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	25MHz
Connectivity	SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	16
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.25V
Data Converters	A/D 16x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	20-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	20-TSSOP
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f531-c-itr

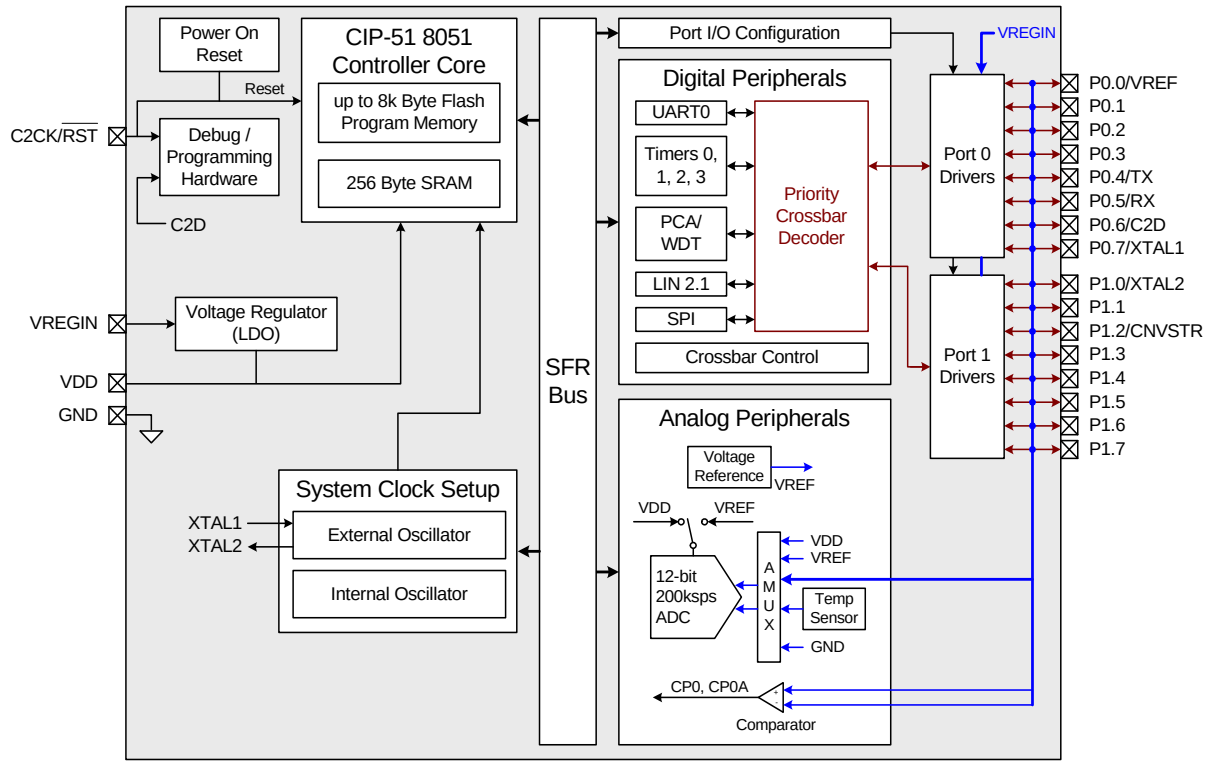

Figure 1.1. C8051F53xA/F53x-C Block Diagram

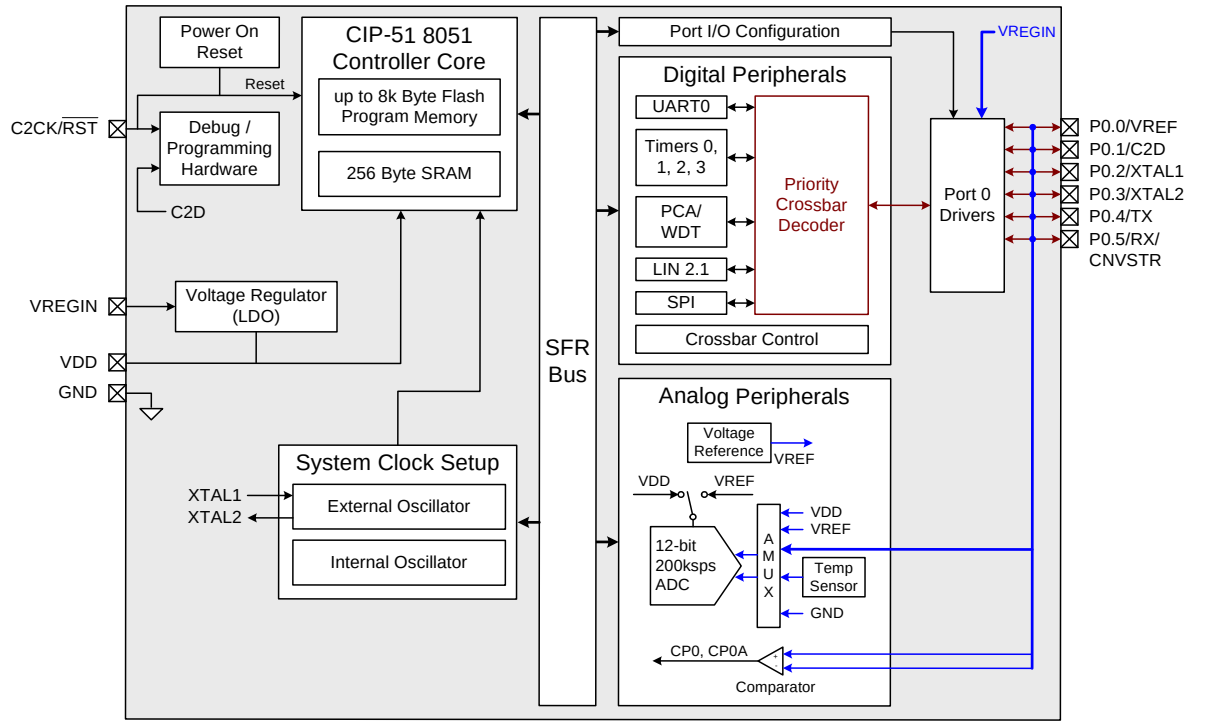

Figure 1.2. C8051F52xA/F52x-C Block Diagram

1.3. On-Chip Memory

The CIP-51 has a standard 8051 program and data address configuration. It includes 256 bytes of data RAM, with the upper 128 bytes dual-mapped. Indirect addressing accesses the upper 128 bytes of general purpose RAM, and direct addressing accesses the 128 byte SFR address space. The lower 128 bytes of RAM are accessible via direct and indirect addressing. The first 32 bytes are addressable as four banks of general purpose registers, and the next 16 bytes can be byte addressable or bit addressable.

Program memory consists of 7680 bytes (’F520/0A/1/1A and ’F530/0A/1/1A), 4 kB (’F523/3A/4/4A and C8051F53x/53xA), or 2 kB (’F526/6A/7/7A and ’F536/6A/7/7A) of Flash. This memory is byte writable and erased in 512-byte sectors, and requires no special off-chip programming voltage.

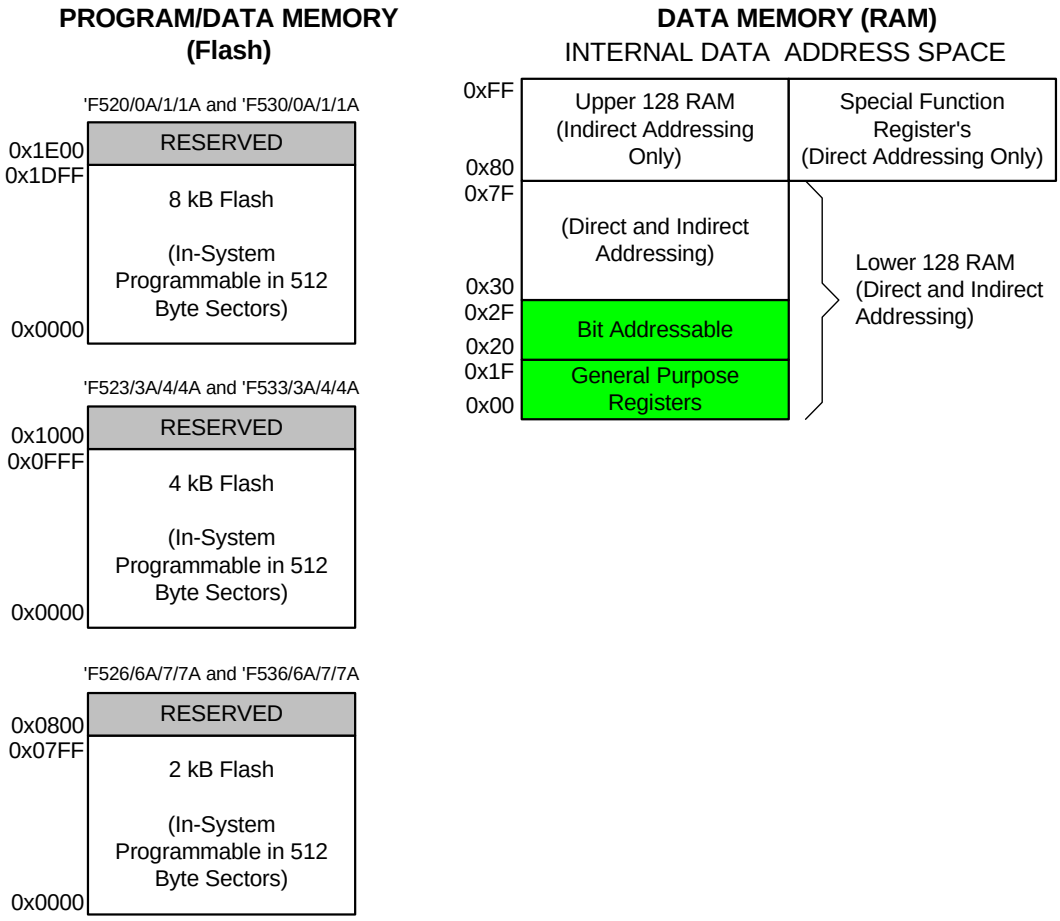

Figure 1.6. Memory Map

1.5. 12-Bit Analog to Digital Converter

The C8051F52x/F52xA/F53x/F53xA devices include an on-chip 12-bit SAR ADC with a maximum throughput of 200 ksps. The ADC system includes a configurable analog multiplexer that selects the positive ADC input, which is measured with respect to GND. Ports 0 and 1 are available as ADC inputs; additionally, the ADC includes an innovative programmable gain stage which allows the ADC to sample inputs sources greater than the VREF voltage. The on-chip Temperature Sensor output and the core supply voltage (V_{DD}) are also available as ADC inputs. User firmware may shut down the ADC or use it in Burst Mode to save power.

Conversions can be initiated in four ways: a software command, an overflow of Timer 1, an overflow of Timer 2, or an external convert start signal. This flexibility allows the start of conversion to be triggered by software events, a periodic signal (timer overflows), or external HW signals. Conversion completions are indicated by a status bit and an interrupt (if enabled) and occur after 1, 4, 8, or 16 samples have been accumulated by a hardware accumulator. The resulting 12-bit to 16-bit data word is latched into the ADC data SFRs upon completion of a conversion. When the system clock is slow, Burst Mode allows ADC0 to automatically wake from a low power shutdown state, acquire and accumulate samples, then re-enter the low power shutdown state without CPU intervention.

Window compare registers for the ADC data can be configured to interrupt the controller when ADC data is either within or outside of a specified range. The ADC can monitor a key voltage continuously in background mode, but not interrupt the controller unless the converted data is within/outside the specified range.

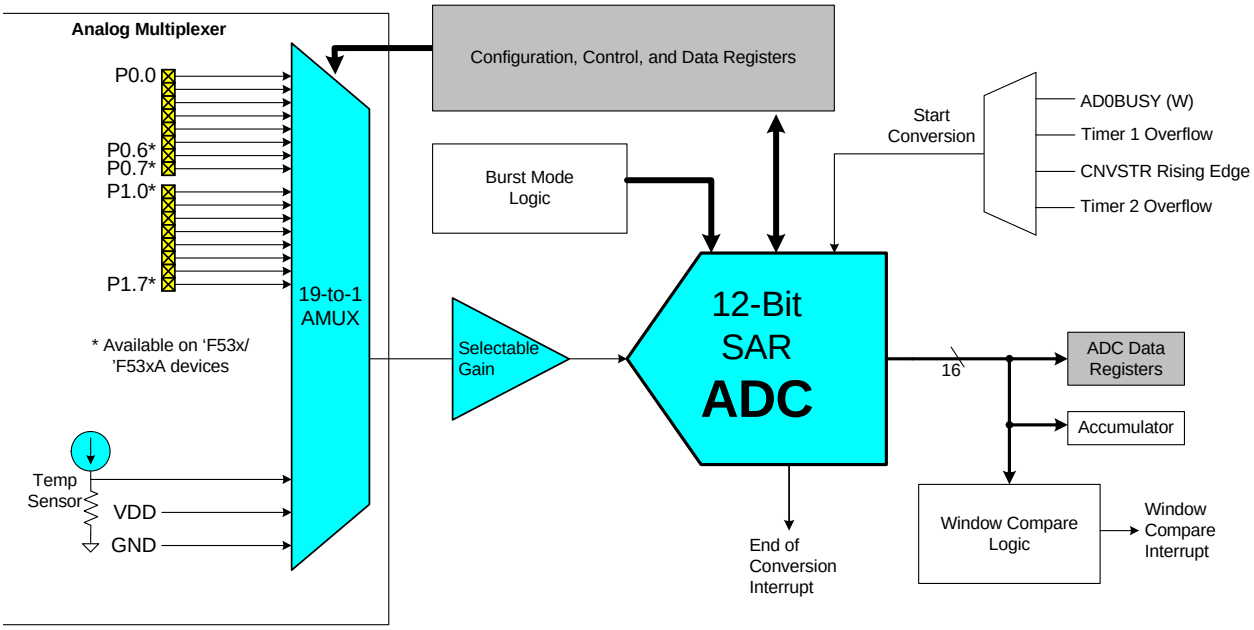

Figure 1.7. 12-Bit ADC Block Diagram

2.2. Electrical Characteristics

Table 2.2. Global DC Electrical Characteristics

–40 to +125 °C, 25 MHz System Clock unless otherwise specified. Typical values are given at 25 °C

Parameter	Conditions	Min	Typ	Max	Units
Supply Input Voltage (V_{REGIN})[1]	Output Current ≤ 1 mA				
	C8051F52x/53x	2.7	—	5.25	V
	C8051F52xA/53xA	1.8[1]	—	5.25	V
	C8051F52x-C/53x-C	2.0[1]	—	5.25	V
Digital Supply Voltage (V_{DD})	C8051F52x/53x	2.0	—	2.7	V
	C8051F52xA/53xA	1.8	—	2.7	V
	C8051F52x-C/53x-C	2.0	—	2.75	V
Core Supply RAM Data Retention Voltage		—	1.5	—	V
SYSCLK (System Clock)[2]		0	—	25	MHz
Specified Operating Temperature Range		–40	—	+125	°C
Digital Supply Current—CPU Active (Normal Mode, fetching instructions from Flash)					
I_{DD}[3,4]	**V_{DD} = 2.1 V:**				
	Clock = 32 kHz	—	13	—	µA
	Clock = 200 kHz	—	60	—	µA
	Clock = 1 MHz	—	0.28	—	mA
	Clock = 25 MHz	—	5.1	9	mA
	V_{DD} = 2.6 V:				
	Clock = 32 kHz	—	22	—	µA
	Clock = 200 kHz	—	105	—	µA
	Clock = 1 MHz	—	0.5	—	mA
	Clock = 25 MHz	—	7.3	13	mA
I_{DD} Frequency Sensitivity[3,5]	T = 25 °C:				
	V_{DD} = 2.1 V, F ≤ 12 MHz	—	0.276	—	mA/MHz
	V_{DD} = 2.1 V, F > 12 MHz	—	0.140	—	mA/MHz
	V_{DD} = 2.6 V, F ≤ 12 MHz	—	0.424	—	mA/MHz
	V_{DD} = 2.6 V, F > 12 MHz	—	0.184	—	mA/MHz

Notes:

1. For more information on V_{REGIN} characteristics, see Table 2.6 on page 30.
2. SYSCLK must be at least 32 kHz to enable debugging.
3. Based on device characterization data; Not production tested.
4. Does not include internal oscillator or internal regulator supply current.
5. I_{DD} can be estimated for frequencies <= 12 MHz by multiplying the frequency of interest by the frequency sensitivity number for that range. When using these numbers to estimate I_{DD} > 12 MHz, the estimate should be the current at 25 MHz minus the difference in current indicated by the frequency sensitivity number. For example: V_{DD} = 2.6 V; F= 20 MHz, I_{DD} = 7.3 mA – (25 MHz – 20 MHz) x 0.184 mA/MHz = 6.38 mA.
6. Idle I_{DD} can be estimated for frequencies <= 1 MHz by multiplying the frequency of interest by the frequency sensitivity number for that range. When using these numbers to estimate I_{DD} > 1 MHz, the estimate should be the current at 25 MHz minus the difference in current indicated by the frequency sensitivity number. For example: V_{DD} = 2.6 V; F= 5 MHz, Idle I_{DD} = 3 mA – (25 MHz– 5 MHz) x 118 µA/MHz = 0.64 mA.

Table 3.1. Pin Definitions for the C8051F52x and C8051F52xA (DFN 10) (Continued)

Name	Pin Numbers		Type	Description
	‘F52xA ‘F52x-C	‘F52x		
P0.3/TX*/	—	8	D I/O or A In	Port 0.3. See Port I/O Section for a complete description.
XTAL2			D I/O	External Clock Output. For an external crystal or resonator, this pin is the excitation driver. This pin is the external clock input for CMOS, capacitor, or RC oscillator configurations. See Section “14. Oscillators” on page 135.
P0.2	9	9	D I/O or	Port 0.2. See Port I/O Section for a complete description.
XTAL1			A In	External Clock Input. This pin is the external oscillator return for a crystal or resonator. Section “14. Oscillators” on page 135.
P0.1/	10	10	D I/O or A In	Port 0.1. See Port I/O Section for a complete description.
C2D			D I/O	Bi-directional data signal for the C2 Debug Interface
Note: Please refer to Section “20. Device Specific Behavior” on page 210.				

4.3.6. Settling Time Requirements

A minimum tracking time is required before an accurate conversion can be performed. This tracking time is determined by the AMUX0 resistance, the ADC0 sampling capacitance, any external source resistance, and the accuracy required for the conversion.

Figure 4.6 shows the equivalent ADC0 input circuit. The required ADC0 settling time for a given settling accuracy (SA) may be approximated by Equation 4.1. When measuring the Temperature Sensor output, use the settling time specified in Table 2.3 on page 28. See Table 2.3 on page 28 for ADC0 minimum settling time requirements.

$$t = \ln\left(\frac{2^n}{SA}\right) \times R_{TOTAL} C_{SAMPLE}$$

Equation 4.1. ADC0 Settling Time Requirements

Where:
SA is the settling accuracy, given as a fraction of an LSB (for example, 0.25 to settle within 1/4 LSB)
t is the required settling time in seconds
R_{TOTAL} is the sum of the AMUX0 resistance and any external source resistance.
n is the ADC resolution in bits (12).

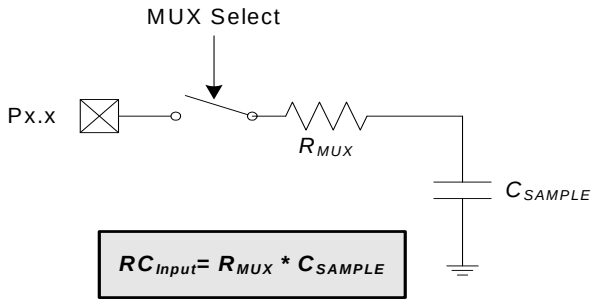

Figure 4.6. ADC0 Equivalent Input Circuits

4.4. Selectable Gain

ADC0 on the C8051F52x/52xA/53x/53xA family of devices implements a selectable gain adjustment option. By writing a value to the gain adjust address range, the user can select gain values between 0 and 1.016.

For example, three analog sources to be measured have full-scale outputs of 5.0 V, 4.0 V, and 3.0 V, respectively. Each ADC measurement would ideally use the full dynamic range of the ADC with an internal voltage reference of 1.5 V or 2.2 V (set to 2.2 V for this example). When selecting signal one (5.0 V full-scale), a gain value of 0.44 (5 V full scale * 0.44 = 2.2 V full scale) provides a full-scale signal of 2.2 V when the input signal is 5.0 V. Likewise, a gain value of 0.55 (4 V full scale * 0.55 = 2.2 V full scale) for the second source and 0.73 (3 V full scale * 0.73 = 2.2 V full scale) for the third source provide full-scale ADC0 measurements when the input signal is full-scale.

Additionally, some sensors or other input sources have small part-to-part variations that must be accounted for to achieve accurate results. In this case, the programmable gain value could be used as a calibration value to eliminate these part-to-part variations.

7. Comparator

C8051F52x/F52xA/F53x/F53xA devices include one on-chip programmable voltage comparator. The Comparator is shown in Figure 7.1.

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 or SUSPEND mode. When assigned to a Port pin, the Comparator output may be configured as open drain or push-pull (see Section “13.2. Port I/O Initialization” on page 126). The Comparator may also be used as a reset source (see Section “11.5. Comparator Reset” on page 110).

The Comparator inputs are selected in the CPT0MX register (SFR Definition 7.2). The CMX0P3–CMX0P0 bits select the Comparator0 positive input; the CMX0N3–CMX0N0 bits select the Comparator0 negative input.

Important Note About Comparator Inputs: The Port pins selected as Comparator inputs should be configured as analog inputs in their associated Port configuration register and configured to be skipped by the Crossbar (for details on Port configuration, see Section “13.3. General Purpose Port I/O” on page 128).

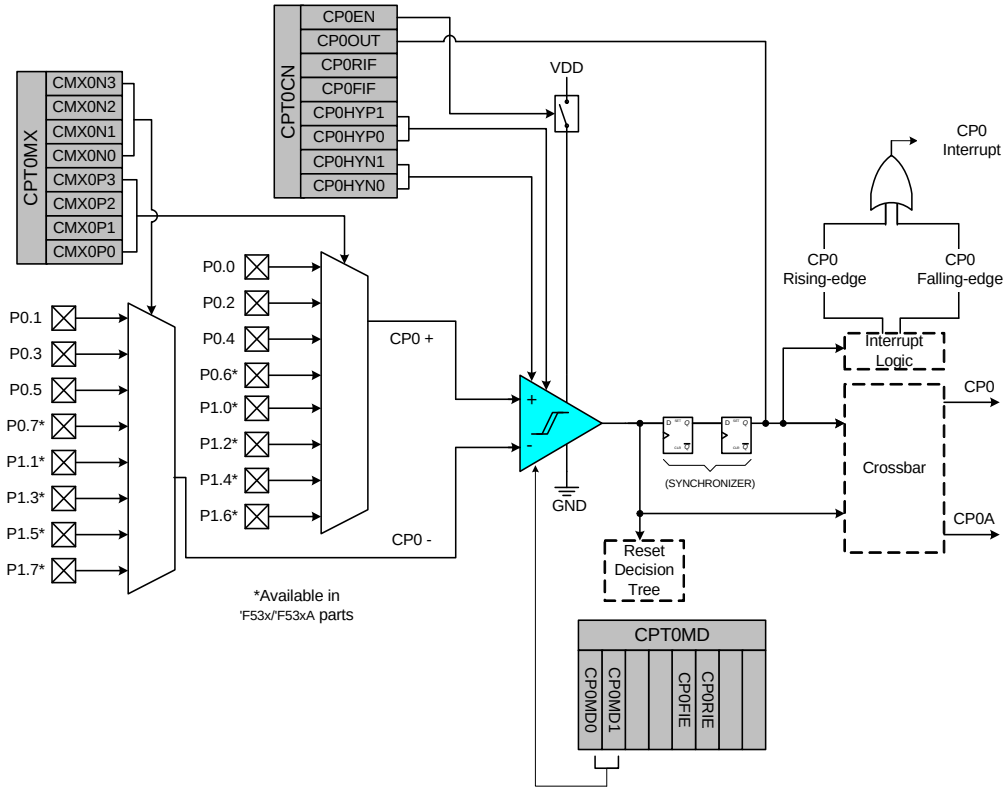

Figure 7.1. Comparator Functional Block Diagram

The Comparator output can be polled in software, used as an interrupt source, internal oscillator suspend awakening 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 or SUSPEND 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 its supply current falls to

SFR Definition 7.3. CPT0MD: Comparator0 Mode Selection

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
Reserved	—	CP0RIE	CP0FIE	—	—	CP0MD1	CP0MD0	00000010
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x9D

Bit7: **RESERVED**. Read = 0b. Must write 0b.
Bit6: **UNUSED**. Read = 0b. Write = don't care.
Bit5: **CP0RIE**: Comparator Rising-Edge Interrupt Enable.
0: Comparator rising-edge interrupt disabled.
1: Comparator rising-edge interrupt enabled.
Bit4: **CP0FIE**: Comparator Falling-Edge Interrupt Enable.
0: Comparator falling-edge interrupt disabled.
1: Comparator falling-edge interrupt enabled.
Note: It is necessary to enable both CP0xIE and the correspondent ECPx bit located in EIE1 SFR.
Bits3–2: **UNUSED**. Read = 00b. Write = don't care.
Bits1–0: **CP0MD1–CP0MD0**: Comparator0 Mode Select
These bits select the response time for Comparator0.

Mode	CP0MD1	CP0MD0	CP0 Falling Edge Response Time (TYP)
0	0	0	Fastest Response Time
1	0	1	—
2	1	0	—
3	1	1	Lowest Power Consumption

Note: Rising Edge response times are approximately double the Falling Edge response times.

10.4. Interrupt Register Descriptions

The SFRs used to enable the interrupt sources and set their priority level are described below. Refer to the data sheet section associated with a particular on-chip peripheral for information regarding valid interrupt conditions for the peripheral and the behavior of its interrupt-pending flag(s).

SFR Definition 10.1. IE: Interrupt Enable

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
EA	ESPI0	ET2	ES0	ET1	EX1	ET0	EX0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Bit Addressable
							SFR Address:	0xA8

Bit7: **EA**: Global Interrupt Enable.
This bit globally enables/disables all interrupts. It overrides the individual interrupt mask settings.
0: Disable all interrupt sources.
1: Enable each interrupt according to its individual mask setting.

Bit6: **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.

Bit5: **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.

Bit4: **ES0**: Enable UART0 Interrupt.
This bit sets the masking of the UART0 interrupt.
0: Disable UART0 interrupt.
1: Enable UART0 interrupt.

Bit3: **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.

Bit2: **EX1**: Enable External Interrupt 1.
This bit sets the masking of the external interrupt 1.
0: Disable external interrupt 1.
1: Enable extern interrupt 1 requests.

Bit1: **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.

Bit0: **EX0**: Enable External Interrupt 0.
This bit sets the masking of the external interrupt 0.
0: Disable external interrupt 0.
1: Enable extern interrupt 0 requests.

SFR Definition 13.13. P0SKIP: Port0 Skip

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xD4

Bits7–0: **P1SKIP[7:0]**: Port1 Crossbar Skip Enable Bits.
These bits select Port pins to be skipped by the Crossbar Decoder. Port pins used as analog inputs (for ADC or Comparator) or used as special functions (V_{REF} input, external oscillator circuit, CNVSTR input) 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.

SFR Definition 13.14. P1MAT: Port1 Match

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xCF

Bits7–0: **P1MAT[7:0]**: Port1 Match Value.
These bits control the value that unmasked P0 Port pins are compared against. A Port Match event is generated if (P1 & P1MASK) does not equal (P1MAT & P1MASK).

SFR Definition 13.15. P1MASK: Port1 Mask

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xBF

Bits7–0: **P1MASK[7:0]**: Port1 Mask Value.
These bits select which Port pins will be compared to the value stored in P1MAT.
0: Corresponding P1.n pin is ignored and cannot cause a Port Match event.
1: Corresponding P1.n pin is compared to the corresponding bit in P1MAT.

Note: The load capacitance depends upon the crystal and the manufacturer. Please 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:

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

For example, a tuning-fork crystal of 32 kHz with a recommended load capacitance of 12.5 pF should use the configuration shown in Figure 14.1, Option 1. With a stray capacitance of 3 pF per pin (6 pF total), the 13 pF capacitors yield an equivalent capacitance of 12.5 pF across the crystal, as shown in Figure 14.2.

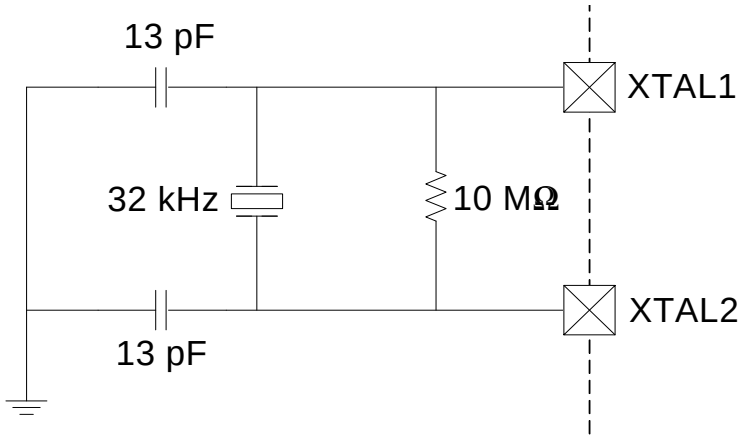

Figure 14.2. 32 kHz External Crystal Example

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.

15.1. 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 15.2), 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.

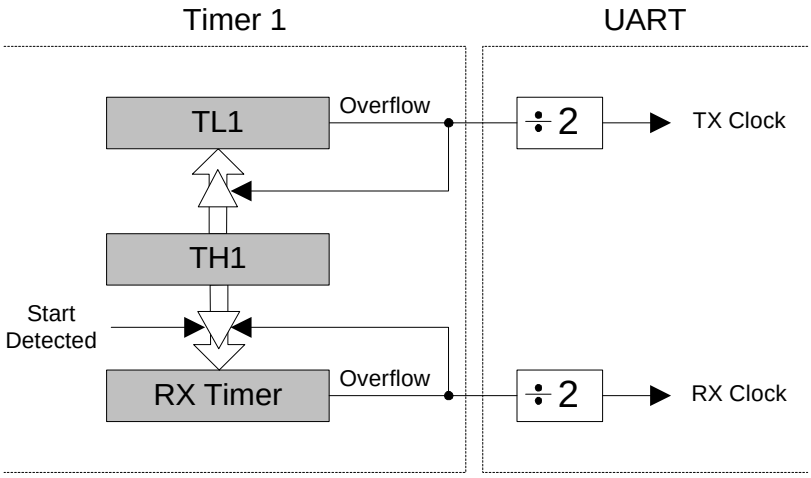

Figure 15.2. UART0 Baud Rate Logic

Timer 1 should be configured for Mode 2, 8-bit auto-reload (see Section “18.1.3. Mode 2: 8-bit Counter/Timer with Auto-Reload” on page 184). 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. The UART0 baud rate is determined by Equation 15.1-A and Equation 15.1-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 15.1. 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 (8-bit auto-reload mode reload value). Timer 1 clock frequency is selected as described in Section “18. Timers” on page 182. A quick reference for typical baud rates and system clock frequencies is given in Table 15.1. Note that the internal oscillator may still generate the system clock when the external oscillator is driving Timer 1.

16.1. Signal Descriptions

The four signals used by SPI0 (MOSI, MISO, SCK, NSS) are described below.

16.1.1. Master Out, Slave In (MOSI)

The master-out, slave-in (MOSI) signal is an output from a master device and an input to slave devices. It is used to serially transfer data from the master to the slave. This signal is an output when SPI0 is operating as a master and an input when SPI0 is operating as a slave. Data is transferred most-significant bit first. When configured as a master, MOSI is driven by the MSB of the shift register in both 3- and 4-wire mode.

16.1.2. Master In, Slave Out (MISO)

The master-in, slave-out (MISO) signal is an output from a slave device and an input to the master device. It is used to serially transfer data from the slave to the master. This signal is an input when SPI0 is operating as a master and an output when SPI0 is operating as a slave. Data is transferred most-significant bit first. The MISO pin is placed in a high-impedance state when the SPI module is disabled and when the SPI operates in 4-wire mode as a slave that is not selected. When acting as a slave in 3-wire mode, MISO is always driven by the MSB of the shift register.

16.1.3. Serial Clock (SCK)

The serial clock (SCK) signal is an output from the master device and an input to slave devices. It is used to synchronize the transfer of data between the master and slave on the MOSI and MISO lines. SPI0 generates this signal when operating as a master. The SCK signal is ignored by a SPI slave when the slave is not selected (NSS = 1) in 4-wire slave mode.

16.1.4. Slave Select (NSS)

The function of the slave-select (NSS) signal is dependent on the setting of the NSSMD1 and NSSMD0 bits in the SPI0CN register. There are three possible modes that can be selected with these bits:

1. NSSMD[1:0] = 00: 3-Wire Master or 3-Wire Slave Mode: SPI0 operates in 3-wire mode, and NSS is disabled. When operating as a slave device, SPI0 is always selected in 3-wire mode. Since no select signal is present, SPI0 must be the only slave on the bus in 3-wire mode. This is intended for point-to-point communication between a master and one slave.
2. NSSMD[1:0] = 01: 4-Wire Slave or Multi-Master Mode: SPI0 operates in 4-wire mode, and NSS is enabled as an input. When operating as a slave, NSS selects the SPI0 device. When operating as a master, a 1-to-0 transition of the NSS signal disables the master function of SPI0 so that multiple master devices can be used on the same SPI bus.
3. NSSMD[1:0] = 1x: 4-Wire Master Mode: SPI0 operates in 4-wire mode, and NSS is enabled as an output. The setting of NSSMD0 determines what logic level the NSS pin will output. This configuration should only be used when operating SPI0 as a master device.

See Figure 16.2, Figure 16.3, and Figure 16.4 for typical connection diagrams of the various operational modes. **Note that the setting of NSSMD bits affects the pinout of the device.** When in 3-wire master or 3-wire slave mode, the NSS pin will not be mapped by the crossbar. In all other modes, the NSS signal will be mapped to a pin on the device. See Section “13. Port Input/Output” on page 120 for general purpose port I/O and crossbar information.

16.5. Serial Clock Timing

Four combinations of serial clock phase and polarity can be selected using the clock control bits in the SPI0 Configuration Register (SPI0CFG). The CKPHA bit (SPI0CFG.5) selects one of two clock phases (edge used to latch the data). The CKPOL bit (SPI0CFG.4) selects between a rising edge or a falling edge. Both master and slave devices must be configured to use the same clock phase and polarity. SPI0 should be disabled (by clearing the SPIEN bit, SPI0CN.0) when changing the clock phase or polarity. The clock and data line relationships are shown in Figure 16.5.

The SPI0 Clock Rate Register (SPI0CKR) as shown in SFR Definition 16.3 controls the master mode serial clock frequency. This register is ignored when operating in slave mode. When the SPI is configured as a master, the maximum data transfer rate (bits/sec) is one-half the system clock frequency or 12.5 MHz, whichever is slower. When the SPI is configured as a slave, the maximum data transfer rate (bits/sec) for full-duplex operation is 1/10 the system clock frequency, provided that the master issues SCK, NSS (in 4-wire slave mode), and the serial input data synchronously with the slave's system clock. If the master issues SCK, NSS, and the serial input data asynchronously, the maximum data transfer rate (bits/sec) must be less than 1/10 the system clock frequency. In the special case where the master only wants to transmit data to the slave and does not need to receive data from the slave (i.e. half-duplex operation), the SPI slave can receive data at a maximum data transfer rate (bits/sec) of 1/4 the system clock frequency. This is provided that the master issues SCK, NSS, and the serial input data synchronously with the slave's system clock.

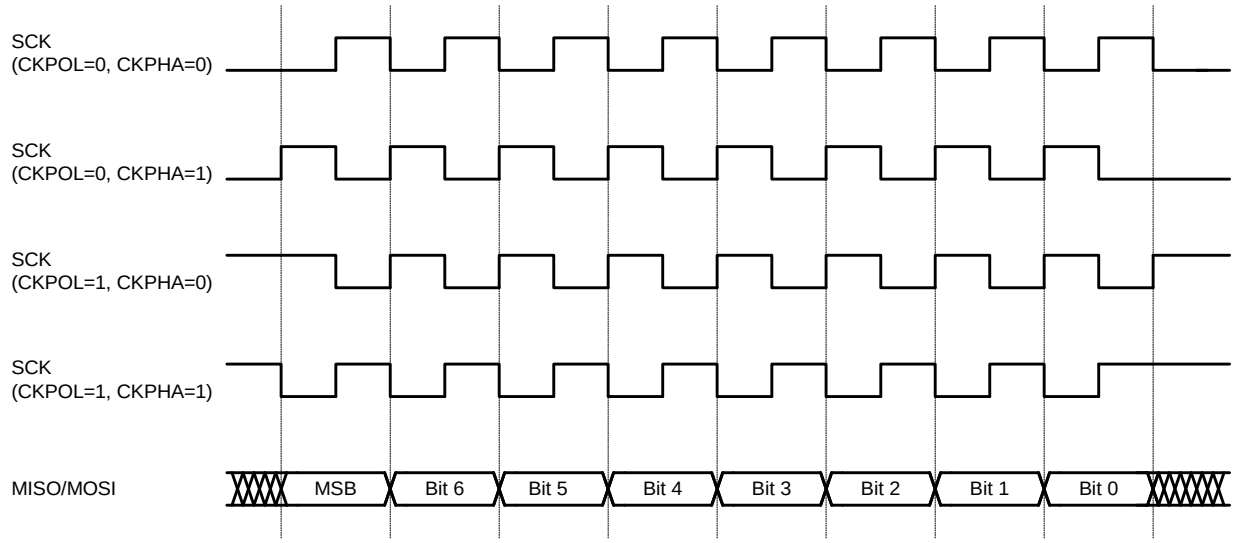

Figure 16.5. Data/Clock Timing Relationship

16.6. SPI Special Function Registers

SPI0 is accessed and controlled through four special function registers in the system controller: SPI0CN Control Register, SPI0DAT Data Register, SPI0CFG Configuration Register, and SPI0CKR Clock Rate Register. The four special function registers related to the operation of the SPI0 Bus are described in the following figures.

SFR Definition 16.4. SPI0DAT: SPI0 Data

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xA3

Bits7–0: **SPI0DAT**: SPI0 Transmit and Receive Data.
The SPI0DAT register is used to transmit and receive SPI0 data. Writing data to SPI0DAT places the data into the transmit buffer and initiates a transfer when in Master Mode. A read of SPI0DAT returns the contents of the receive buffer.

17.4. LIN Slave Mode Operation

When the device is configured for slave mode operation, it must wait for a command from a master node. Access from the firmware to data buffer and ID registers of the LIN peripheral is only possible when a data request is pending (DTREQ bit (LIN0ST.4) is 1) and also when the LIN bus is not active (ACTIVE bit (LIN0ST.7) is set to 0).

The LIN peripheral in slave mode detects the header of the message frame sent by the LIN master. If slave synchronization is enabled (autobaud), the slave synchronizes its internal bit time to the master bit time.

The LIN peripheral configured for slave mode will generated an interrupt in one of three situations:

1. After the reception of the IDENTIFIER FIELD.
2. When an error is detected.
3. When the message transfer is completed.

The application should perform the following steps when an interrupt is detected:

1. Check the status of the DTREQ bit (LIN0ST.4). This bit is set when the IDENTIFIER FIELD has been received.
2. If DTREQ (LIN0ST.4) is set, read the identifier from LIN0ID and process it. If DTREQ (LIN0ST.4) is not set, continue to step 7.
3. Set the TXRX bit (LIN0CTRL.5) to 1 if the current frame is a transmit operation for the slave and set to 0 if the current frame is a receive operation for the slave.
4. Load the data length into LIN0SIZE.
5. For a slave transmit operation, load the data to transmit into the data buffer.
6. Set the DTACK bit (LIN0CTRL.4). Continue to step 10.
7. If DTREQ (LIN0ST.4) is not set, check the DONE bit (LIN0ST.0). The transmission was successful if the DONE bit is set.
8. If the transmission was successful and the current frame was a receive operation for the slave, load the received data bytes from the data buffer.
9. If the transmission was not successful, check LIN0ERR to determine the nature of the error. Further error handling has to be done by the application.
10. Set the RSTINT (LIN0CTRL.3) and RSTERR bits (LIN0CTRL.2) to reset the interrupt request and the error flags.

In addition to these steps, the application should be aware of the following:

1. If the current frame is a transmit operation for the slave, steps 1 through 5 must be completed during the IN-FRAME RESPONSE SPACE. If it is not completed in time, a timeout will be detected by the master.
2. If the current frame is a receive operation for the slave, steps 1 through 5 have to be finished until the reception of the first byte after the IDENTIFIER FIELD. Otherwise, the internal receive buffer of the LIN peripheral will be overwritten and a timeout error will be detected in the LIN peripheral.
3. The LIN module does not directly support LIN Version 1.3 Extended Frames. If the application detects an unknown identifier (e.g. extended identifier), it has to write a 1 to the STOP bit (LIN0CTRL.7) instead of setting the DTACK (LIN0CTRL.4) bit. At that time, steps 2 through 5 can then be skipped. In this situation, the LIN peripheral stops the processing of the LIN communication until the next SYNC BREAK is received.
4. Changing the configuration of the checksum during a transaction will cause the interface to reset and the transaction to be lost. To prevent this, the checksum should not be configured while a transaction is

SFR Definition 18.2. TMOD: Timer Mode

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
GATE1	C/T1	T1M1	T1M0	GATE0	C/T0	T0M1	T0M0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
							SFR Address:	0x89

Bit7: **GATE1**: Timer 1 Gate Control.
0: Timer 1 enabled when TR1 = 1 irrespective of $\overline{\text{INT0}}$ logic level.
1: Timer 1 enabled only when TR1 = 1 AND $\overline{\text{INT0}}$ is active as defined by bit IN1PL in register IT01CF (see SFR Definition 10.5. “IT01CF: INT0/INT1 Configuration” on page 105).

Bit6: **C/T1**: Counter/Timer 1 Select.
0: Timer Function: Timer 1 incremented by clock defined by T1M bit (CKCON.4).
1: Counter Function: Timer 1 incremented by high-to-low transitions on external input pin (T1).

Bits5–4: **T1M1–T1M0**: Timer 1 Mode Select.
These bits select the Timer 1 operation mode.

T1M1	T1M0	Mode
0	0	Mode 0: 13-bit counter/timer
0	1	Mode 1: 16-bit counter/timer
1	0	Mode 2: 8-bit counter/timer with auto-reload
1	1	Mode 3: Timer 1 inactive

Bit3: **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 10.5. “IT01CF: INT0/INT1 Configuration” on page 105).

Bit2: **C/T0**: Counter/Timer Select.
0: Timer Function: Timer 0 incremented by clock defined by T0M bit (CKCON.3).
1: Counter Function: Timer 0 incremented by high-to-low transitions on external input pin (T0).

Bits1–0: **T0M1–T0M0**: Timer 0 Mode Select.
These bits select the Timer 0 operation mode.

T0M1	T0M0	Mode
0	0	Mode 0: 13-bit counter/timer
0	1	Mode 1: 16-bit counter/timer
1	0	Mode 2: 8-bit counter/timer with auto-reload
1	1	Mode 3: Two 8-bit counter/timers

19.2.3. High Speed Output Mode

In High Speed Output mode, a module's associated CEXn pin is toggled each time a match occurs between the PCA Counter and the module's 16-bit capture/compare register (PCA0CPHn and PCA0CPLn) Setting the TOGn, MATn, and ECOMn bits in the PCA0CPMn register enables the High-Speed Output mode.

Important Note About Capture/Compare Registers: When writing a 16-bit value to the PCA0 Capture/Compare registers, the low byte should always be written first. Writing to PCA0CPLn clears the ECOMn bit to 0; writing to PCA0CPHn sets ECOMn to 1.

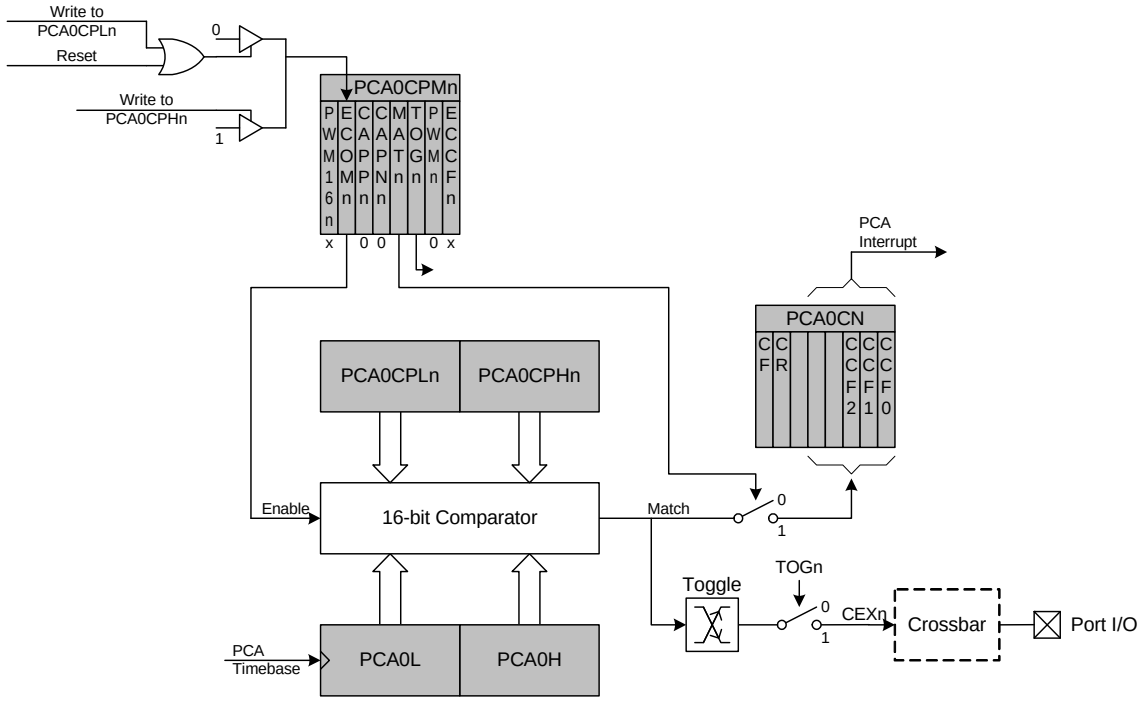

Figure 19.6. PCA High-Speed Output Mode Diagram

Note: The initial state of the Toggle output is logic 1 and is initialized to this state when the module enters High Speed Output Mode.