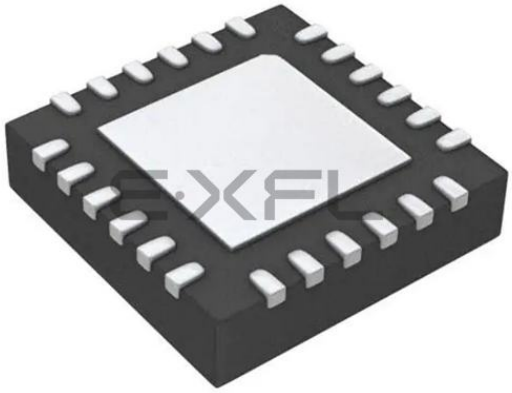

Welcome to E-XFL.COM

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	50MHz
Connectivity	SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	21
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°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/c8051f391-a-gm

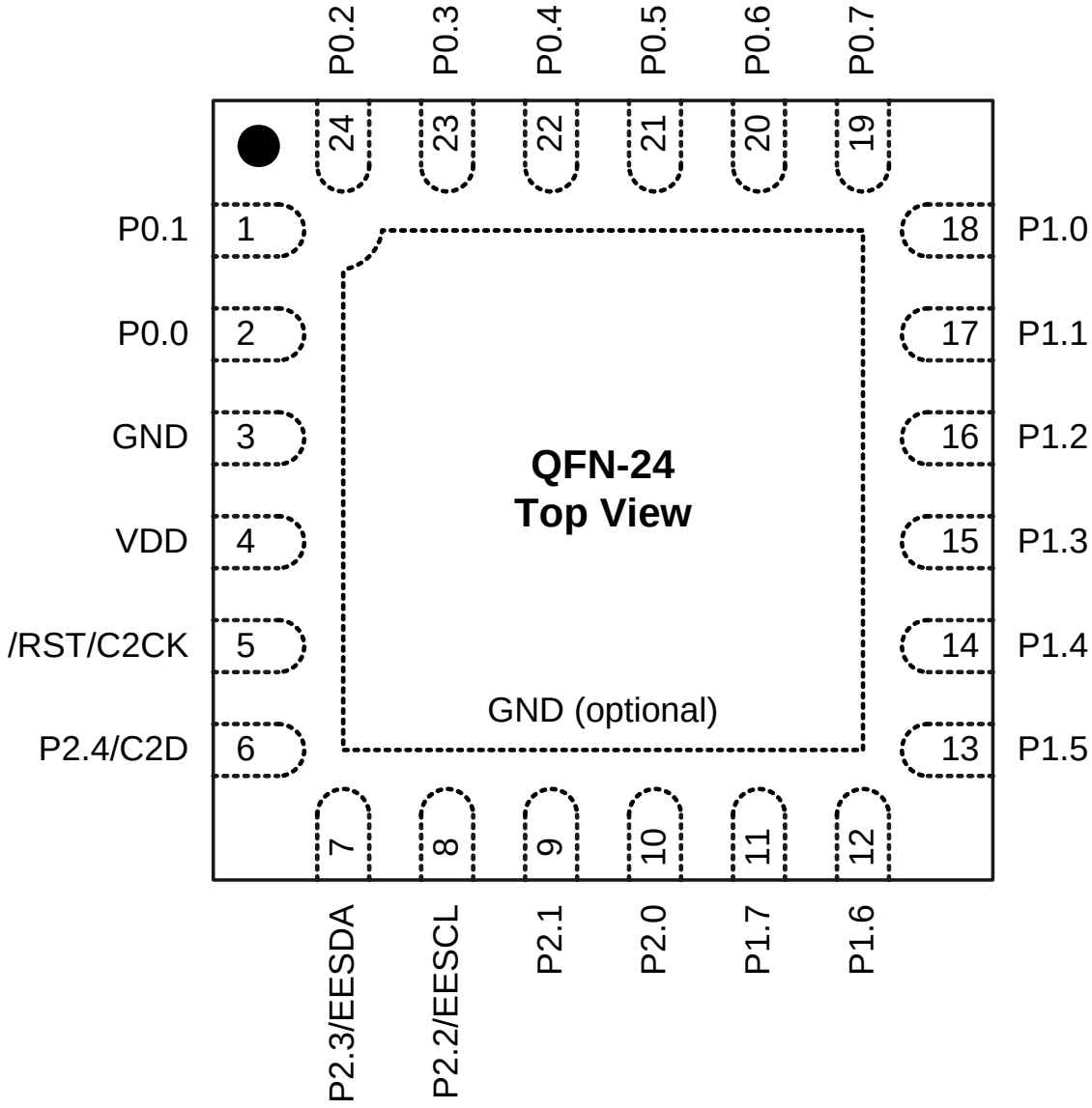

Figure 4.3. C8051F370/1/4/5 Pinout Diagram (Top View)

Table 7.10. ADC0 Electrical Characteristics

V_{DD} = 3.0 V, VREF = 2.40 V (REFSL = 0), –40 to +105 °C (C8051F39x), –40 to +85 °C (C8051F37x), unless otherwise specified.

Parameter	Test Condition	Min	Typ	Max	Unit
DC Accuracy					
Resolution	C8051F394/5/6/7, C8051F374/5		10		bits
Integral Nonlinearity	C8051F398/9	—	<±0.5	±2.0	LSB
Differential Nonlinearity		—	<±0.5	±1	LSB
Offset Error		–2	0	2	LSB
Full Scale Error		–5	–2	1	LSB
Offset Temperature Coefficient		—	0.005	—	LSB/°C
Dynamic performance (10 kHz sine-wave single-ended input, 1 dB below Full Scale, 500 ksps)					
Signal-to-Noise Plus Distortion		55	58	—	dB
Total Harmonic Distortion	Up to the 5th harmonic	—	–73	—	dB
Spurious-Free Dynamic Range		—	68	—	dB
Conversion Rate					
SAR Conversion Clock		—	—	8.33	MHz
Conversion Time in SAR Clocks		13	—	—	clocks
Track/Hold Acquisition Time		300	—	—	ns
Throughput Rate		—	—	500	ksps
Analog Inputs					
ADC Input Voltage Range	Single Ended (AIN+ – GND) Differential (AIN+ – AIN–)	0 –VREF	— —	VREF VREF	V V
Absolute Pin Voltage with respect to GND	Single Ended or Differential	0	—	V_{DD}	V
Sampling Capacitance (C_{SAMPLE})		—	5	—	pF
Input Multiplexer Impedance (R_{MUX})		—	1.6	—	kΩ
Power Specifications					
Power Supply Current (V_{DD} supplied to ADC0)	Operating Mode, 500 ksps	—	860	1010	µA
Power Supply Rejection	Single Ended (AIN+ – GND) Differential (AIN+ – AIN–)	— —	1.15 2.45	— —	mV/V mV/V

Table 7.11. ADC Temperature Sensor Electrical Characteristics

V_{DD} = 3.0 V, –40 to +105 °C (C8051F39x), –40 to +85 °C (C8051F37x), unless otherwise specified.

Parameter	Test Condition	Min	Typ	Max	Unit
Linearity		—	0.75	—	°C
Slope		—	2.92	—	mV/°C
Slope Error*		—	70	—	µV/°C
Offset	Temp = 0 °C	—	785	—	mV
Offset Error*	Temp = 0 °C	—	13	—	mV
Supply Current		—	90	120	µA
Note: Represents one standard deviation from the mean.					

Table 7.12. Precision Temperature Sensor Electrical Characteristics

V_{DD} = 1.8 to 3.6 V, –40 to +105 °C (C8051F39x), –40 to +85 °C (C8051F37x), unless otherwise specified.

Parameter	Test Condition	Min	Typ	Max	Unit
Range		–40	—	105	°C
Absolute Error		–2	0.2	+2	°C
Integral Nonlinearity		—	0	±0.4	°C
Resolution			0.0078125		°C
Power Supply Rejection		—	0.05	0.2	°C/V
Supply Current		—	230	280	µA
Clock Frequency (F_{TS0})		320	520	730	kHz

9.2. 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.

9.2.1. 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 “31. Timers” on page 242 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 “27. Port Input/Output” on page 173 for details on Port I/O configuration.

10.1. Calibration

The uncalibrated temperature sensor output is extremely linear and suitable for relative temperature measurements (see Table 7.11 on page 40 for specifications). For absolute temperature measurements, offset and/or gain calibration is recommended.

Figure 10.2 shows the typical temperature sensor error assuming a 1-point calibration at 0 °C. **Parameters that affect ADC measurement, in particular the voltage reference value, will also affect temperature measurement.**

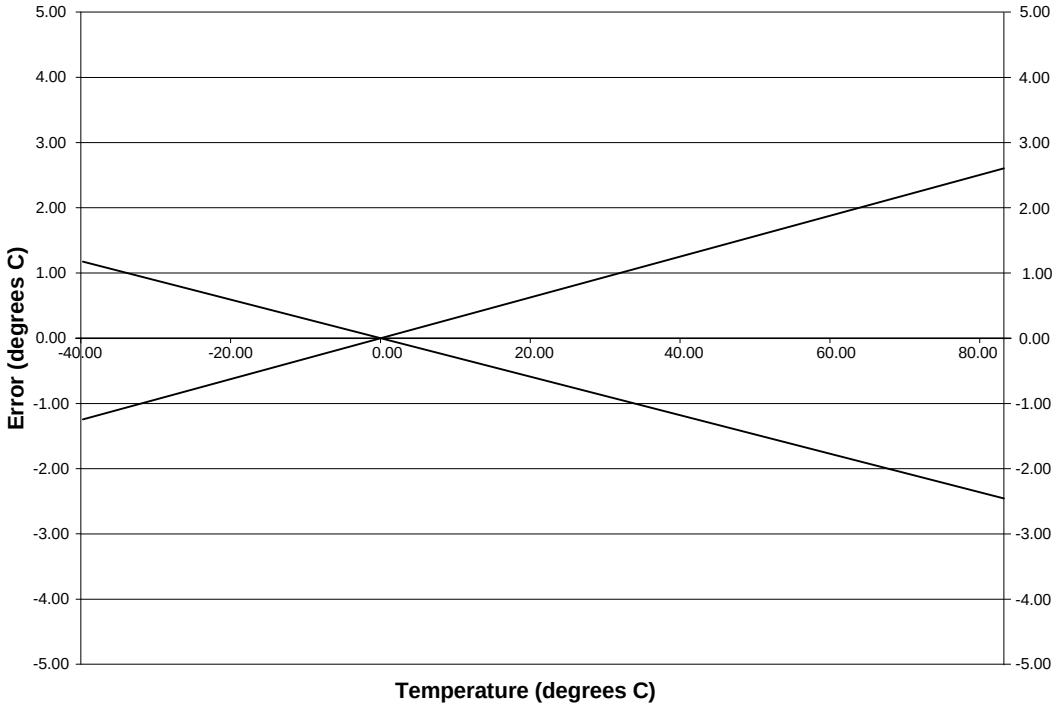

Figure 10.2. Temperature Sensor Error with 1-Point Calibration at 0 °C

14. Comparator0

C8051F39x/37x devices include an on-chip programmable voltage comparator, Comparator0, shown in Figure 14.1.

The Comparator offers programmable response time and hysteresis, an analog input multiplexer, and two outputs that are optionally available at the Port pins: a synchronous “latched” output (CP0), or an asynchronous “raw” output (CP0A). The asynchronous CP0A signal is available even when the system clock is not active. This allows the Comparator to operate and generate an output with the device in STOP mode. When assigned to a Port pin, the Comparator output may be configured as open drain or push-pull (see Section “27.4. Port I/O Initialization” on page 180). Comparator0 may also be used as a reset source (see Section “24.5. Comparator0 Reset” on page 159), or as a trigger to kill a PCA output channel.

The Comparator0 inputs are selected by the comparator input multiplexer, as detailed in Section “14.1. Comparator Multiplexer” on page 80.

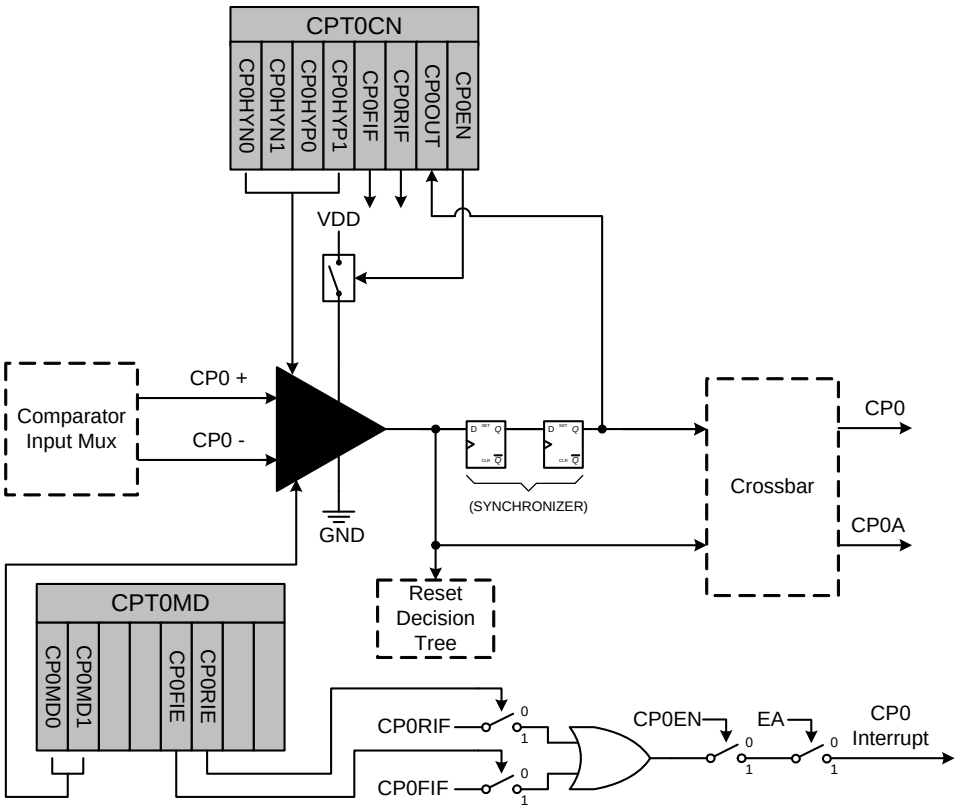

Figure 14.1. Comparator0 Functional Block Diagram

The Comparator output can be polled in software, used as an interrupt source, and/or routed to a Port pin. When routed to a Port pin, the Comparator output is available asynchronous or synchronous to the system clock; the asynchronous output is available even in STOP mode (with no system clock active). When disabled, the Comparator output (if assigned to a Port I/O pin via the Crossbar) defaults to the logic low state, and the power supply to the comparator is turned off. See Section “27.3. Priority Crossbar Decoder” on page 178 for details on configuring Comparator outputs via the digital Crossbar. Comparator inputs can be externally driven from –0.25 V to (V_{DD}) + 0.25 V without damage or upset. The complete Comparator electrical specifications are given in Section “7. Electrical Characteristics” on page 32.

Table 15.1. CIP-51 Instruction Set Summary (Continued)

Mnemonic	Description	Bytes	Clock Cycles
ANL C, bit	AND direct bit to Carry	2	2
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	2/4
JNC rel	Jump if Carry is not set	2	2/4
JB bit, rel	Jump if direct bit is set	3	3/5
JNB bit, rel	Jump if direct bit is not set	3	3/5
JBC bit, rel	Jump if direct bit is set and clear bit	3	3/5
Program Branching			
ACALL addr11	Absolute subroutine call	2	4*
LCALL addr16	Long subroutine call	3	5*
RET	Return from subroutine	1	6*
RETI	Return from interrupt	1	6*
AJMP addr11	Absolute jump	2	4*
LJMP addr16	Long jump	3	5*
SJMP rel	Short jump (relative address)	2	4*
JMP @A+DPTR	Jump indirect relative to DPTR	1	4*
JZ rel	Jump if A equals zero	2	2/4*
JNZ rel	Jump if A does not equal zero	2	2/4*
CJNE A, direct, rel	Compare direct byte to A and jump if not equal	3	4/6*
CJNE A, #data, rel	Compare immediate to A and jump if not equal	3	3/5*
CJNE Rn, #data, rel	Compare immediate to Register and jump if not equal	3	3/5*
CJNE @Ri, #data, rel	Compare immediate to indirect and jump if not equal	3	4/6*
DJNZ Rn, rel	Decrement Register and jump if not zero	2	2/4*
DJNZ direct, rel	Decrement direct byte and jump if not zero	3	3/5*
NOP	No operation	1	1
* Clock cycles for branch instructions with prefetch enabled, Align = 0, FLRT = 0			

24.2. 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 24.2). When V_{DD} returns to a level above V_{RST}, the CIP-51 will be released from the reset state. Note that 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 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 24.2 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 32 for complete electrical characteristics of the V_{DD} monitor.

26.4. Internal Low-Power Oscillator

All C8051F39x/37x devices include a low-power internal oscillator with a nominal frequency of 20 MHz. The low-power oscillator is automatically enabled when selected as the system clock and disabled when not in use. See Table 7.9, “Internal Low-Power Oscillator Electrical Characteristics,” on page 38 for complete oscillator specifications.

26.5. External Oscillator Drive Circuit

The external oscillator circuit may drive an external crystal, ceramic resonator, capacitor, or RC network. A CMOS clock may also provide a clock input. Figure 26.1 shows a block diagram of the four external oscillator options. The external oscillator is enabled and configured using the OSCXCN register (see SFR Definition 26.5).

Important Note on External Oscillator Usage: Port pins must be configured when using the external oscillator circuit. When the external oscillator drive circuit is enabled in crystal/resonator mode, Port pins P0.2 and P0.3 are used as XTAL1 and XTAL2, respectively. When the external oscillator drive circuit is enabled in capacitor, RC, or CMOS clock mode, Port pin P0.3 is used as XTAL2. The Port I/O Crossbar should be configured to skip the Port pin used by the oscillator circuit; see Section “27.3. Priority Crossbar Decoder” on page 178 for Crossbar configuration. Additionally, when using the external oscillator circuit in crystal/resonator, capacitor, or RC mode, the associated Port pins should be configured as **analog inputs**. In CMOS clock mode, the associated pin should be configured as a **digital input**. See Section “27.4. Port I/O Initialization” on page 180 for details on Port input mode selection.

The external oscillator output may be selected as the system clock or used to clock some of the digital peripherals (e.g. Timers, PCA, etc.). See the data sheet chapters for each digital peripheral for details. See Section “7. Electrical Characteristics” on page 32 for complete oscillator specifications.

26.5.1. External Crystal Mode

If a crystal or ceramic resonator is used as the external oscillator, the crystal/resonator and a 10 MΩ resistor must be wired across the XTAL1 and XTAL2 pins as shown in Figure 26.1, “Crystal Mode”. Appropriate loading capacitors should be added to XTAL1 and XTAL2, and both pins should be configured for analog I/O with the digital output drivers disabled.

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 recommended load capacitance depends upon the crystal and the manufacturer. Refer to the crystal data sheet when completing these calculations.

The equation for determining the load capacitance for two capacitors is

$$C_L = \frac{C_A \times C_B}{C_A + C_B} + C_S$$

Where:
C_A and C_B are the capacitors connected to the crystal leads.
C_S is the total stray capacitance of the PCB.
The stray capacitance for a typical layout where the crystal is as close as possible to the pins is 2-5 pF per pin.

If C_A and C_B are the same (C), then the equation becomes

$$C_L = \frac{C}{2} + C_S$$

26.5.2. External RC Example

If an RC network is used as an external oscillator source for the MCU, the circuit should be configured as shown in Figure 26.1, “RC Mode”. The capacitor should be no greater than 100 pF; however, for very small capacitors, the total capacitance may be dominated by parasitic capacitance in the PCB layout. To determine the required External Oscillator Frequency Control value (XFCN) in the OSCXCN Register, first select the RC network value to produce the desired frequency of oscillation, according to Equation , where f = the frequency of oscillation in MHz, C = the capacitor value in pF, and R = the pull-up resistor value in kΩ.

$$f = 1.23 \times 10^3 / (R \times C)$$

Equation 26.1. RC Mode Oscillator Frequency

For example: If the frequency desired is 100 kHz, let R = 246 kΩ and C = 50 pF:

f = 1.23(10^3) / RC = 1.23 (10^3) / [246 x 50] = 0.1 MHz = 100 kHz

Referring to the table in SFR Definition 26.5, the required XFCN setting is 010b.

26.5.3. External Capacitor Example

If a capacitor is used as an external oscillator for the MCU, the circuit should be configured as shown in Figure 26.1, “C Mode”. The capacitor should be no greater than 100 pF; however, for very small capacitors, the total capacitance may be dominated by parasitic capacitance in the PCB layout. To determine the required External Oscillator Frequency Control value (XFCN) in the OSCXCN Register, select the capacitor to be used and find the frequency of oscillation according to Equation , where f = the frequency of oscillation in MHz, C = the capacitor value in pF, and V_{DD} = the MCU power supply in Volts.

$$f = (KF) / (C \times V_{DD})$$

Equation 26.2. C Mode Oscillator Frequency

For example: Assume V_{DD} = 3.0 V and f = 150 kHz:

f = KF / (C x VDD)
0.150 MHz = KF / (C x 3.0)

Since the frequency of roughly 150 kHz is desired, select the K Factor from the table in SFR Definition 26.5 (OSCXCN) as KF = 22:

0.150 MHz = 22 / (C x 3.0)
C x 3.0 = 22 / 0.150 MHz
C = 146.6 / 3.0 pF = 48.8 pF

Therefore, the XFCN value to use in this example is 011b and C = 50 pF.

SFR Definition 27.17. P2MDOUT: Port 2 Output Mode

Bit	7	6	5	4	3	2	1	0
Name				P2MDOUT[4:0]				
Type	R	R	R	R/W				
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xA6; SFR Page = All Pages

Bit	Name	Function
7:5	Unused	Read = 000b; Write = Don't Care
4:0	P2MDOUT[4:0]	**Output Configuration Bits for P2.4–P2.0 (respectively).** These bits are ignored if the corresponding bit in register P2MDIN is logic 0. 0: Corresponding P2.n Output is open-drain. 1: Corresponding P2.n Output is push-pull.
Note: P2.0 is not available for analog input in the QFN20-packaged devices, and P2.1-P2.4 are only available in the QFN24-packaged devices.		

SFR Definition 27.18. P2SKIP: Port 2 Skip

Bit	7	6	5	4	3	2	1	0
Name					P2SKIP[7:0]			
Type	R	R	R	R	R/W			
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD6; SFR Page = All Pages

Bit	Name	Function
7:4	Unused	Read = 0000b; Write = Don't Care
3:0	P2SKIP[3:0]	**Port 2 Crossbar Skip Enable Bits.** These bits select Port 2 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 P2.n pin is not skipped by the Crossbar. 1: Corresponding P2.n pin is skipped by the Crossbar.
Note: P2.0 is not available for crossbar peripherals in the QFN20-packaged devices, and P2.1-P2.4 are only available in the QFN24-packaged devices.		

Table 28.3. Sources for Hardware Changes to SMBnCN

Bit	Set by Hardware When:	Cleared by Hardware When:
MASTERn	■ A START is generated.	■ A STOP is generated. ■ Arbitration is lost.
TXMODEn	■ START is generated. ■ SMBnDAT is written before the start of an SMBus frame.	■ A START is detected. ■ Arbitration is lost. ■ 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. ■ 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). ■ SCLn is sensed low while attempting to generate a STOP or repeated START condition. ■ 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. ■ 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.

28.4.5. 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 28.4.4.2.

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

the incoming slave address. Additionally, if the GCn bit in register SMBnADR is set to 1, hardware will recognize the General Call Address (0x00). Table 28.4 shows some example parameter settings and the slave addresses that will be recognized by hardware under those conditions.

Table 28.4. Hardware Address Recognition Examples (EHACK = 1)

Hardware Slave Address SLVn[6:0]	Slave Address Mask SLVMn[6:0]	GCn bit	Slave Addresses Recognized by Hardware
0x34	0x7F	0	0x34
0x34	0x7F	1	0x34, 0x00 (General Call)
0x34	0x7E	0	0x34, 0x35
0x34	0x7E	1	0x34, 0x35, 0x00 (General Call)
0x70	0x73	0	0x70, 0x74, 0x78, 0x7C

SFR Definition 28.6. SMB0ADR: SMBus0 Slave Address

Bit	7	6	5	4	3	2	1	0
Name	SLV0[6:0]							GC0
Type	R/W							R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD7; SFR Page = 0

Bit	Name	Function
7:1	SLV0[6:0]	**SMBus Hardware Slave Address.** Defines the SMBus0 Slave Address(es) for automatic hardware acknowledgment. Only address bits which have a 1 in the corresponding bit position in SLVM0[6:0] are checked against the incoming address. This allows multiple addresses to be recognized.
0	GC0	**General Call Address Enable.** When hardware address recognition is enabled (EHACK0 = 1), this bit will determine whether the General Call Address (0x00) is also recognized by hardware. 0: General Call Address is ignored. 1: General Call Address is recognized.

Table 28.5. SMBus Status Decoding: Hardware ACK Disabled (EHACK = 0)

Mode	Values Read				Current SMbus State	Typical Response Options	Values to Write			Next Status Vector Expected
	Status Vector	ACKRQ	ARBLOST	ACK			STA	STO	ACK	
Master Transmitter	1110	0	0	X	A master START was generated.	Load slave address + R/W into SMB0DAT.	0	0	X	1100
	1100	0	0	0	A master data or address byte was transmitted; NACK received.	Set STA to restart transfer.	1	0	X	1110
						Abort transfer.	0	1	X	—
		0	0	1	A master data or address byte was transmitted; ACK received.	Load next data byte into SMB0DAT.	0	0	X	1100
						End transfer with STOP.	0	1	X	—
						End transfer with STOP and start another transfer.	1	1	X	—
						Send repeated START.	1	0	X	1110
						Switch to Master Receiver Mode (clear SI without writing new data to SMB0DAT).	0	0	X	1000
Master Receiver	1000	1	0	X	A master data byte was received; ACK requested.	Acknowledge received byte; Read SMB0DAT.	0	0	1	1000
						Send NACK to indicate last byte, and send STOP.	0	1	0	—
						Send NACK to indicate last byte, and send STOP followed by START.	1	1	0	1110
						Send ACK followed by repeated START.	1	0	1	1110
						Send NACK to indicate last byte, and send repeated START.	1	0	0	1110
						Send ACK and switch to Master Transmitter Mode (write to SMB0DAT before clearing SI).	0	0	1	1100
						Send NACK and switch to Master Transmitter Mode (write to SMB0DAT before clearing SI).	0	0	0	1100

30. Enhanced Serial Peripheral Interface (SPI0)

The Enhanced Serial Peripheral Interface (SPI0) provides access to a flexible, full-duplex synchronous serial bus. SPI0 can operate as a master or slave device in both 3-wire or 4-wire modes, and supports multiple masters and slaves on a single SPI bus. The slave-select (NSS) signal can be configured as an input to select SPI0 in slave mode, or to disable Master Mode operation in a multi-master environment, avoiding contention on the SPI bus when more than one master attempts simultaneous data transfers. NSS can also be configured as a chip-select output in master mode, or disabled for 3-wire operation. Additional general purpose port I/O pins can be used to select multiple slave devices in master mode.

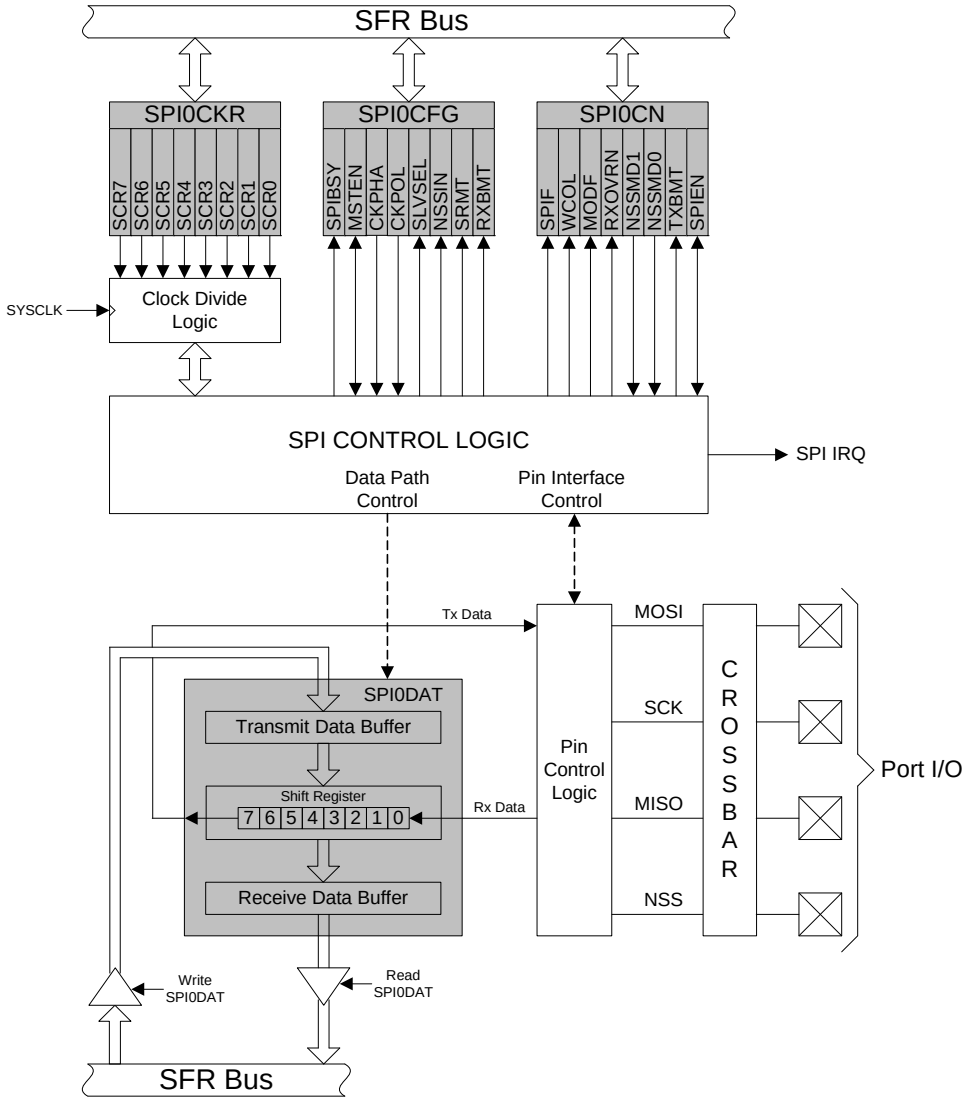

Figure 30.1. SPI Block Diagram

31.1.3. Mode 2: 8-bit Counter/Timer with Auto-Reload

Mode 2 configures Timer 0 and Timer 1 to operate as 8-bit counter/timers with automatic reload of the start value. TL0 holds the count and TH0 holds the reload value. When the counter in TL0 overflows from all ones to 0x00, the timer overflow flag TF0 in the TCON register is set and the counter in TL0 is reloaded from TH0. If Timer 0 interrupts are enabled, an interrupt will occur when the TF0 flag is set. The reload value in TH0 is not changed. TL0 must be initialized to the desired value before enabling the timer for the first count to be correct. When in Mode 2, Timer 1 operates identically to Timer 0.

Both counter/timers are enabled and configured in Mode 2 in the same manner as Mode 0. Setting the TR0 bit (TCON.4) enables the timer when either GATE0 in the TMOD register is logic 0 or when the input signal INT0 is active as defined by bit IN0PL in register IT01CF (see Section “20.3. External Interrupts INT0 and INT1” on page 128 for details on the external input signals INT0 and INT1).

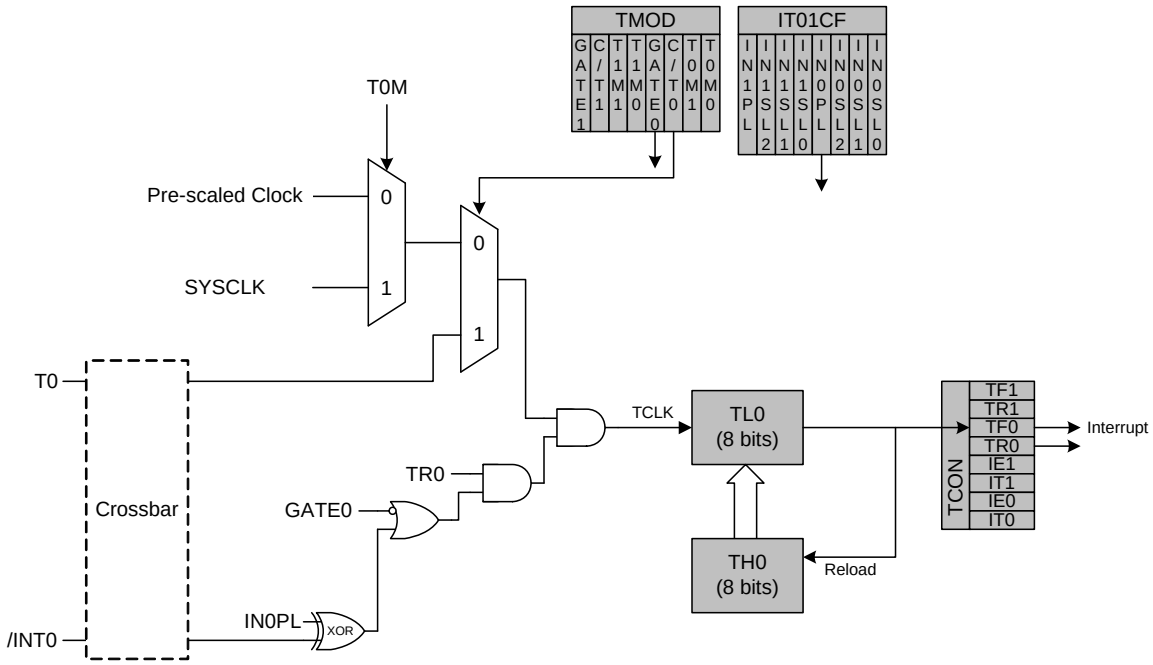

Figure 31.2. T0 Mode 2 Block Diagram

SFR Definition 31.5. TL0: Timer 0 Low Byte

Bit	7	6	5	4	3	2	1	0
Name	TL0[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8A; SFR Page = All Pages

Bit	Name	Function
7:0	TL0[7:0]	**Timer 0 Low Byte.** The TL0 register is the low byte of the 16-bit Timer 0.

SFR Definition 31.6. TL1: Timer 1 Low Byte

Bit	7	6	5	4	3	2	1	0
Name	TL1[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8B; SFR Page = All Pages

Bit	Name	Function
7:0	TL1[7:0]	**Timer 1 Low Byte.** The TL1 register is the low byte of the 16-bit Timer 1.

31.3.2. 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 31.8. 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 the internal Low-frequency Oscillator. 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	External Clock / 8
0	10	Reserved
0	11	Internal LFO
1	X	SYSCLK

T3ML	T3XCLK[1:0]	TMR3L Clock Source
0	00	SYSCLK / 12
0	01	External Clock / 8
0	10	Reserved
0	11	Internal LFO
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.

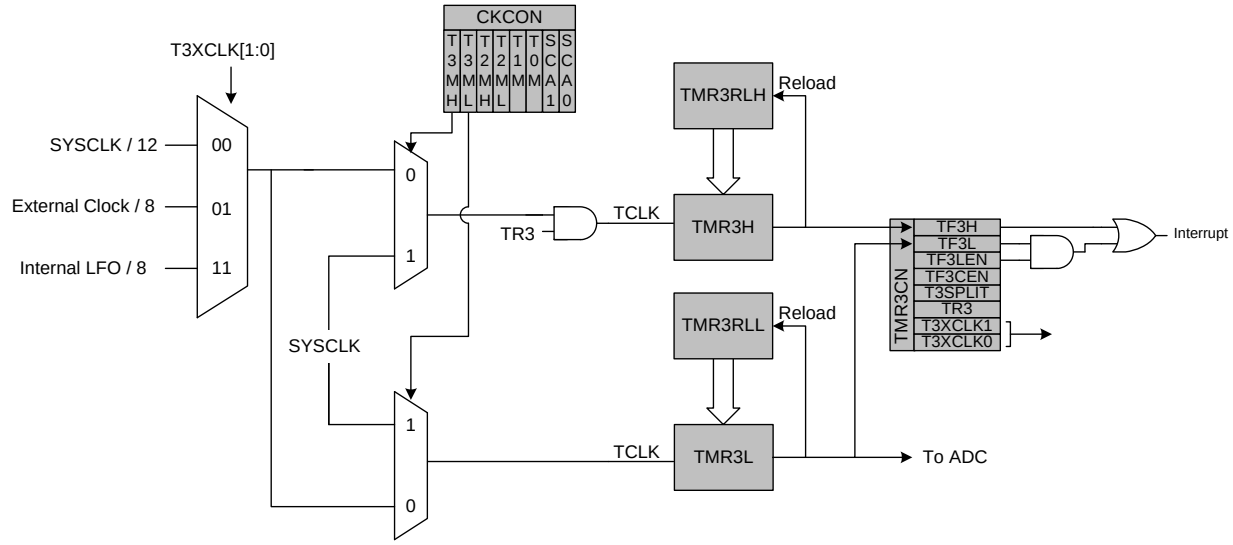

Figure 31.8. Timer 3 8-Bit Mode Block Diagram

SFR Definition 31.14. TMR3CN: Timer 3 Control

Bit	7	6	5	4	3	2	1	0
Name	TF3H	TF3L	TF3LEN	TF3CEN	T3SPLIT	TR3	T3XCLK[1:0]	
Type	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 = 0x91; SFR Page = 0

Bit	Name	Function
7	TF3H	**Timer 3 High Byte Overflow Flag.** Set by hardware when the Timer 3 high byte overflows from 0xFF to 0x00. In 16 bit mode, this will occur when Timer 3 overflows from 0xFFFF to 0x0000. When the Timer 3 interrupt is enabled, setting this bit causes the CPU to vector to the Timer 3 interrupt service routine. This bit is not automatically cleared by hardware.
6	TF3L	**Timer 3 Low Byte Overflow Flag.** Set by hardware when the Timer 3 low byte overflows from 0xFF to 0x00. TF3L will be set when the low byte overflows regardless of the Timer 3 mode. This bit is not automatically cleared by hardware.
5	TF3LEN	**Timer 3 Low Byte Interrupt Enable.** When set to 1, this bit enables Timer 3 Low Byte interrupts. If Timer 3 interrupts are also enabled, an interrupt will be generated when the low byte of Timer 3 overflows.
4	TF3CEN	**Timer 3 Low-Frequency Oscillator Capture Enable.** When set to 1, this bit enables Timer 3 Low-Frequency Oscillator Capture Mode. If TF3CEN is set and Timer 3 interrupts are enabled, an interrupt will be generated on a falling edge of the low-frequency oscillator output, and the current 16-bit timer value in TMR3H:TMR3L will be copied to TMR3RLH:TMR3RLL.
3	T3SPLIT	**Timer 3 Split Mode Enable.** When this bit is set, Timer 3 operates as two 8-bit timers with auto-reload. 0: Timer 3 operates in 16-bit auto-reload mode. 1: Timer 3 operates as two 8-bit auto-reload timers.
2	TR3	**Timer 3 Run Control.** Timer 3 is enabled by setting this bit to 1. In 8-bit mode, this bit enables/disables TMR3H only; TMR3L is always enabled in split mode.
1:0	T3XCLK[1:0]	**Timer 3 External Clock Select.** This bit selects the “external” clock source for Timer 3. If Timer 3 is in 8-bit mode, this bit selects the external oscillator clock source for both timer bytes. However, the Timer 3 Clock Select bits (T3MH and T3ML in register CKCON) may still be used to select between the external clock and the system clock for either timer. 00: System clock divided by 12. 01: External clock divided by 8 (synchronized with SYSCLK when not in suspend). 10: Reserved. 11: Internal LFO/8 (synchronized with SYSCLK when not in suspend).

C2 Register Definition 33.4. 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.** 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 33.5. 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.** 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