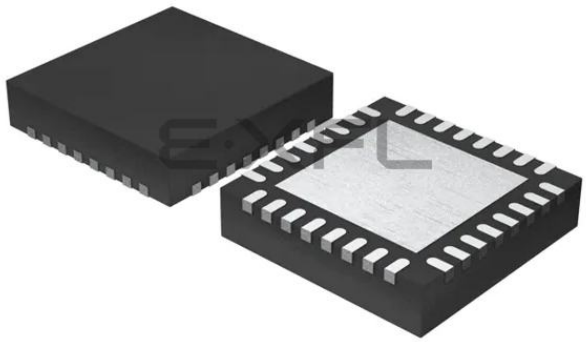

Welcome to E-XFL.COM

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	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	24
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4.25K x 8
Voltage - Supply (Vcc/Vdd)	0.9V ~ 3.6V
Data Converters	A/D 23x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-VFQFN Exposed Pad
Supplier Device Package	32-QFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f930-gm

The ADC can sample the voltage at any of the GPIO pins (with the exception of P2.7) and has an on-chip attenuator that allows it to measure voltages up to twice the voltage reference. Additional ADC inputs include an on-chip temperature sensor, the VDD/DC+ supply voltage, the VBAT supply voltage, and the internal digital supply voltage.

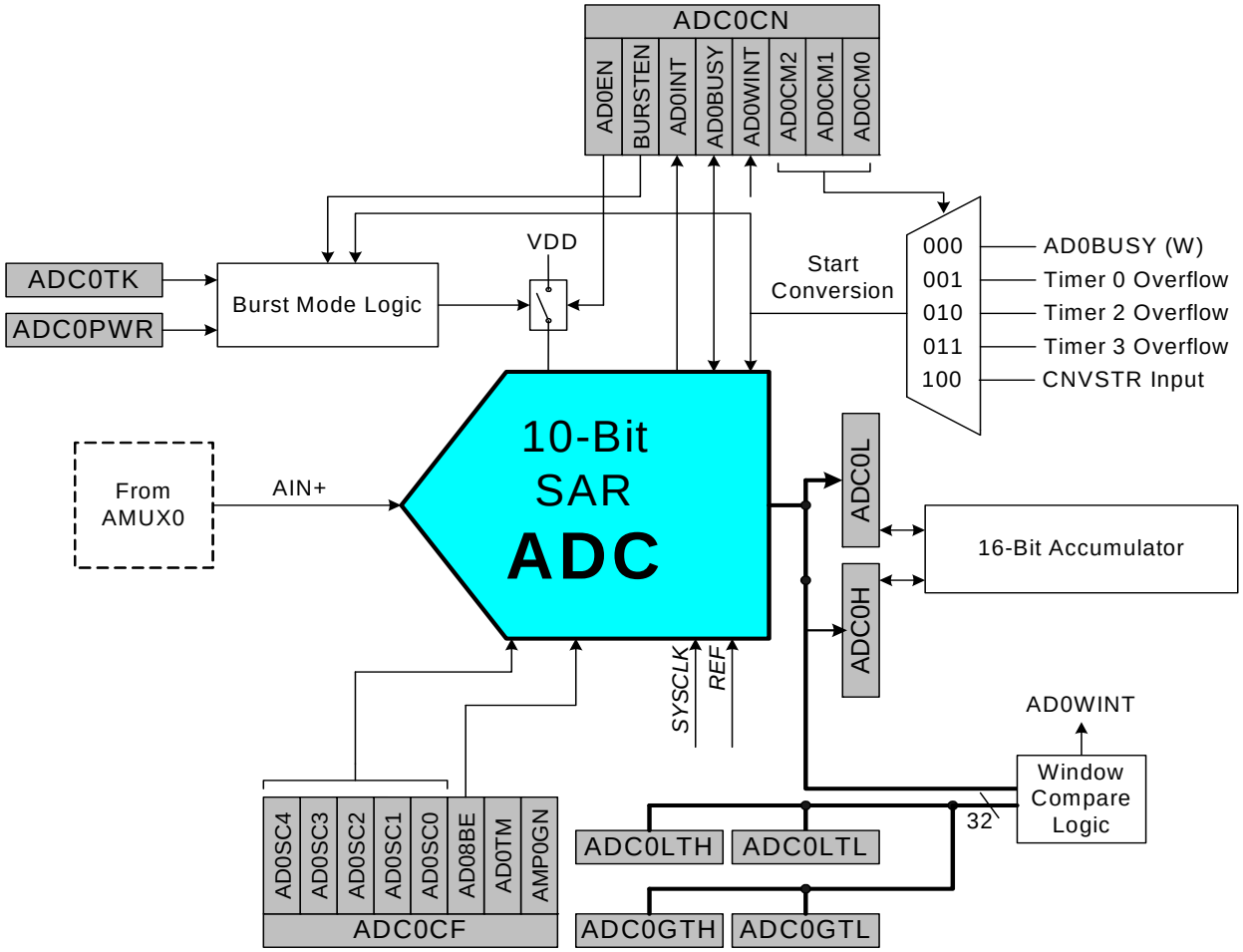

Figure 1.7. ADC0 Functional Block Diagram

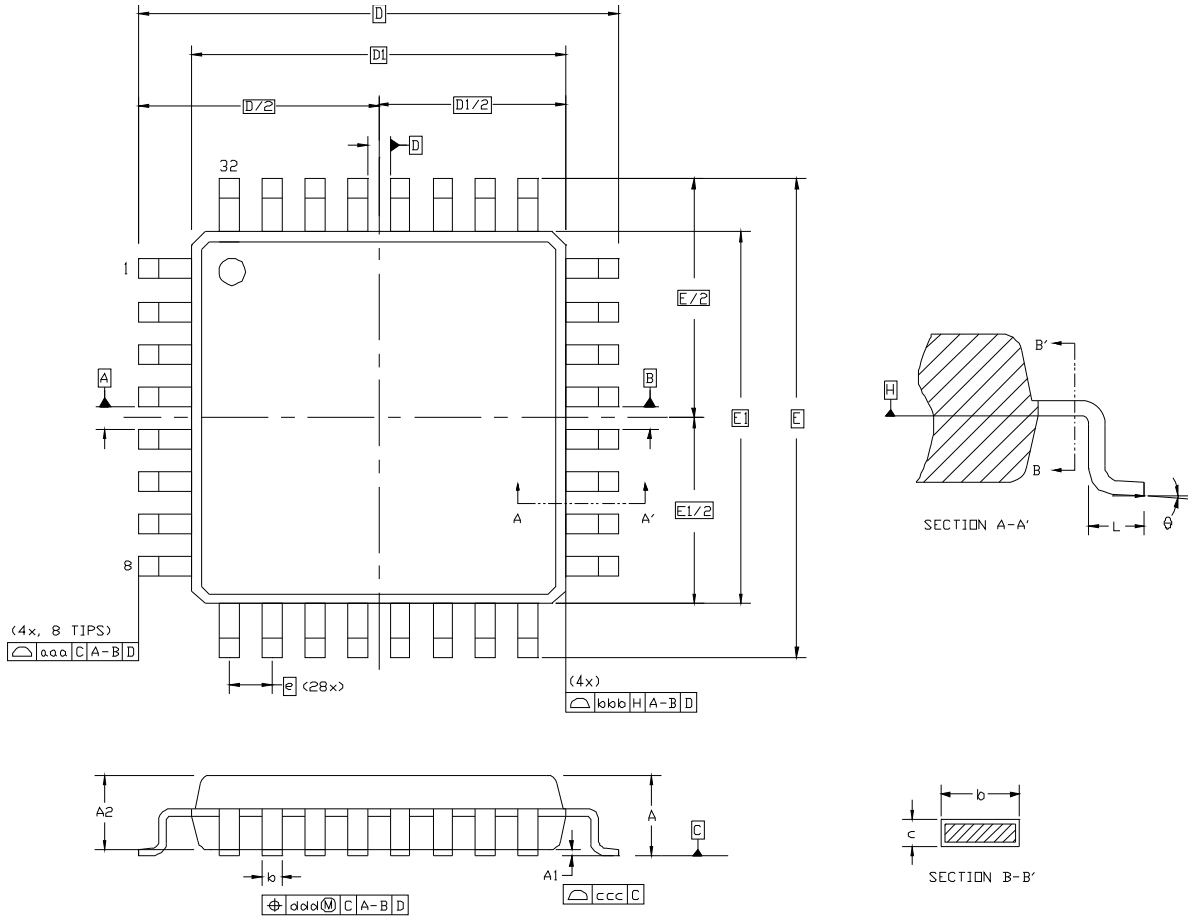

Figure 3.8. LQFP-32 Package Diagram

Table 3.6. LQFP-32 Package Dimensions

Dimension	Min	Typ	Max	Dimension	Min	Typ	Max
A	—	—	1.60	E	9.00 BSC		
A1	0.05	—	0.15	E1	7.00 BSC		
A2	1.35	1.40	1.45	L	0.45	0.60	0.75
b	0.30	0.37	0.45	aaa	0.20		
c	0.09	—	0.20	bbb	0.20		
D	9.00 BSC.			ccc	0.10		
D1	7.00 BSC			ddd	0.20		
e	0.80 BSC			θ	0º	3.5º	7º

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 JEDEC outline MS-026, variation BBA.
4. Recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

7. Comparators

C8051F93x-C8051F92x devices include two on-chip programmable voltage comparators: Comparator 0 (CPT0) is shown in Figure 7.1; Comparator 1 (CPT1) is shown in Figure 7.2. The two comparators operate identically, but may differ in their ability to be used as reset or wake-up sources. See the Reset Sources chapter and the Power Management chapter for details on reset sources and low power mode wake-up sources, respectively.

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, CP1), or an asynchronous “raw” output (CP0A, CP1A). 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.

7.1. Comparator Inputs

Each Comparator performs an analog comparison of the voltage levels at its positive (CP0+ or CP1+) and negative (CP0- or CP1-) input. Both comparators support multiple port pin inputs multiplexed to their positive and negative comparator inputs using analog input multiplexers. The analog input multiplexers are completely under software control and configured using SFR registers. See Section “7.6. Comparator0 and Comparator1 Analog Multiplexers” on page 97 for details on how to select and configure Comparator inputs.

Important Note About Comparator Inputs: The Port pins selected as Comparator inputs should be configured as analog inputs and skipped by the Crossbar. See the Port I/O chapter for more details on how to configure Port I/O pins as Analog Inputs. The Comparator may also be used to compare the logic level of digital signals, however, Port I/O pins configured as digital inputs must be driven to a valid logic state (HIGH or LOW) to avoid increased power consumption.

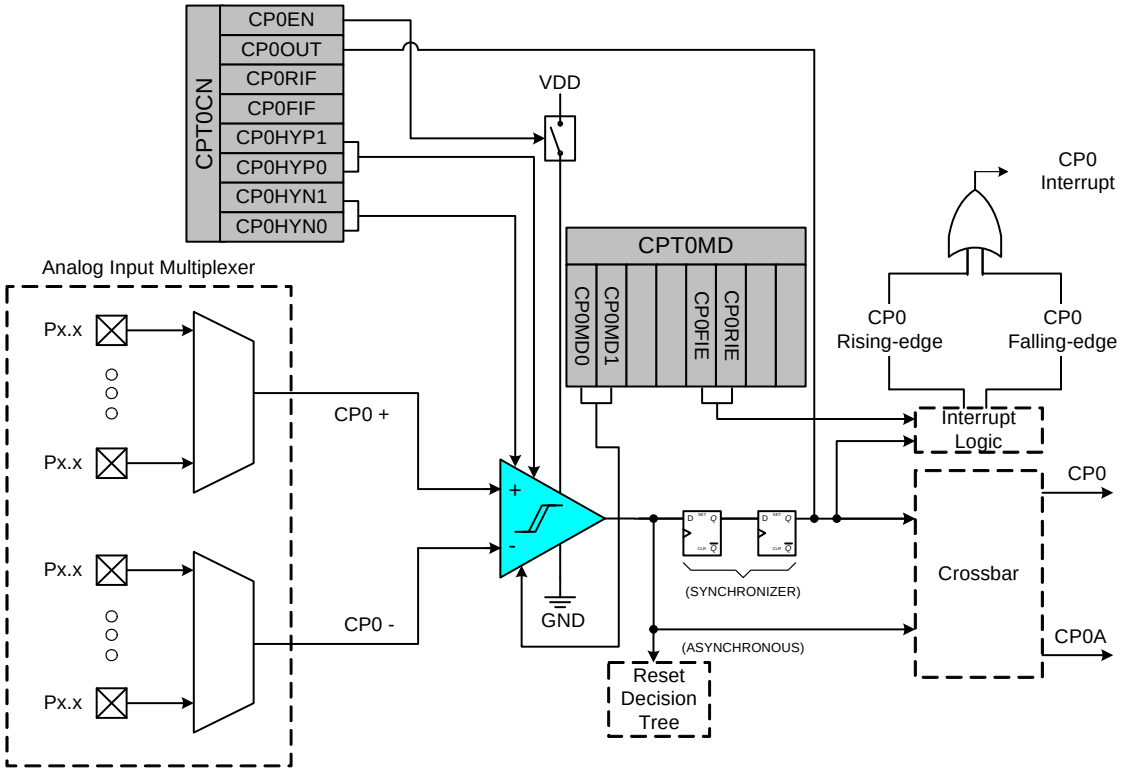

Figure 7.1. Comparator 0 Functional Block Diagram

SFR Definition 7.2. CPT0MD: Comparator 0 Mode Selection

Bit	7	6	5	4	3	2	1	0
Name			CP0RIE	CP0FIE			CP0MD[1:0]	
Type	R/W	R	R/W	R/W	R	R	R/W	
Reset	1	0	0	0	0	0	1	0

SFR Page = All Pages; SFR Address = 0x9D

Bit	Name	Function
7	Reserved	Reserved. Read = 1b, Must Write 1b.
6	Unused	**Unused.** Read = 0b, Write = don't care.
5	CP0RIE	**Comparator0 Rising-Edge Interrupt Enable.** 0: Comparator0 Rising-edge interrupt disabled. 1: Comparator0 Rising-edge interrupt enabled.
4	CP0FIE	**Comparator0 Falling-Edge Interrupt Enable.** 0: Comparator0 Falling-edge interrupt disabled. 1: Comparator0 Falling-edge interrupt enabled.
3:2	Unused	**Unused.** Read = 00b, Write = don't care.
1:0	CP0MD[1:0]	**Comparator0 Mode Select** These bits affect the response time and power consumption for Comparator0. 00: Mode 0 (Fastest Response Time, Highest Power Consumption) 01: Mode 1 10: Mode 2 11: Mode 3 (Slowest Response Time, Lowest Power Consumption)

Table 11.3. Special Function Registers (Continued)

SFRs are listed in alphabetical order. All undefined SFR locations are reserved

Register	Address	SFR Page	Description	Page
PCA0MD	0xD9	0x0	PCA0 Mode	314
PCA0PWM	0xDF	0x0	PCA0 PWM Configuration	315
PCON	0x87	0x0	Power Control	163
PMU0CF	0xB5	0x0	PMU0 Configuration	162
PSCTL	0x8F	0x0	Program Store R/W Control	153
PSW	0xD0	All	Program Status Word	108
REF0CN	0xD1	0x0	Voltage Reference Control	88
REG0CN	0xC9	0x0	Voltage Regulator (VREG0) Control	179
RSTSRC	0xEF	0x0	Reset Source Configuration/Status	186
RTC0ADR	0xAC	0x0	RTC0 Address	201
RTC0DAT	0xAD	0x0	RTC0 Data	201
RTC0KEY	0xAE	0x0	RTC0 Key	200
SBUF0	0x99	0x0	UART0 Data Buffer	261
SCON0	0x98	0x0	UART0 Control	260
SFRPAGE	0xA7	All	SFR Page	128
SMB0ADM	0xF5	0x0	SMBus Slave Address Mask	245
SMB0ADR	0xF4	0x0	SMBus Slave Address	245
SMB0CF	0xC1	0x0	SMBus0 Configuration	240
SMB0CN	0xC0	0x0	SMBus0 Control	242
SMB0DAT	0xC2	0x0	SMBus0 Data	246
SP	0x81	All	Stack Pointer	107
SPI0CFG	0xA1	0x0	SPI0 Configuration	271
SPI0CKR	0xA2	0x0	SPI0 Clock Rate Control	273
SPI0CN	0xF8	0x0	SPI0 Control	272
SPI0DAT	0xA3	0x0	SPI0 Data	274
SPI1CFG	0x84	0x0	SPI1 Configuration	271
SPI1CKR	0x85	0x0	SPI1 Clock Rate Control	273
SPI1CN	0xB0	0x0	SPI1 Control	272
SPI1DAT	0x86	0x0	SPI1 Data	274
TCON	0x88	0x0	Timer/Counter Control	284
TH0	0x8C	0x0	Timer/Counter 0 High	287
TH1	0x8D	0x0	Timer/Counter 1 High	287
TL0	0x8A	0x0	Timer/Counter 0 Low	286
TL1	0x8B	0x0	Timer/Counter 1 Low	286

19.1. Programmable Precision Internal Oscillator

All C8051F93x-C8051F92x devices include a programmable precision internal oscillator that may be selected as the system clock. OSCICL is factory calibrated to obtain a 24.5 MHz frequency. See Section “4. Electrical Characteristics” on page 43 for complete oscillator specifications.

The precision oscillator supports a spread spectrum mode which modulates the output frequency in order to reduce the EMI generated by the system. When enabled (SSE = 1), the oscillator output frequency is modulated by a stepped triangle wave whose frequency is equal to the oscillator frequency divided by 384 (63.8 kHz using the factory calibration). The deviation from the nominal oscillator frequency is +0%, –1.6%, and the step size is typically 0.26% of the nominal frequency. When using this mode, the typical average oscillator frequency is lowered from 24.5 MHz to 24.3 MHz.

19.2. Low Power Internal Oscillator

All C8051F93x-C8051F92x devices include a low power internal oscillator that defaults as the system clock after a system reset. The low power internal oscillator frequency is 20 MHz ± 10% and is automatically enabled when selected as the system clock and disabled when not in use. See Section “4. Electrical Characteristics” on page 43 for complete oscillator specifications.

19.3. External Oscillator Drive Circuit

All C8051F93x-C8051F92x devices include an external oscillator circuit that may drive an external crystal, ceramic resonator, capacitor, or RC network. A CMOS clock may also provide a clock input. Figure 19.1 shows a block diagram of the four external oscillator options. The external oscillator is enabled and configured using the OSCXCN register.

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 “4. Electrical Characteristics” on page 43 for complete oscillator specifications.

19.3.1. 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 19.1, Option 1. 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.

Figure 19.2 shows the external oscillator circuit for a 20 MHz quartz crystal with a manufacturer recommended load capacitance of 12.5 pF. Loading capacitors are "in series" as seen by the crystal and "in parallel" with the stray capacitance of the XTAL1 and XTAL2 pins. The total value of the each loading capacitor and the stray capacitance of each XTAL pin should equal 12.5pF x 2 = 25 pF. With a stray capacitance of 10 pF per pin, the 15 pF capacitors yield an equivalent series capacitance of 12.5 pF across the crystal.

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

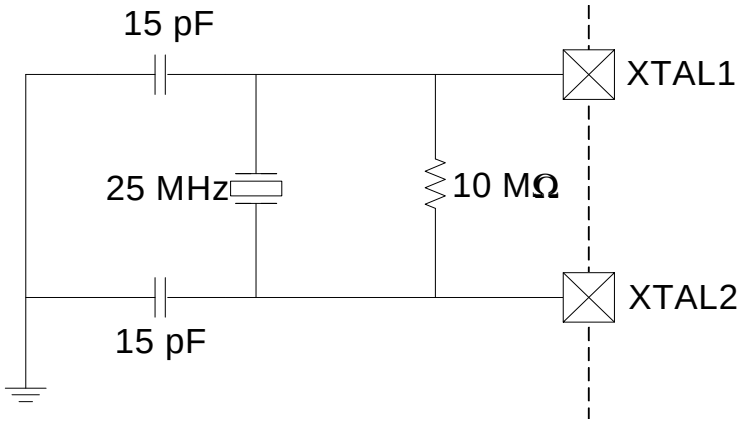

Figure 19.2. 25 MHz External Crystal Example

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:

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 ≥ 1.
4. Switch the system clock to the external oscillator.

19.3.4. External CMOS Clock Mode

If an external CMOS clock is used as the external oscillator, the clock should be directly routed into XTAL2. The XTAL2 pin should be configured as a digital input. XTAL1 is not used in external CMOS clock mode.

The external oscillator valid detector will always return zero when the external oscillator is configured to External CMOS Clock mode.

Table 20.3 shows a summary of the oscillator bias settings. The SmaRTClock Bias Doubling feature allows the self-oscillation frequency to be increased (almost doubled) and allows a higher crystal drive strength in crystal mode. High crystal drive strength is recommended when the crystal is exposed to poor environmental conditions such as excessive moisture. SmaRTClock Bias Doubling is enabled by setting BIASX2 (RTC0XCN.5) to 1.

.

Table 20.3. SmaRTClock Bias Settings

Mode	Setting	Power Consumption
Crystal	Bias Double Off, AGC On	Lowest 600 nA
	Bias Double Off, AGC Off	Low 800 nA
	Bias Double On, AGC On	High
	Bias Double On, AGC Off	Highest
Self-Oscillate	Bias Double Off	Low
	Bias Double On	High

Internal Register Definition 20.6. RTC0XCF: SmaRTClock Oscillator Configuration

Bit	7	6	5	4	3	2	1	0
Name	AUTOSTP	LOADRDY			LOADCAP			
Type	R/W	R	R	R	R/W			
Reset	0	0	0	0	Varies	Varies	Varies	Varies

SmaRTClock Address = 0x06

Bit	Name	Function
7	AUTOSTP	**Automatic Load Capacitance Stepping Enable.** Enables/disables automatic load capacitance stepping. 0: Load capacitance stepping disabled. 1: Load capacitance stepping enabled.
6	LOADRDY	**Load Capacitance Ready Indicator.** Set by hardware when the load capacitance matches the programmed value. 0: Load capacitance is currently stepping. 1: Load capacitance has reached it programmed value.
5:4	Unused	**Unused.** Read = 00b; Write = Don't Care.
3:0	LOADCAP	**Load Capacitance Programmed Value.** Holds the user's desired value of the load capacitance. See Table 20.2 on page 203.

Internal Register Definition 20.7. RTC0PIN: SmaRTClock Pin Configuration

Bit	7	6	5	4	3	2	1	0
Name	RTC0PIN							
Type	W							
Reset	0	1	1	0	0	1	1	1

SmaRTClock Address = 0x07

Bit	Name	Function
7:0	RTC0PIN	**SmaRTClock Pin Configuration.** Writing 0xE7 to this register forces XTAL3 and XTAL4 to be internally shorted for use with Self Oscillate Mode. Writing 0x67 returns XTAL3 and XTAL4 to their normal configuration.

SFR Definition 21.12. P0DRV: Port0 Drive Strength

Bit	7	6	5	4	3	2	1	0
Name	P0DRV[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page = 0xF; SFR Address = 0xA4

Bit	Name	Function
7:0	P0DRV[7:0]	**Drive Strength Configuration Bits for P0.7–P0.0 (respectively).** Configures digital I/O Port cells to high or low output drive strength. 0: Corresponding P0.n Output has low output drive strength. 1: Corresponding P0.n Output has high output drive strength.

SFR Definition 21.15. P1MDIN: Port1 Input Mode

Bit	7	6	5	4	3	2	1	0
Name	P1MDIN[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Page = 0x0; SFR Address = 0xF2

Bit	Name	Function
7:0	P1MDIN[7:0]	**Analog Configuration Bits for P1.7–P1.0 (respectively).** Port pins configured for analog mode have their weak pullup and digital receiver disabled. The digital driver is not explicitly disabled. 0: Corresponding P1.n pin is configured for analog mode. 1: Corresponding P1.n pin is not configured for analog mode.
Note: Pin P1.7 is only available in 32-pin devices.		

SFR Definition 21.16. P1MDOUT: Port1 Output Mode

Bit	7	6	5	4	3	2	1	0
Name	P1MDOUT[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xA5

Bit	Name	Function
7:0	P1MDOUT[7:0]	**Output Configuration Bits for P1.7–P1.0 (respectively).** These bits control the digital driver even when the corresponding bit in register P1MDIN is logic 0. 0: Corresponding P1.n Output is open-drain. 1: Corresponding P1.n Output is push-pull.
Note: Pin P1.7 is only available in 32-pin devices.		

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.

22.5. SMBus Transfer Modes

The SMBus interface may be configured to operate as master and/or slave. At any particular time, it will be operating in one of the following four modes: Master Transmitter, Master Receiver, Slave Transmitter, or Slave Receiver. The SMBus interface enters Master Mode any time a START is generated, and remains in Master Mode until it loses an arbitration or generates a STOP. An SMBus interrupt is generated at the end of all SMBus byte frames. Note that the position of the ACK interrupt when operating as a receiver depends on whether hardware ACK generation is enabled. As a receiver, the interrupt for an ACK occurs **before** the ACK with hardware ACK generation disabled, and **after** the ACK when hardware ACK generation is enabled. As a transmitter, interrupts occur **after** the ACK, regardless of whether hardware ACK generation is enabled or not.

22.5.1. Write Sequence (Master)

During a write sequence, an SMBus master writes data to a slave device. The master in this transfer will be a transmitter during the address byte, and a transmitter 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 0 (WRITE). The master then transmits one or more bytes of serial data. After each byte is transmitted, an acknowledge bit is generated by the slave. The transfer is ended when the STO bit is set and a STOP is generated. Note that the interface will switch to Master Receiver Mode if SMB0DAT is not written following a Master Transmitter interrupt. Figure 22.5 shows a typical master write sequence. Two transmit data bytes are shown, though any number of bytes may be transmitted. All “data byte transferred” interrupts occur **after** the ACK cycle in this mode, regardless of whether hardware ACK generation is enabled.

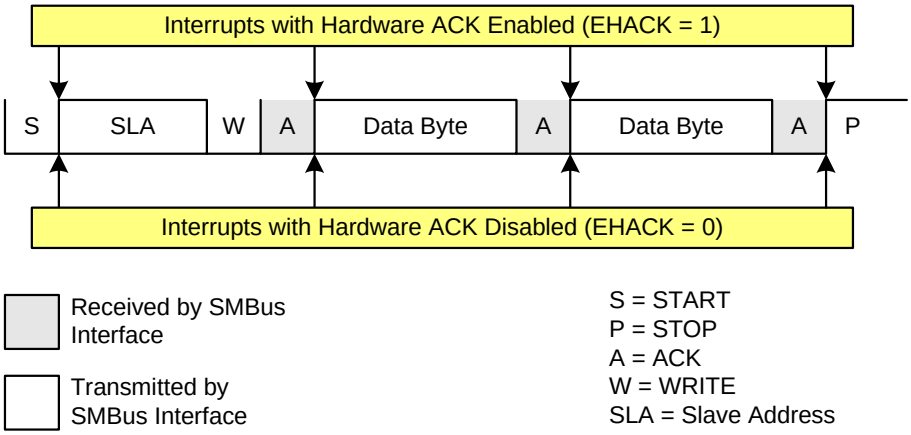

Figure 22.5. Typical Master Write Sequence

23.1. Enhanced Baud Rate Generation

The UART0 baud rate is generated by Timer 1 in 8-bit auto-reload mode. The TX clock is generated by TL1; the RX clock is generated by a copy of TL1 (shown as RX Timer in Figure 23.2), which is not user-accessible. Both TX and RX Timer overflows are divided by two to generate the TX and RX baud rates. The RX Timer runs when Timer 1 is enabled, and uses the same reload value (TH1). However, an RX Timer reload is forced when a START condition is detected on the RX pin. This allows a receive to begin any time a START is detected, independent of the TX Timer state.

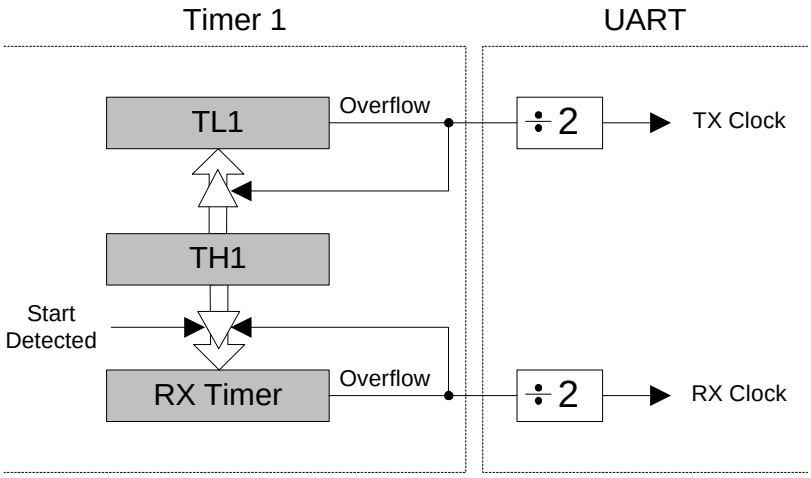

Figure 23.2. UART0 Baud Rate Logic

Timer 1 should be configured for Mode 2, 8-bit auto-reload (see Section “25.1.3. Mode 2: 8-bit Counter/Timer with Auto-Reload” on page 282). The Timer 1 reload value should be set so that overflows will occur at two times the desired UART baud rate frequency. Note that Timer 1 may be clocked by one of six sources: SYSCLK, SYSCLK / 4, SYSCLK / 12, SYSCLK / 48, the external oscillator clock / 8, or an external input T1. For any given Timer 1 clock source, the UART0 baud rate is determined by Equation 23.1-A and Equation 23.1-B.

A) $$\text{UartBaudRate} = \frac{1}{2} \times \text{T1\_Overflow\_Rate}$$

B) $$\text{T1\_Overflow\_Rate} = \frac{\text{T1}_{CLK}}{256 - \text{TH1}}$$

Equation 23.1. UART0 Baud Rate

Where $T1_{CLK}$ is the frequency of the clock supplied to Timer 1, and *T1H* is the high byte of Timer 1 (reload value). Timer 1 clock frequency is selected as described in Section “25.1. Timer 0 and Timer 1” on page 280. A quick reference for typical baud rates and system clock frequencies is given in Table 23.1 through Table 23.2. Note that the internal oscillator may still generate the system clock when the external oscillator is driving Timer 1.

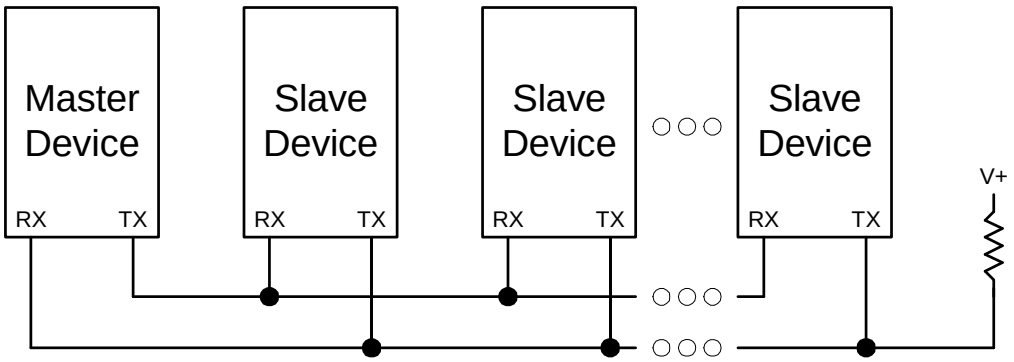

Figure 23.6. UART Multi-Processor Mode Interconnect Diagram

SFR Definition 23.1. SCON0: Serial Port 0 Control

Bit	7	6	5	4	3	2	1	0
Name	S0MODE		MCE0	REN0	TB80	RB80	TI0	RI0
Type	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0x98; Bit-Addressable

Bit	Name	Function
7	S0MODE	**Serial Port 0 Operation Mode.** Selects the UART0 Operation Mode. 0: 8-bit UART with Variable Baud Rate. 1: 9-bit UART with Variable Baud Rate.
6	Unused	**Unused.** Read = 1b. Write = Don't Care.
5	MCE0	**Multiprocessor Communication Enable.** **For Mode 0 (8-bit UART): Checks for valid stop bit.** 0: Logic level of stop bit is ignored. 1: RI0 will only be activated if stop bit is logic level 1. **For Mode 1 (9-bit UART): Multiprocessor Communications Enable.** 0: Logic level of ninth bit is ignored. 1: RI0 is set and an interrupt is generated only when the ninth bit is logic 1.
4	REN0	**Receive Enable.** 0: UART0 reception disabled. 1: UART0 reception enabled.
3	TB80	**Ninth Transmission Bit.** The logic level of this bit will be sent as the ninth transmission bit in 9-bit UART Mode (Mode 1). Unused in 8-bit mode (Mode 0).
2	RB80	**Ninth Receive Bit.** RB80 is assigned the value of the STOP bit in Mode 0; it is assigned the value of the 9th data bit in Mode 1.
1	TI0	**Transmit Interrupt Flag.** Set by hardware when a byte of data has been transmitted by UART0 (after the 8th bit in 8-bit UART Mode, or at the beginning of the STOP bit in 9-bit UART Mode). When the UART0 interrupt is enabled, setting this bit causes the CPU to vector to the UART0 interrupt service routine. This bit must be cleared manually by software.
0	RI0	**Receive Interrupt Flag.** Set to 1 by hardware when a byte of data has been received by UART0 (set at the STOP bit sampling time). When the UART0 interrupt is enabled, setting this bit to 1 causes the CPU to vector to the UART0 interrupt service routine. This bit must be cleared manually by software.

SFR Definition 25.3. TMOD: Timer Mode

Bit	7	6	5	4	3	2	1	0
Name	GATE1	C/T1	T1M[1:0]		GATE0	C/T0	T0M[1:0]	
Type	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 = 0x89

Bit	Name	Function
7	GATE1	**Timer 1 Gate Control.** 0: Timer 1 enabled when TR1 = 1 irrespective of $\overline{\text{INT1}}$ logic level. 1: Timer 1 enabled only when TR1 = 1 AND $\overline{\text{INT1}}$ is active as defined by bit IN1PL in register IT01CF (see SFR Definition 12.7).
6	C/T1	**Counter/Timer 1 Select.** 0: Timer: Timer 1 incremented by clock defined by T1M bit in register CKCON. 1: Counter: Timer 1 incremented by high-to-low transitions on external pin (T1).
5:4	T1M[1:0]	**Timer 1 Mode Select.** These bits select the Timer 1 operation mode. 00: Mode 0, 13-bit Counter/Timer 01: Mode 1, 16-bit Counter/Timer 10: Mode 2, 8-bit Counter/Timer with Auto-Reload 11: Mode 3, Timer 1 Inactive
3	GATE0	**Timer 0 Gate Control.** 0: Timer 0 enabled when TR0 = 1 irrespective of $\overline{\text{INT0}}$ logic level. 1: Timer 0 enabled only when TR0 = 1 AND $\overline{\text{INT0}}$ is active as defined by bit IN0PL in register IT01CF (see SFR Definition 12.7).
2	C/T0	**Counter/Timer 0 Select.** 0: Timer: Timer 0 incremented by clock defined by T0M bit in register CKCON. 1: Counter: Timer 0 incremented by high-to-low transitions on external pin (T0).
1:0	T0M[1:0]	**Timer 0 Mode Select.** These bits select the Timer 0 operation mode. 00: Mode 0, 13-bit Counter/Timer 01: Mode 1, 16-bit Counter/Timer 10: Mode 2, 8-bit Counter/Timer with Auto-Reload 11: Mode 3, Two 8-bit Counter/Timers

25.2.3. Comparator 0/SmaRTClock Capture Mode

The Capture Mode in Timer 2 allows either Comparator 0 or the SmaRTClock period to be measured against the system clock or the system clock divided by 12. Comparator 0 and the SmaRTClock period can also be compared against each other. Timer 2 Capture Mode is enabled by setting TF2CEN to 1. Timer 2 should be in 16-bit auto-reload mode when using Capture Mode.

When Capture Mode is enabled, a capture event will be generated either every Comparator 0 rising edge or every 8 SmaRTClock clock cycles, depending on the T2XCLK1 setting. When the capture event occurs, the contents of Timer 2 (TMR2H:TMR2L) are loaded into the Timer 2 reload registers (TMR2RLH:TMR2RLL) and the TF2H flag is set (triggering an interrupt if Timer 2 interrupts are enabled). By recording the difference between two successive timer capture values, the Comparator 0 or SmaRTClock period can be determined with respect to the Timer 2 clock. The Timer 2 clock should be much faster than the capture clock to achieve an accurate reading.

For example, if T2ML = 1b, T2XCLK1 = 0b, and TF2CEN = 1b, Timer 2 will clock every SYSCLK and capture every SmaRTClock clock divided by 8. If the SYSCLK is 24.5 MHz and the difference between two successive captures is 5984, then the SmaRTClock clock is:

24.5 MHz/(5984/8) = 0.032754 MHz or 32.754 kHz.

This mode allows software to determine the exact SmaRTClock frequency in self-oscillate mode and the time between consecutive Comparator 0 rising edges, which is useful for detecting changes in the capacitance of a Touch Sense Switch.

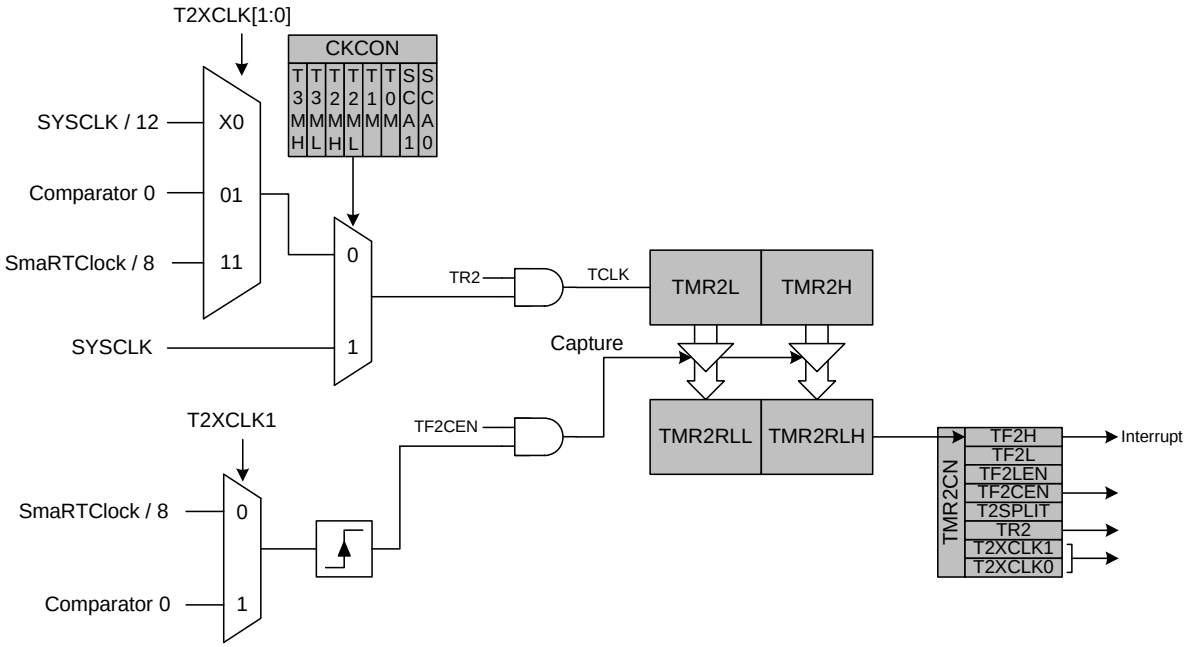

Figure 25.6. Timer 2 Capture Mode Block Diagram