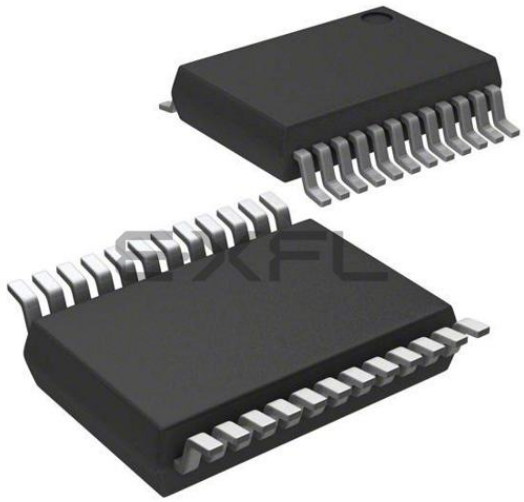

Welcome to E-XFL.COM

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	CIP-51 8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	17
Program Memory Size	4KB (4K 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	24-SSOP (0.154", 3.90mm Width)
Supplier Device Package	24-QSOP
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f989-c-gu

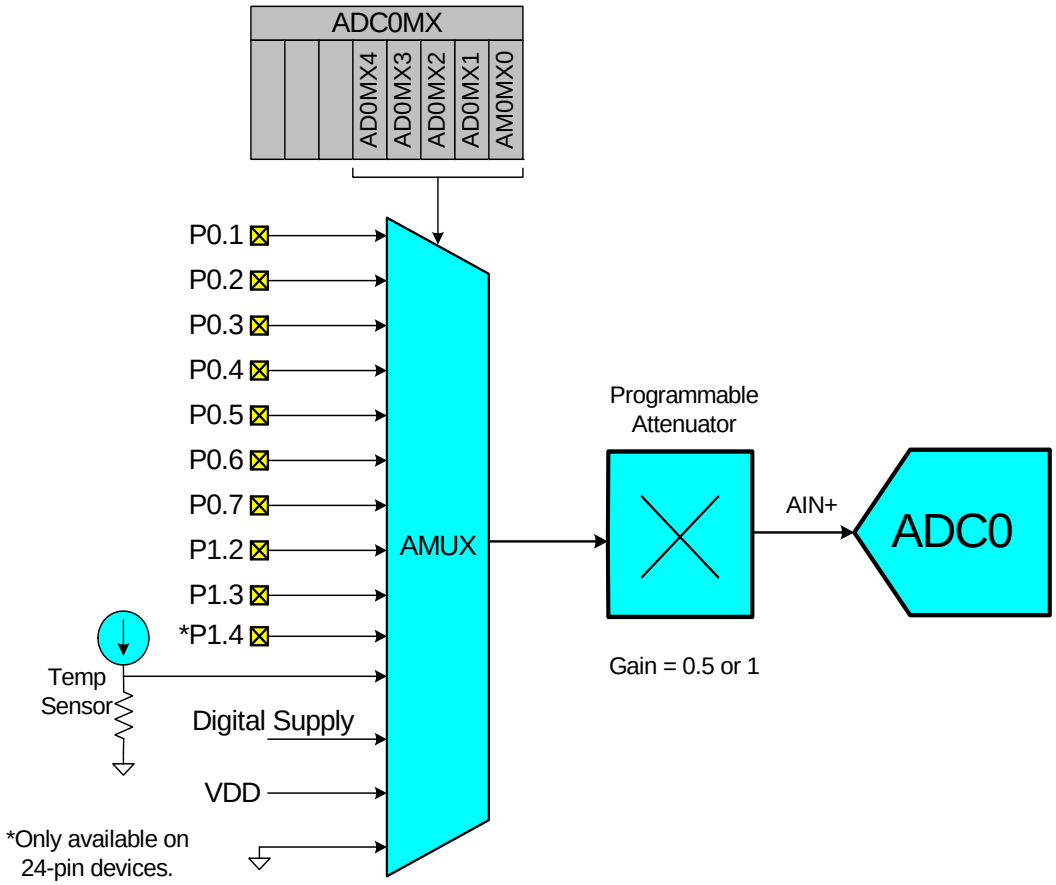

Figure 1.17. ADC0 Multiplexer Block Diagram

1.6. Programmable Current Reference (IREF0)

C8051F99x-C8051F98x devices include an on-chip programmable current reference (source or sink) with two output current settings: low power mode and high current mode. The maximum current output in low power mode is 63 µA (1 µA steps) and the maximum current output in high current mode is 504 µA (8 µA steps).

1.7. Comparator

C8051F99x-C8051F98x devices include an on-chip programmable voltage comparator: Comparator 0 (CPT0) which is shown in Figure 1.18.

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 when the device is in some low power modes.

The comparator inputs may be connected to Port I/O pins or to other internal signals. Port pins may also be used to directly sense capacitive touch switches.

Table 3.1. Pin Definitions for the C8051F99x-C8051F98x (Continued)

Name	Pin Numbers			Type	Description
	‘F980/1/2 ‘F983/5 ‘F990/1 -GM	‘F986/7 ‘F988/9 ‘F996/7 -GM	‘F986/7 ‘F988/9 ‘F996/7 -GU		
P0.6/ CNVSTR*	16	18	21	D I/O or A In D In	Port 0.6. See Section “21. Port Input/Output” on page 215 for a complete description. External Convert Start Input for ADC0. See Section “5.7. ADC0 Analog Multiplexer” on page 83 for a complete description.
P0.7/ IREF0	15	17	20	D I/O or A In A Out	Port 0.7. See Section “21. Port Input/Output” on page 215 for a complete description. IREF0 Output. See IREF Section for complete description.
P1.0 CP0+	14	16	19	D I/O or A In A In	Port 1.0. See Section “21. Port Input/Output” on page 215 for a complete description. May also be used as SCK for SPI1. Comparator0 positive input. See Comparator Section for complete description.
P1.1 CP0-	13	15	18	D I/O or A In A In	Port 1.1. See Section “21. Port Input/Output” on page 215 for a complete description. Comparator0 negative input. See Comparator Section for complete description.
P1.2	11	14	17	D I/O or A In	Port 1.2. See Section “21. Port Input/Output” on page 215 for a complete description.
P1.3	10	13	16	D I/O or A In	Port 1.3. See Section “21. Port Input/Output” on page 215 for a complete description.
P1.4	—	12	15	D I/O or A In	Port 1.4. See Section “21. Port Input/Output” on page 215 for a complete description.
P1.5	9	11	14	D I/O or A In	Port 1.5. See Section “21. Port Input/Output” on page 215 for a complete description.
***Note:** Available only on the C8051F980/2/6/8 and C8051F990/6 devices.					

Table 4.10. ADC0 Electrical Characteristics (Continued)

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

Parameter	Conditions	Min	Typ	Max	Units
Analog Inputs					
ADC Input Voltage Range	Single Ended (AIN+ – GND)	0	—	VREF	V
Absolute Pin Voltage with respect to GND	Single Ended	0	—	VDD	V
Sampling Capacitance	1x Gain 0.5x Gain	—	16 13	—	pF
Input Multiplexer Impedance		—	5	—	kΩ
Power Specifications					
Power Supply Current (V_{DD} supplied to ADC0)	Normal Power Mode: Conversion Mode (300 ksps) Tracking Mode (0 ksps) Low Power Mode: Conversion Mode (150 ksps) Tracking Mode (0 ksps)	 — — — —	 650 740 370 400	 — — — —	µA
Power Supply Rejection	Internal High Speed VREF External VREF	— —	67 74	— —	dB

1. INL and DNL specifications for 12-bit mode do not include the first or last four ADC codes.
2. The maximum code in 12-bit mode is 0xFFFC. The Full Scale Error is referenced from the maximum code.
3. Performance in 8-bit mode is similar to 10-bit mode.

Table 4.11. Temperature Sensor Electrical Characteristics

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

Parameter	Conditions	Min	Typ	Max	Units
Linearity		—	±1	—	°C
Slope		—	3.40	—	mV/°C
Slope Error*		—	40	—	µV/°C
Offset	Temp = 25 °C	—	1025	—	mV
Offset Error*	Temp = 25 °C	—	18	—	mV
Temperature Sensor Turn-On Time		—	1.7	—	µs
Supply Current		—	35	—	µA

***Note:** Represents one standard deviation from the mean.

SFR Definition 5.4. ADC0PWR: ADC0 Burst Mode Power-Up Time

Bit	7	6	5	4	3	2	1	0
Name	AD0LPM				AD0PWR[3:0]			
Type	R/W	R	R	R	R/W			
Reset	0	0	0	0	1	1	1	1

SFR Page = All; SFR Address = 0xBB

Bit	Name	Function
7	AD0LPM	**ADC0 Low Power Mode Enable.** Enables Low Power Mode Operation. 0: Low Power Mode disabled. 1: Low Power Mode enabled.
6:4	Unused	Read = 0000b; Write = Don't Care.
3:0	AD0PWR[3:0]	**ADC0 Burst Mode Power-Up Time.** Sets the time delay required for ADC0 to power up from a low power state. For BURSTEN = 0: ADC0 power state controlled by AD0EN. For BURSTEN = 1 and AD0EN = 1: ADC0 remains enabled and does not enter a low power state after all conversions are complete. Conversions can begin immediately following the start-of-conversion signal. For BURSTEN = 1 and AD0EN = 0: ADC0 enters a low power state after all conversions are complete. Conversions can begin a programmed delay after the start-of-conversion signal. The ADC0 Burst Mode Power-Up time is programmed according to the following equation: $AD0PWR = \frac{Tstartup}{400ns} - 1$ or $Tstartup = (AD0PWR + 1)400ns$ **Note:** Setting AD0PWR to 0x04 provides a typical tracking time of 2 us for the first sample taken after the start of conversion.

8.11. CS0 Conversion Accumulator

CS0 can be configured to accumulate multiple conversions on an input channel. The number of samples to be accumulated is configured using the CS0ACU2:0 bits (CS0CF2:0). The accumulator can accumulate 1, 4, 8, 16, 32, or 64 samples. After the defined number of samples have been accumulated, the result is divided by either 1, 4, 8, 16, 32, or 64 (depending on the CS0ACU[2:0] setting) and copied to the CS0DH:CS0DL SFRs.

Table 8.1. Operation with Auto-scan and Accumulate

Auto-Scan Enabled	Accumulator Enabled	CS0 Conversion Complete Interrupt Behavior	CS0 Greater Than Interrupt Behavior	CS0MX Behavior
N	N	CS0INT Interrupt serviced after 1 conversion completes	Interrupt serviced after 1 conversion completes if value in CS0DH:CS0DL is greater than CS0THH:CS0THL	CS0MX unchanged.
N	Y	CS0INT Interrupt serviced after *M* conversions complete	Interrupt serviced after *M* conversions complete if value in CS0DH:CS0DL (post accumulate and divide) is greater than CS0THH:CS0THL	CS0MX unchanged.
Y	N	CS0INT Interrupt serviced after 1 conversion completes	Interrupt serviced after conversion completes if value in CS0DH:CS0DL is greater than CS0THH:CS0THL; Auto-Scan stopped	If greater-than comparator detects conversion value is greater than CS0THH:CS0THL, CS0MX is left unchanged; otherwise, CS0MX updates to the next channel (CS0MX + 1) and wraps back to CS0SS after passing CS0SE.
Y	Y	CS0INT Interrupt serviced after *M* conversions complete	Interrupt serviced after *M* conversions complete if value in CS0DH:CS0DL (post accumulate and divide) is greater than CS0THH:CS0THL; Auto-Scan stopped	If greater-than comparator detects conversion value is greater than CS0THH:CS0THL, CS0MX is left unchanged; otherwise, CS0MX updates to the next channel (CS0MX + 1) and wraps back to CS0SS after passing CS0SE.

Note: M = Accumulator setting (1x, 4x, 8x, 16x, 32x, 64x).

SFR Definition 8.15. CS0MX: Capacitive Sense Mux Channel Select

Bit	7	6	5	4	3	2	1	0
Name	Reserved	Reserved	Reserved	Reserved	CS0MX[3:0]			
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xAB

Bit	Name	Description	
7:5	Reserved	Read = 0000b; Write = 0000b.	
4:0	CS0MX[4:0]	**CS0 Mux Channel Select.** Selects one of the 14 input channels for Capacitive Sense conversion.	
		Value	**Channel**
		0000	P0.0
		0001	P0.1
		0010	P0.2
		0011	P0.3
		0100	P0.4
		0101	P0.5
		0110	P0.6
		0111	P0.7
		1000	P1.0
		1001	P1.1
		1010	P1.2
		1011	P1.3
		1100	P1.4 (24-pin packages only)
		1101	P1.5
		1110	Reserved
		1111	Reserved

13.3. 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 13.1 on page 140 to determine the fixed priority order used to arbitrate between simultaneously recognized interrupts.

13.4. 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.

SFR Definition 13.1. IE: Interrupt Enable

Bit	7	6	5	4	3	2	1	0
Name	EA	ESPI0	ET2	ES0	ET1	EX1	ET0	EX0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Page = All; SFR Address = 0xA8; Bit-Addressable

Bit	Name	Function
7	EA	**Enable All Interrupts.** Globally enables/disables all interrupts. It overrides individual interrupt mask settings. 0: Disable all interrupt sources. 1: Enable each interrupt according to its individual mask setting.
6	ESPI0	**Enable Serial Peripheral Interface (SPI0) Interrupt.** This bit sets the masking of the SPI0 interrupts. 0: Disable all SPI0 interrupts. 1: Enable interrupt requests generated by SPI0.
5	ET2	**Enable Timer 2 Interrupt.** This bit sets the masking of the Timer 2 interrupt. 0: Disable Timer 2 interrupt. 1: Enable interrupt requests generated by the TF2L or TF2H flags.
4	ES0	**Enable UART0 Interrupt.** This bit sets the masking of the UART0 interrupt. 0: Disable UART0 interrupt. 1: Enable UART0 interrupt.
3	ET1	**Enable Timer 1 Interrupt.** This bit sets the masking of the Timer 1 interrupt. 0: Disable all Timer 1 interrupt. 1: Enable interrupt requests generated by the TF1 flag.
2	EX1	**Enable External Interrupt 1.** This bit sets the masking of External Interrupt 1. 0: Disable external interrupt 1. 1: Enable interrupt requests generated by the $\overline{\text{INT1}}$ input.
1	ET0	**Enable Timer 0 Interrupt.** This bit sets the masking of the Timer 0 interrupt. 0: Disable all Timer 0 interrupt. 1: Enable interrupt requests generated by the TF0 flag.
0	EX0	**Enable External Interrupt 0.** This bit sets the masking of External Interrupt 0. 0: Disable external interrupt 0. 1: Enable interrupt requests generated by the $\overline{\text{INT0}}$ input.

14. Flash Memory

On-chip, re-programmable Flash memory is included for program code and non-volatile data storage. The Flash memory can be programmed in-system through the C2 interface or by software using the MOVX write instruction. Once cleared to logic 0, a Flash bit must be erased to set it back to logic 1. Flash bytes would typically be erased (set to 0xFF) before being reprogrammed. The write and erase operations are automatically timed by hardware for proper execution; data polling to determine the end of the write/erase operations is not required. Code execution is stalled during Flash write/erase operations. Refer to Table 4.6 for complete Flash memory electrical characteristics.

14.1. Programming the Flash Memory

The simplest means of programming the Flash memory is through the C2 interface using programming tools provided by Silicon Laboratories or a third party vendor. This is the only means for programming a non-initialized device. For details on the C2 commands to program Flash memory, see Section “27. C2 Interface” on page 319.

The Flash memory can be programmed by software using the MOVX write instruction with the address and data byte to be programmed provided as normal operands. Before programming Flash memory using MOVX, Flash programming operations must be enabled by: (1) setting the PSWE Program Store Write Enable bit (PSCTL.0) to logic 1 (this directs the MOVX writes to target Flash memory); and (2) Writing the Flash key codes in sequence to the Flash Lock register (FLKEY). The PSWE bit remains set until cleared by software. For detailed guidelines on programming Flash from firmware, please see Section “14.5. Flash Write and Erase Guidelines” on page 156.

To ensure the integrity of the Flash contents, the on-chip VDD Monitor must be enabled and enabled as a reset source in any system that includes code that writes and/or erases Flash memory from software. Furthermore, there should be no delay between enabling the V_{DD} Monitor and enabling the V_{DD} Monitor as a reset source. Any attempt to write or erase Flash memory while the V_{DD} Monitor is disabled, or not enabled as a reset source, will cause a Flash Error device reset.

14.1.1. Flash Lock and Key Functions

Flash writes and erases by user software are protected with a lock and key function. The Flash Lock and Key Register (FLKEY) must be written with the correct key codes, in sequence, before Flash operations may be performed. The key codes are: 0xA5, 0xF1. The timing does not matter, but the codes must be written in order. If the key codes are written out of order, or the wrong codes are written, Flash writes and erases will be disabled until the next system reset. Flash writes and erases will also be disabled if a Flash write or erase is attempted before the key codes have been written properly. The Flash lock resets after each write or erase; the key codes must be written again before a following Flash operation can be performed. The FLKEY register is detailed in SFR Definition 14.4.

15.2. Idle Mode

Setting the Idle Mode Select bit (PCON.0) causes the CIP-51 to halt the CPU and enter Idle mode as soon as the instruction that sets the bit completes execution. All internal registers and memory maintain their original data. All analog and digital peripherals can remain active during Idle mode.

Note: To ensure the MCU enters a low power state upon entry into Idle Mode, the one-shot circuit should be enabled by clearing the BYPASS bit (FLSCL.6).

Idle mode is terminated when an enabled interrupt is asserted or a reset occurs. The assertion of an enabled interrupt will cause the Idle Mode Selection bit (PCON.0) to be cleared and the CPU to resume operation. The pending interrupt will be serviced and the next instruction to be executed after the return from interrupt (RETI) will be the instruction immediately following the one that set the Idle Mode Select bit. If Idle mode is terminated by an internal or external reset, the CIP-51 performs a normal reset sequence and begins program execution at address 0x0000.

If enabled, the Watchdog Timer (WDT) will eventually cause an internal watchdog reset and thereby terminate the Idle mode. This feature protects the system from an unintended permanent shutdown in the event of an inadvertent write to the PCON register. If this behavior is not desired, the WDT may be disabled by software prior to entering the idle mode if the WDT was initially configured to allow this operation. This provides the opportunity for additional power savings, allowing the system to remain in the idle mode indefinitely, waiting for an external stimulus to wake up the system. Refer to Section “18.6. PCA Watchdog Timer Reset” on page 185 for more information on the use and configuration of the WDT.

15.3. Stop Mode

Setting the Stop Mode Select bit (PCON.1) causes the CIP-51 to enter stop mode as soon as the instruction that sets the bit completes execution. In stop mode the precision internal oscillator and CPU are stopped; the state of the low power oscillator and the external oscillator circuit is not affected. Each analog peripheral (including the external oscillator circuit) may be shut down individually prior to entering stop mode. Stop mode can only be terminated by an internal or external reset. On reset, the CIP-51 performs the normal reset sequence and begins program execution at address 0x0000.

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

Stop mode is a legacy 8051 power mode; it will not result in optimal power savings. Sleep or suspend mode will provide more power savings if the MCU needs to be inactive for a long period of time.

Note: To ensure the MCU enters a low power state upon entry into Stop Mode, the one-shot circuit should be enabled by clearing the BYPASS bit (FLSCL.6).

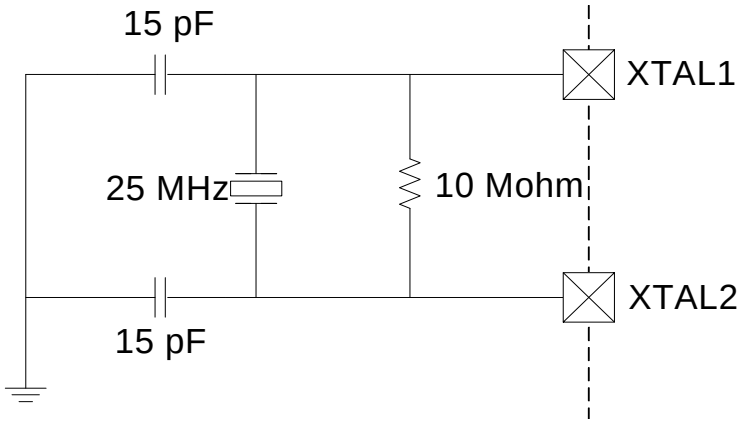

Figure 19.2. 25 MHz External Crystal Example

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.

When using an external crystal, the external oscillator drive circuit must be configured by software for *Crystal Oscillator Mode* or *Crystal Oscillator Mode with divide by 2 stage*. The divide by 2 stage ensures that the clock derived from the external oscillator has a duty cycle of 50%. The External Oscillator Frequency Control value (XFCN) must also be specified based on the crystal frequency. The selection should be based on Table 19.1. For example, a 25 MHz crystal requires an XFCN setting of 111b.

Table 19.1. Recommended XFCN Settings for Crystal Mode

XFCN	Crystal Frequency	Bias Current	Typical Supply Current (VDD = 2.4 V)
000	f ≤ 20 kHz	0.5 µA	3.0 µA, f = 32.768 kHz
001	20 kHz < f ≤ 58 kHz	1.5 µA	4.8 µA, f = 32.768 kHz
010	58 kHz < f ≤ 155 kHz	4.8 µA	9.6 µA, f = 32.768 kHz
011	155 kHz < f ≤ 415 kHz	14 µA	28 µA, f = 400 kHz
100	415 kHz < f ≤ 1.1 MHz	40 µA	71 µA, f = 400 kHz
101	1.1 MHz < f ≤ 3.1 MHz	120 µA	193 µA, f = 400 kHz
110	3.1 MHz < f ≤ 8.2 MHz	550 µA	940 µA, f = 8 MHz
111	8.2 MHz < f ≤ 25 MHz	2.6 mA	3.9 mA, f = 25 MHz

When the crystal oscillator is first enabled, the external oscillator valid detector allows software to determine when the external system clock has stabilized. Switching to the external oscillator before the crystal oscillator has stabilized can result in unpredictable behavior. The recommended procedure for starting the crystal is as follows:

1. Configure XTAL1 and XTAL2 for analog I/O and disable the digital output drivers.
2. Configure and enable the external oscillator.
3. Poll for XTLVLD $\geq$ 1.
4. Switch the system clock to the external oscillator.

21.4. Port Match

Port match functionality allows system events to be triggered by a logic value change on P0 or P1. A software controlled value stored in the PnMAT registers specifies the expected or normal logic values of P0 and P1. A Port mismatch event occurs if the logic levels of the Port's input pins no longer match the software controlled value. This allows Software to be notified if a certain change or pattern occurs on P0 or P1 input pins regardless of the XBRn settings.

The PnMASK registers can be used to individually select which P0 and P1 pins should be compared against the PnMAT registers. A Port mismatch event is generated if (P0 & P0MASK) does not equal (PnMAT & P0MASK) or if (P1 & P1MASK) does not equal (PnMAT & P1MASK).

A Port mismatch event may be used to generate an interrupt or wake the device from a low power mode. See Section "13. Interrupt Handler" on page 138 and Section "15. Power Management" on page 162 for more details on interrupt and wake-up sources.

SFR Definition 21.4. P0MASK: Port0 Mask Register

Bit	7	6	5	4	3	2	1	0
Name	P0MASK[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page= 0x0; SFR Address = 0xC7

Bit	Name	Function
7:0	P0MASK[7:0]	**Port0 Mask Value.** Selects the P0 pins to be compared with the corresponding bits in P0MAT. 0: P0.n pin pad logic value is ignored and cannot cause a Port Mismatch event. 1: P0.n pin pad logic value is compared to P0MAT.n.

SFR Definition 21.5. P0MAT: Port0 Match Register

Bit	7	6	5	4	3	2	1	0
Name	P0MAT[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Page= 0x0; SFR Address = 0xD7

Bit	Name	Function
7:0	P0MAT[7:0]	**Port 0 Match Value.** Match comparison value used on Port 0 for bits in P0MASK which are set to 1. 0: P0.n pin logic value is compared with logic LOW. 1: P0.n pin logic value is compared with logic HIGH.

Table 22.3. 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.

25.1.3. 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 (TCON.5) 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 (TMOD.3) is logic 0 or when the input signal $\overline{INT0}$ is active as defined by bit IN0PL in register IT01CF (see Section “13.6. External Interrupts INT0 and INT1” on page 148 for details on the external input signals $\overline{INT0}$ and $\overline{INT1}$).

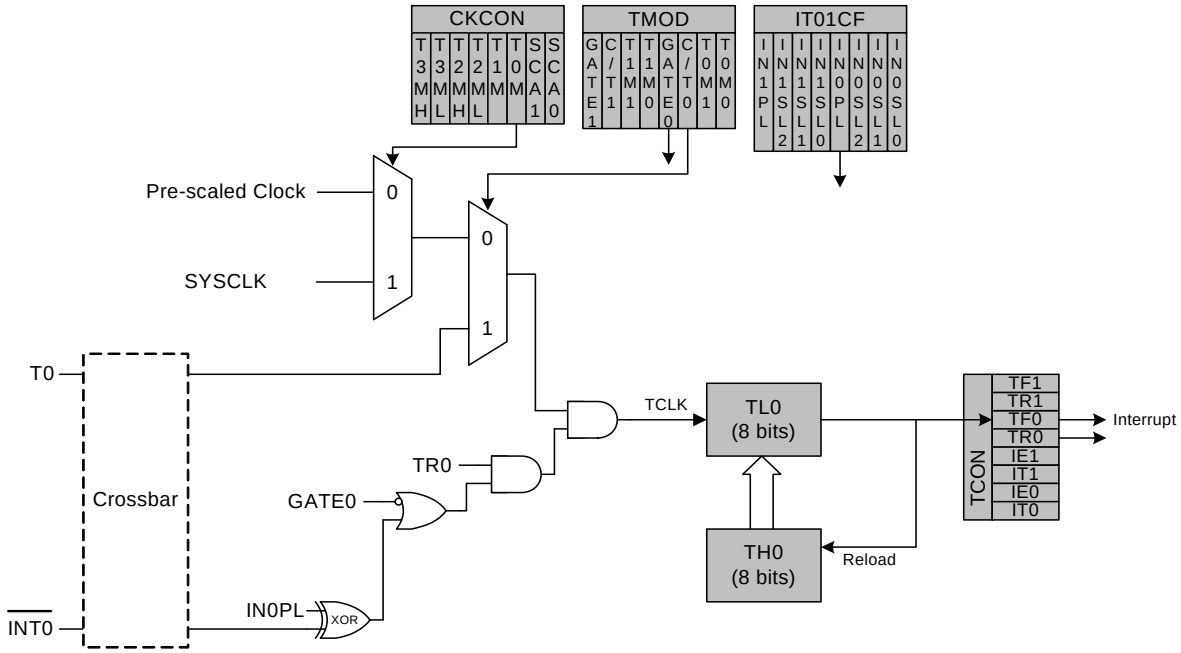

Figure 25.2. T0 Mode 2 Block Diagram

SFR Definition 25.4. 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 Page = 0x0; SFR Address = 0x8A

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 25.5. 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 Page = 0x0; SFR Address = 0x8B

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.

25.2. Timer 2

Timer 2 is a 16-bit timer formed by two 8-bit SFRs: TMR2L (low byte) and TMR2H (high byte). Timer 2 may operate in 16-bit auto-reload mode or (split) 8-bit auto-reload mode. The T2SPLIT bit (TMR2CN.3) defines the Timer 2 operation mode. Timer 2 can also be used in Capture Mode to measure the SmaRTClock or the Comparator 0 period with respect to another oscillator. The ability to measure the Comparator 0 period with respect to the system clock is makes using Touch Sense Switches very easy.

Timer 2 may be clocked by the system clock, the system clock divided by 12, SmaRTClock divided by 8, or Comparator 0 output. Note that the SmaRTClock divided by 8 and Comparator 0 output is synchronized with the system clock.

25.2.1. 16-bit Timer with Auto-Reload

When T2SPLIT (TMR2CN.3) is zero, Timer 2 operates as a 16-bit timer with auto-reload. Timer 2 can be clocked by SYSCLK, SYSCLK divided by 12, SmaRTClock divided by 8, or Comparator 0 output. As the 16-bit timer register increments and overflows from 0xFFFF to 0x0000, the 16-bit value in the Timer 2 reload registers (TMR2RLH and TMR2RLL) is loaded into the Timer 2 register as shown in Figure 25.4, and the Timer 2 High Byte Overflow Flag (TMR2CN.7) is set. If Timer 2 interrupts are enabled (if IE.5 is set), an interrupt will be generated on each Timer 2 overflow. Additionally, if Timer 2 interrupts are enabled and the TF2LEN bit is set (TMR2CN.5), an interrupt will be generated each time the lower 8 bits (TMR2L) overflow from 0xFF to 0x00.

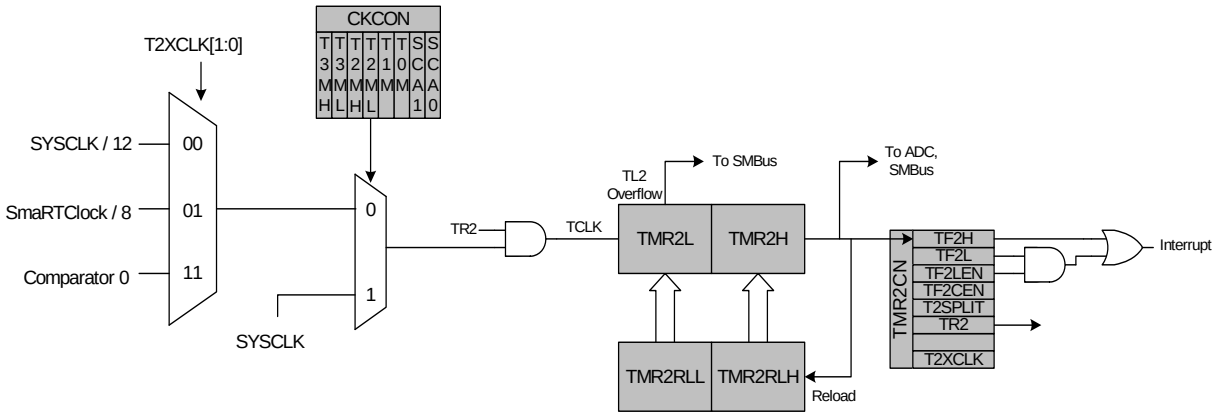

Figure 25.4. Timer 2 16-Bit Mode Block Diagram

SFR Definition 25.9. 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 Page = 0x0; 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 25.10. 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 Page = 0x0; 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.

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.** 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.** 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