x02	FPCTL	Selects the C2 Flash Programming Control Register
		0xBF	FPDAT	Selects the C2 Flash Data Register
		0xD2	CRC0AUTO*	Selects the CRC0AUTO Register
		0xD3	CRC0CNT*	Selects the CRC0CNT Register
		0xCE	CRC0CN*	Selects the CRC0CN Register
		0xDE	CRC0DATA*	Selects the CRC0DATA Register
		0xCF	CRC0FLIP*	Selects the CRC0FLIP Register
		0xDD	CRC0IN*	Selects the CRC0IN Register
***Note:** CRC registers and functions are described in Section “24. Cyclic Redundancy Check Unit (CRC0)” on page 159.				