



Welcome to [E-XFL.COM](https://www.e-xfl.com)

What is "[Embedded - Microcontrollers](#)"?

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Not For New Designs
Core Processor	8051
Core Size	8-Bit
Speed	48MHz
Connectivity	EBI/EMI, SMBus (2-Wire/I ² C), SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	40
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4.25K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 20x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-TQFP
Supplier Device Package	48-TQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f340-gq

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

14.5.1.System Clock Selection	139
14.5.2.USB Clock Selection	139
15.Port Input/Output.....	142
15.1.Priority Crossbar Decoder	144
15.2.Port I/O Initialization	147
15.3.General Purpose Port I/O.....	150
16.Universal Serial Bus Controller (USB0).....	159
16.1.Endpoint Addressing	160
16.2.USB Transceiver	160
16.3.USB Register Access	162
16.4.USB Clock Configuration.....	166
16.5.FIFO Management	167
16.5.1.FIFO Split Mode	167
16.5.2.FIFO Double Buffering	168
16.5.3.FIFO Access	168
16.6.Function Addressing.....	169
16.7.Function Configuration and Control.....	169
16.8.Interrupts	172
16.9.The Serial Interface Engine	176
16.10.Endpoint0	176
16.10.1.Endpoint0 SETUP Transactions	177
16.10.2.Endpoint0 IN Transactions.....	177
16.10.3.Endpoint0 OUT Transactions.....	178
16.11.Configuring Endpoints1-3.....	180
16.12.Controlling Endpoints1-3 IN.....	180
16.12.1.Endpoints1-3 IN Interrupt or Bulk Mode.....	180
16.12.2.Endpoints1-3 IN Isochronous Mode.....	181
16.13.Controlling Endpoints1-3 OUT.....	183
16.13.1.Endpoints1-3 OUT Interrupt or Bulk Mode.....	183
16.13.2.Endpoints1-3 OUT Isochronous Mode.....	184
17.SMBus	188
17.1.Supporting Documents.....	189
17.2.SMBus Configuration.....	189
17.3.SMBus Operation	189
17.3.1.Arbitration.....	190
17.3.2.Clock Low Extension.....	191
17.3.3.SCL Low Timeout.....	191
17.3.4.SCL High (SMBus Free) Timeout	191
17.4.Using the SMBus.....	191
17.4.1.SMBus Configuration Register.....	192
17.4.2.SMB0CN Control Register	195
17.4.3.Data Register	198
17.5.SMBus Transfer Modes.....	198
17.5.1.Master Transmitter Mode	198
17.5.2.Master Receiver Mode	200

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

1. System Overview

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D devices are fully integrated mixed-signal System-on-a-Chip MCUs. Highlighted features are listed below. Refer to Table 1.1 for specific product feature selection.

- High-speed pipelined 8051-compatible microcontroller core (up to 48 MIPS)
- In-system, full-speed, non-intrusive debug interface (on-chip)
- Universal Serial Bus (USB) Function Controller with eight flexible endpoint pipes, integrated transceiver, and 1 kB FIFO RAM
- Supply Voltage Regulator
- True 10-bit 200 ksps differential / single-ended ADC with analog multiplexer
- On-chip Voltage Reference and Temperature Sensor
- On-chip Voltage Comparators (2)
- Precision internal calibrated 12 MHz internal oscillator and 4x clock multiplier
- Internal low-frequency oscillator for additional power savings
- Up to 64 kB of on-chip Flash memory
- Up to 4352 Bytes of on-chip RAM (256 + 4 kB)
- External Memory Interface (EMIF) available on 48-pin versions.
- SMBus/I2C, up to 2 UARTs, and Enhanced SPI serial interfaces implemented in hardware
- Four general-purpose 16-bit timers
- Programmable Counter/Timer Array (PCA) with five capture/compare modules and Watchdog Timer function
- On-chip Power-On Reset, V_{DD} Monitor, and Missing Clock Detector
- Up to 40 Port I/O (5 V tolerant)

With on-chip Power-On Reset, V_{DD} monitor, Voltage Regulator, Watchdog Timer, and clock oscillator, C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D devices are truly stand-alone System-on-a-Chip solutions. The Flash memory can be reprogrammed in-circuit, providing non-volatile data storage, and also allowing field upgrades of the 8051 firmware. User software has complete control of all peripherals, and may individually shut down any or all peripherals for power savings.

The on-chip Silicon Labs 2-Wire (C2) Development Interface allows non-intrusive (uses no on-chip resources), full speed, in-circuit debugging using the production MCU installed in the final application. This debug logic supports inspection and modification of memory and registers, setting breakpoints, single stepping, run and halt commands. All analog and digital peripherals are fully functional while debugging using C2. The two C2 interface pins can be shared with user functions, allowing in-system debugging without occupying package pins.

Each device is specified for 2.7–5.25 V operation over the industrial temperature range (–40 to +85 °C). For voltages above 3.6 V, the on-chip Voltage Regulator must be used. A minimum of 3.0 V is required for USB communication. The Port I/O and RST pins are tolerant of input signals up to 5 V. C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D devices are available in 48-pin TQFP, 32-pin LQFP, or 32-pin QFN packages. See Table 1.1, “Product Selection Guide,” on page 18 for feature and package choices.

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

Table 1.1. Product Selection Guide

Ordering Part Number	MIPS (Peak)	Flash Memory (Bytes)	RAM	Calibrated Internal Oscillator	Low Frequency Oscillator	USB with 1k Endpoint RAM	Supply Voltage Regulator	SMBus/I2C	Enhanced SPI	UARTs	Timers (16-bit)	Programmable Counter Array	Digital Port I/Os	External Memory Interface (EMIF)	10-bit 200 kpsps ADC	Temperature Sensor	Voltage Reference	Analog Comparators	Package
C8051F340-GQ	48	64k	4352	✓	✓	✓	✓	✓	✓	2	4	✓	40	✓	✓	✓	✓	2	TQFP48
C8051F341-GQ	48	32k	2304	✓	✓	✓	✓	✓	✓	2	4	✓	40	✓	✓	✓	✓	2	TQFP48
C8051F342-GQ	48	64k	4352	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	LQFP32
C8051F342-GM	48	64k	4352	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	QFN32
C8051F343-GQ	48	32k	2304	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	LQFP32
C8051F343-GM	48	32k	2304	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	QFN32
C8051F344-GQ	25	64k	4352	✓	✓	✓	✓	✓	✓	2	4	✓	40	✓	✓	✓	✓	2	TQFP48
C8051F345-GQ	25	32k	2304	✓	✓	✓	✓	✓	✓	2	4	✓	40	✓	✓	✓	✓	2	TQFP48
C8051F346-GQ	25	64k	4352	✓	—	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	LQFP32
C8051F346-GM	25	64k	4352	✓	—	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	QFN32
C8051F347-GQ	25	32k	2304	✓	—	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	LQFP32
C8051F347-GM	25	32k	2304	✓	—	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	QFN32
C8051F348-GQ	25	32k	2304	✓	✓	✓	✓	✓	✓	2	4	✓	40	✓	—	—	—	2	TQFP48
C8051F349-GQ	25	32k	2304	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	—	—	—	2	LQFP32
C8051F349-GM	25	32k	2304	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	—	—	—	2	QFN32
C8051F34A-GQ	48	64k	4352	✓	✓	✓	✓	✓	✓	2	4	✓	25	—	✓	✓	✓	2	LQFP32
C8051F34A-GM	48	64k	4352	✓	✓	✓	✓	✓	✓	2	4	✓	25	—	✓	✓	✓	2	QFN32
C8051F34B-GQ	48	32k	2304	✓	✓	✓	✓	✓	✓	2	4	✓	25	—	✓	✓	✓	2	LQFP32
C8051F34B-GM	48	32k	2304	✓	✓	✓	✓	✓	✓	2	4	✓	25	—	✓	✓	✓	2	QFN32
C8051F34C-GQ	48	64k	4352	✓	✓	✓	✓	✓	✓	2	4	✓	40	✓	—	—	—	2	TQFP48
C8051F34D-GQ	48	64k	4352	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	—	—	—	2	LQFP32

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

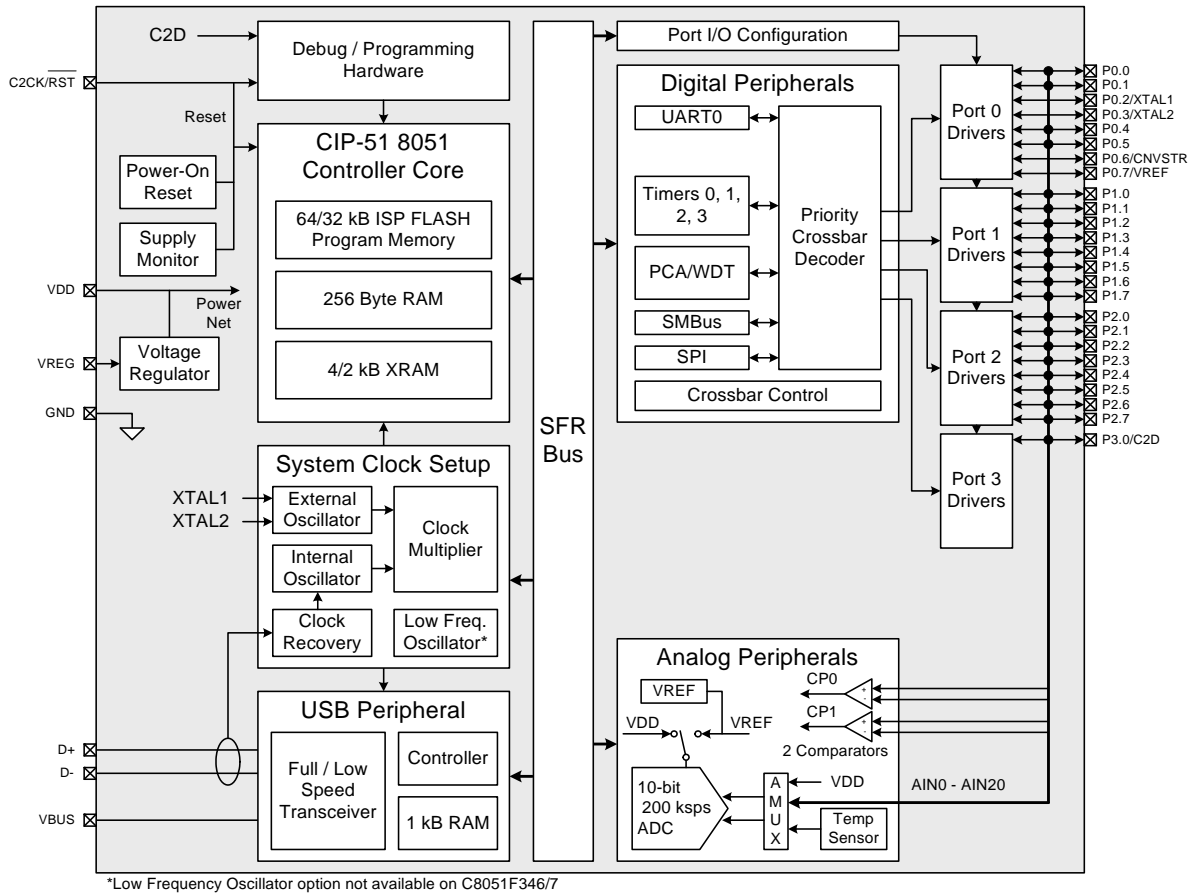


Figure 1.2. C8051F342/3/6/7 Block Diagram

14.3.3. External RC Example

If an RC network is used as an external oscillator source for the MCU, the circuit should be configured as shown in Figure 14.1, Option 2. The capacitor should be no greater than 100 pF; however for very small capacitors, the total capacitance may be dominated by parasitic capacitance in the PCB layout. To determine the required External Oscillator Frequency Control value (XFCN) in the OSCXCN Register, first select the RC network value to produce the desired frequency of oscillation. If the frequency desired is 100 kHz, let R = 246 kΩ and C = 50 pF:

$$f = \frac{1.23(10^3)}{RC} = \frac{1.23(10^3)}{[246 \times 50]} = 0.1 \text{ MHz} = 100 \text{ kHz}$$

Referring to the table in SFR Definition 14.4, the required XFCN setting is 010b. Programming XFCN to a higher setting in RC mode will improve frequency accuracy at an increased external oscillator supply current.

14.3.4. External Capacitor Example

If a capacitor is used as an external oscillator for the MCU, the circuit should be configured as shown in Figure 14.1, Option 3. The capacitor should be no greater than 100 pF; however for very small capacitors, the total capacitance may be dominated by parasitic capacitance in the PCB layout. To determine the required External Oscillator Frequency Control value (XFCN) in the OSCXCN Register, select the capacitor to be used and find the frequency of oscillation from the equations below. Assume $V_{DD} = 3.0 \text{ V}$ and $C = 50 \text{ pF}$:

$$f = \frac{KF}{(C \times V_{DD})} = \frac{KF}{(50 \times 3)\text{MHz}}$$

$$f = \frac{KF}{150 \text{ MHz}}$$

If a frequency of roughly 150 kHz is desired, select the K Factor from the table in SFR Definition 14.4 as $KF = 22$:

$$f = \frac{22}{150} = 0.146 \text{ MHz, or } 146 \text{ kHz}$$

Therefore, the XFCN value to use in this example is 011b.

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

SFR Definition 15.8. P1: Port1 Latch

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: (bit addressable) 0x90

Bits7–0: P1.[7:0]
 Write - Output appears on I/O pins per Crossbar Registers (when XBARE = '1').
 0: Logic Low Output.
 1: Logic High Output (high impedance if corresponding P1MDOUT.n bit = 0).
 Read - Always reads '0' if selected as analog input in register P1MDIN. Directly reads Port pin when configured as digital input.
 0: P1.n pin is logic low.
 1: P1.n pin is logic high.

SFR Definition 15.9. P1MDIN: Port1 Input Mode

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: 0xF2

Bits7–0: Analog Input Configuration Bits for P1.7–P1.0 (respectively).
 Port pins configured as analog inputs have their weak pull-up, digital driver, and digital receiver disabled.
 0: Corresponding P1.n pin is configured as an analog input.
 1: Corresponding P1.n pin is not configured as an analog input.

SFR Definition 15.10. P1MDOUT: Port1 Output Mode

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: 0xA5

Bits7–0: Output Configuration Bits for P1.7–P1.0 (respectively): ignored if corresponding bit in register P1MDIN is logic 0.
 0: Corresponding P1.n Output is open-drain.
 1: Corresponding P1.n Output is push-pull.

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

SFR Definition 15.21. P4MDIN: Port4 Input Mode

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: 0xF5

Bits7–0: Analog Input Configuration Bits for P4.7–P4.0 (respectively).
Port pins configured as analog inputs have their weak pull-up, digital driver, and digital receiver disabled.
0: Corresponding P4.n pin is configured as an analog input.
1: Corresponding P4.n pin is not configured as an analog input.

Note: P4 is only available on 48-pin devices.

SFR Definition 15.22. P4MDOUT: Port4 Output Mode

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: 0xAE

Bits7–0: Output Configuration Bits for P4.7–P4.0 (respectively); ignored if corresponding bit in register P4MDIN is logic 0.
0: Corresponding P4.n Output is open-drain.
1: Corresponding P4.n Output is push-pull.

Note: P4 is only available on 48-pin devices.

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

16.1. Endpoint Addressing

A total of eight endpoint pipes are available. The control endpoint (Endpoint0) always functions as a bi-directional IN/OUT endpoint. The other endpoints are implemented as three pairs of IN/OUT endpoint pipes:

Table 16.1. Endpoint Addressing Scheme

Endpoint	Associated Pipes	USB Protocol Address
Endpoint0	Endpoint0 IN	0x00
	Endpoint0 OUT	0x00
Endpoint1	Endpoint1 IN	0x81
	Endpoint1 OUT	0x01
Endpoint2	Endpoint2 IN	0x82
	Endpoint2 OUT	0x02
Endpoint3	Endpoint3 IN	0x83
	Endpoint3 OUT	0x03

16.2. USB Transceiver

The USB Transceiver is configured via the USB0XCN register shown in SFR Definition 16.1. This configuration includes Transceiver enable/disable, pull-up resistor enable/disable, and device speed selection (Full or Low Speed). When bit SPEED = '1', USB0 operates as a Full Speed USB function, and the on-chip pull-up resistor (if enabled) appears on the D+ pin. When bit SPEED = '0', USB0 operates as a Low Speed USB function, and the on-chip pull-up resistor (if enabled) appears on the D- pin. Bits4-0 of register USB0XCN can be used for Transceiver testing as described in SFR Definition 16.1. The pull-up resistor is enabled only when VBUS is present (see **Section "8.2. VBUS Detection" on page 69** for details on VBUS detection).

Important Note: The USB clock should be active before the Transceiver is enabled.

17.3.2. Clock Low Extension

SMBus provides a clock synchronization mechanism, similar to I2C, which allows devices with different speed capabilities to coexist on the bus. A clock-low extension is used during a transfer in order to allow slower slave devices to communicate with faster masters. The slave may temporarily hold the SCL line LOW to extend the clock low period, effectively decreasing the serial clock frequency.

17.3.3. SCL Low Timeout

If the SCL line is held low by a slave device on the bus, no further communication is possible. Furthermore, the master cannot force the SCL line high to correct the error condition. To solve this problem, the SMBus protocol specifies that devices participating in a transfer must detect any clock cycle held low longer than 25 ms as a “timeout” condition. Devices that have detected the timeout condition must reset the communication no later than 10 ms after detecting the timeout condition.

When the SMBTOE bit in SMB0CF is set, Timer 3 is used to detect SCL low timeouts. Timer 3 is forced to reload when SCL is high, and allowed to count when SCL is low. With Timer 3 enabled and configured to overflow after 25 ms (and SMBTOE set), the Timer 3 interrupt service routine can be used to reset (disable and re-enable) the SMBus in the event of an SCL low timeout.

17.3.4. SCL High (SMBus Free) Timeout

The SMBus specification stipulates that if the SCL and SDA lines remain high for more than 50 μ s, the bus is designated as free. When the SMBFTE bit in SMB0CF is set, the bus will be considered free if SCL and SDA remain high for more than 10 SMBus clock source periods. If the SMBus is waiting to generate a Master START, the START will be generated following this timeout. Note that a clock source is required for free timeout detection, even in a slave-only implementation.

17.4. Using the SMBus

The SMBus can operate in both Master and Slave modes. The interface provides timing and shifting control for serial transfers; higher level protocol is determined by user software. The SMBus interface provides the following application-independent features:

- Byte-wise serial data transfers
- Clock signal generation on SCL (Master Mode only) and SDA data synchronization
- Timeout/bus error recognition, as defined by the SMB0CF configuration register
- START/STOP timing, detection, and generation
- Bus arbitration
- Interrupt generation
- Status information

SMBus interrupts are generated for each data byte or slave address that is transferred. When transmitting, this interrupt is generated after the ACK cycle so that software may read the received ACK value; when receiving data, this interrupt is generated before the ACK cycle so that software may define the outgoing ACK value. See **Section “17.5. SMBus Transfer Modes” on page 198** for more details on transmission sequences.

Interrupts are also generated to indicate the beginning of a transfer when a master (START generated), or the end of a transfer when a slave (STOP detected). Software should read the SMB0CN (SMBus Control register) to find the cause of the SMBus interrupt. The SMB0CN register is described in **Section “17.4.2. SMB0CN Control Register” on page 195**; Table 17.4 provides a quick SMB0CN decoding reference.

17.5.2. Master Receiver Mode

Serial data is received on SDA while the serial clock is output on SCL. The SMBus interface generates the START condition and transmits the first byte containing the address of the target slave and the data direction bit. In this case the data direction bit (R/W) will be logic 1 (READ). Serial data is then received from the slave on SDA while the SMBus outputs the serial clock. The slave transmits one or more bytes of serial data. After each byte is received, ACKRQ is set to '1' and an interrupt is generated. Software must write the ACK bit (SMB0CN.1) to define the outgoing acknowledge value (Note: writing a '1' to the ACK bit generates an ACK; writing a '0' generates a NACK). Software should write a '0' to the ACK bit after the last byte is received, to transmit a NACK. The interface exits Master Receiver Mode after the STO bit is set and a STOP is generated. Note that the interface will switch to Master Transmitter Mode if SMB0DAT is written while an active Master Receiver. Figure 17.6 shows a typical Master Receiver sequence. Two received data bytes are shown, though any number of bytes may be received. Notice that the 'data byte transferred' interrupts occur **before** the ACK cycle in this mode.

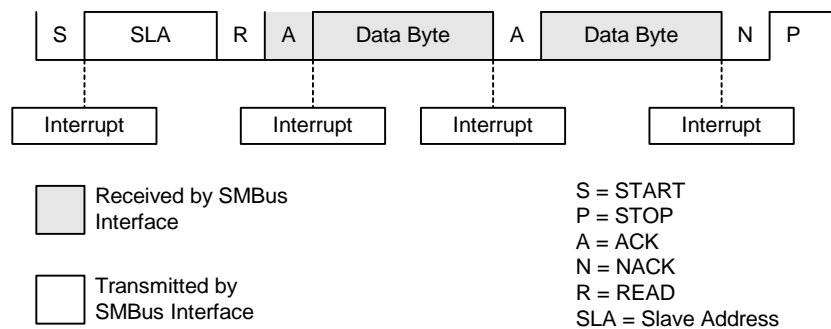
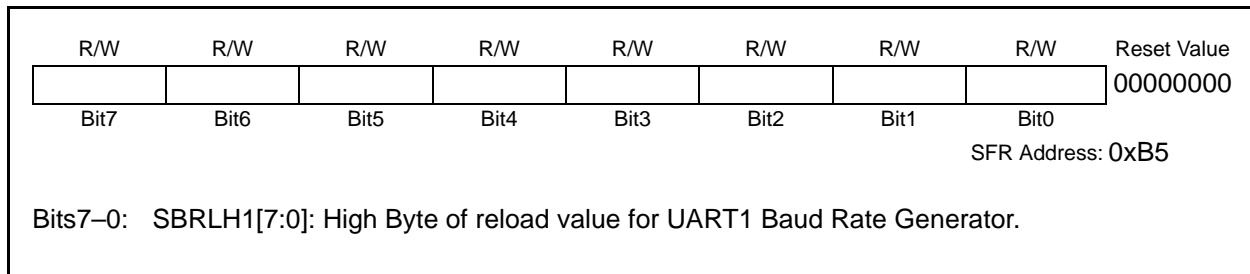


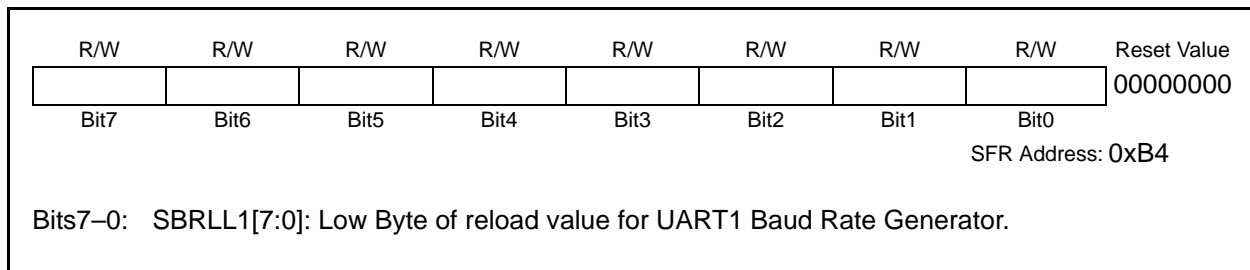
Figure 17.6. Typical Master Receiver Sequence

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

SFR Definition 19.5. SBRLH1: UART1 Baud Rate Generator High Byte



SFR Definition 19.6. SBRL1: UART1 Baud Rate Generator Low Byte



C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

The C/T0 bit (TMOD.2) selects the counter/timer's clock source. When C/T0 is set to logic 1, high-to-low transitions at the selected Timer 0 input pin (T0) increment the timer register (Refer to **Section “15.1. Priority Crossbar Decoder” on page 144** for information on selecting and configuring external I/O pins). Clearing C/T selects the clock defined by the TOM bit (CKCON.3). When TOM is set, Timer 0 is clocked by the system clock. When TOM is cleared, Timer 0 is clocked by the source selected by the Clock Scale bits in CKCON (see SFR Definition 21.3).

Setting the TR0 bit (TCON.4) enables the timer when either GATE0 (TMOD.3) is logic 0 or the input signal $\overline{\text{INT0}}$ is active as defined by bit IN0PL in register INT01CF (see SFR Definition 9.13). Setting GATE0 to ‘1’ allows the timer to be controlled by the external input signal $\overline{\text{INT0}}$ (see **Section “9.3.5. Interrupt Register Descriptions” on page 90**), facilitating pulse width measurements.

TR0	GATE0	INT0	Counter/Timer
0	X	X	Disabled
1	0	X	Enabled
1	1	0	Disabled
1	1	1	Enabled

X = Don't Care

Setting TR0 does not force the timer to reset. The timer registers should be loaded with the desired initial value before the timer is enabled.

TL1 and TH1 form the 13-bit register for Timer 1 in the same manner as described above for TL0 and TH0. Timer 1 is configured and controlled using the relevant TCON and TMOD bits just as with Timer 0. The input signal INT1 is used with Timer 1; the INT1 polarity is defined by bit IN1PL in register INT01CF (see SFR Definition 9.13).

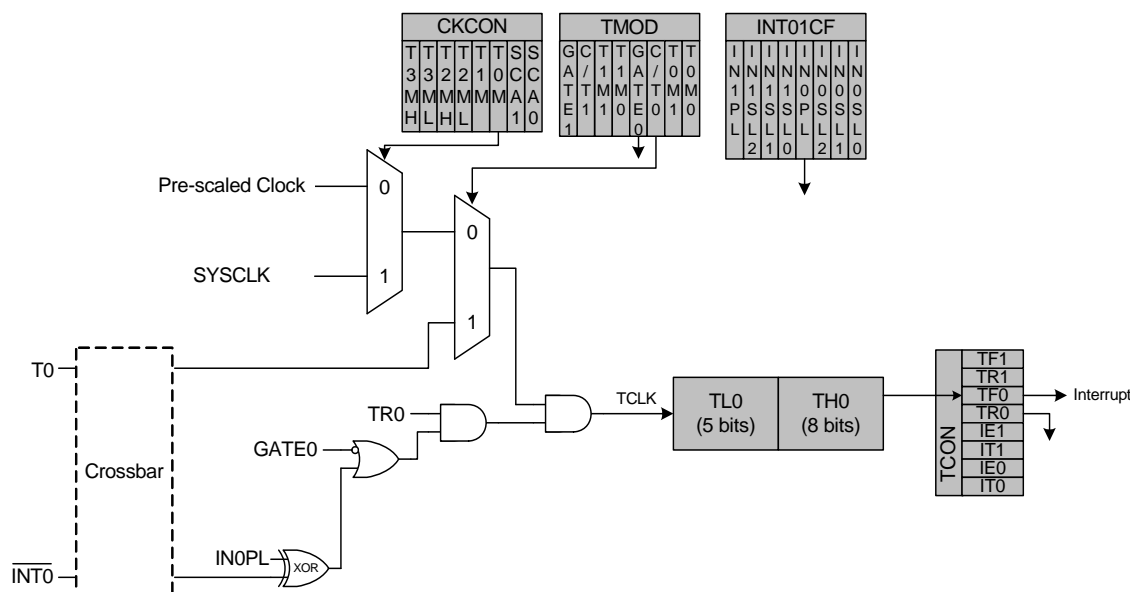


Figure 21.1. T0 Mode 0 Block Diagram

21.1.2. Mode 1: 16-bit Counter/Timer

Mode 1 operation is the same as Mode 0, except that the counter/timer registers use all 16 bits. The counter/timers are enabled and configured in Mode 1 in the same manner as for Mode 0.

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

SFR Definition 21.3. CKCON: Clock Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
T3MH	T3ML	T2MH	T2ML	T1M	T0M	SCA1	SCA0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x8E

Bit7: T3MH: Timer 3 High Byte Clock Select.
This bit selects the clock supplied to the Timer 3 high byte if Timer 3 is configured in split 8-bit timer mode. T3MH is ignored if Timer 3 is in any other mode.
0: Timer 3 high byte uses the clock defined by the T3XCLK bit in TMR3CN.
1: Timer 3 high byte uses the system clock.

Bit6: T3ML: Timer 3 Low Byte Clock Select.
This bit selects the clock supplied to Timer 3. If Timer 3 is configured in split 8-bit timer mode, this bit selects the clock supplied to the lower 8-bit timer.
0: Timer 3 low byte uses the clock defined by the T3XCLK bit in TMR3CN.
1: Timer 3 low byte uses the system clock.

Bit5: T2MH: Timer 2 High Byte Clock Select.
This bit selects the clock supplied to the Timer 2 high byte if Timer 2 is configured in split 8-bit timer mode. T2MH is ignored if Timer 2 is in any other mode.
0: Timer 2 high byte uses the clock defined by the T2XCLK bit in TMR2CN.
1: Timer 2 high byte uses the system clock.

Bit4: T2ML: Timer 2 Low Byte Clock Select.
This bit selects the clock supplied to Timer 2. If Timer 2 is configured in split 8-bit timer mode, this bit selects the clock supplied to the lower 8-bit timer.
0: Timer 2 low byte uses the clock defined by the T2XCLK bit in TMR2CN.
1: Timer 2 low byte uses the system clock.

Bit3: T1M: Timer 1 Clock Select.
This select the clock source supplied to Timer 1. T1M is ignored when C/T1 is set to logic 1.
0: Timer 1 uses the clock defined by the prescale bits, SCA1-SCA0.
1: Timer 1 uses the system clock.

Bit2: T0M: Timer 0 Clock Select.
This bit selects the clock source supplied to Timer 0. T0M is ignored when C/T0 is set to logic 1.
0: Counter/Timer 0 uses the clock defined by the prescale bits, SCA1-SCA0.
1: Counter/Timer 0 uses the system clock.

Bits1–0: SCA1-SCA0: Timer 0/1 Prescale Bits.
These bits control the division of the clock supplied to Timer 0 and/or Timer 1 if configured to use prescaled clock inputs.

SCA1	SCA0	Prescaled Clock
0	0	System clock divided by 12
0	1	System clock divided by 4
1	0	System clock divided by 48
1	1	External clock divided by 8

Note: External clock divided by 8 is synchronized with the system clock.

21.2.3. Timer 2 Capture Modes: USB Start-of-Frame or LFO Falling Edge

When T2CE = '1', Timer 2 will operate in one of two special capture modes. The capture event can be selected between a USB Start-of-Frame (SOF) capture, and a Low-Frequency Oscillator (LFO) Falling Edge capture, using the T2CSS bit. The USB SOF capture mode can be used to calibrate the system clock or external oscillator against the known USB host SOF clock. The LFO falling-edge capture mode can be used to calibrate the internal Low-Frequency Oscillator against the internal High-Frequency Oscillator or an external clock source. When T2SPLIT = '0', Timer 2 counts up and overflows from 0xFFFF to 0x0000. Each time a capture event is received, the contents of the Timer 2 registers (TMR2H:TMR2L) are latched into the Timer 2 Reload registers (TMR2RLH:TMR2RLL). A Timer 2 interrupt is generated if enabled.

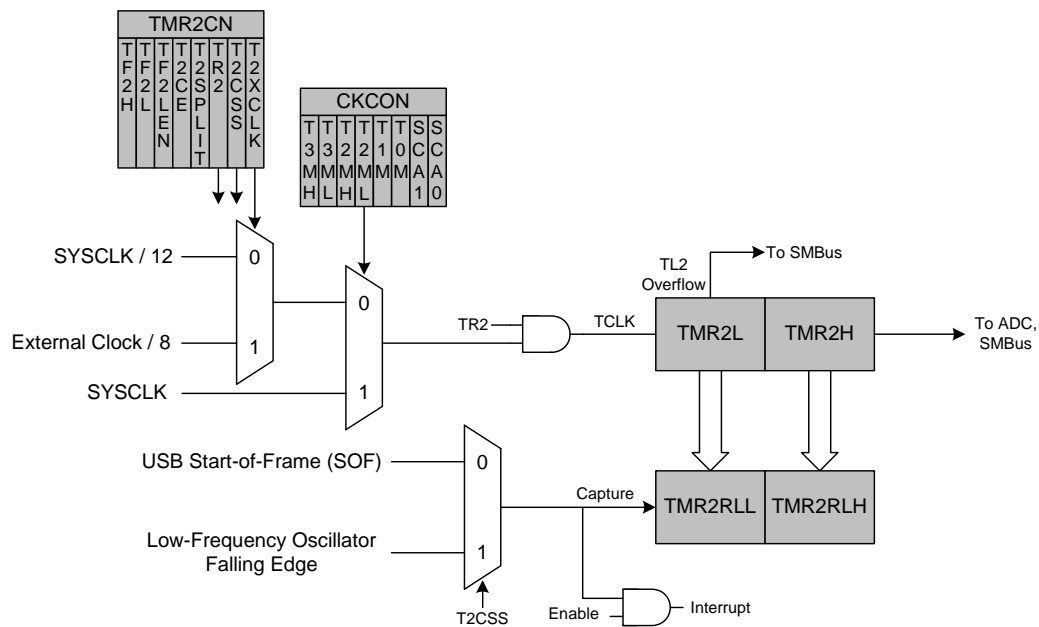


Figure 21.6. Timer 2 Capture Mode (T2SPLIT = '0')

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

SFR Definition 21.8. TMR2CN: Timer 2 Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
TF2H	TF2L	TF2LEN	T2CE	T2SPLIT	TR2	T2CSS	T2XCLK	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: (bit addressable) 0xC8

- Bit7:** TF2H: Timer 2 High Byte Overflow Flag.
Set by hardware when the Timer 2 high byte overflows from 0xFF to 0x00. In 16 bit mode, this will occur when Timer 2 overflows from 0xFFFF to 0x0000. When the Timer 2 interrupt is enabled, setting this bit causes the CPU to vector to the Timer 2 interrupt service routine. TF2H is not automatically cleared by hardware and must be cleared by software.
- Bit6:** TF2L: Timer 2 Low Byte Overflow Flag.
Set by hardware when the Timer 2 low byte overflows from 0xFF to 0x00. When this bit is set, an interrupt will be generated if TF2LEN is set and Timer 2 interrupts are enabled. TF2L will set when the low byte overflows regardless of the Timer 2 mode. This bit is not automatically cleared by hardware.
- Bit5:** TF2LEN: Timer 2 Low Byte Interrupt Enable.
This bit enables/disables Timer 2 Low Byte interrupts. If TF2LEN is set and Timer 2 interrupts are enabled, an interrupt will be generated when the low byte of Timer 2 overflows.
0: Timer 2 Low Byte interrupts disabled.
1: Timer 2 Low Byte interrupts enabled.
- Bit4:** T2CE: Timer 2 Capture Enable
0: Capture function disabled.
1: Capture function enabled. The timer is in capture mode, with the capture event selected by bit T2CSS. Each time a capture event is received, the contents of the Timer 2 registers (TMR2H and TMR2L) are latched into the Timer 2 reload registers (TMR2RLH and TMR2RLH), and a Timer 2 interrupt is generated (if enabled).
- Bit3:** T2SPLIT: Timer 2 Split Mode Enable.
When this bit is set, Timer 2 operates as two 8-bit timers with auto-reload.
0: Timer 2 operates in 16-bit auto-reload mode.
1: Timer 2 operates as two 8-bit auto-reload timers.
- Bit2:** TR2: Timer 2 Run Control.
This bit enables/disables Timer 2. In 8-bit mode, this bit enables/disables TMR2H only; TMR2L is always enabled in this mode.
0: Timer 2 disabled.
1: Timer 2 enabled.
- Bit1:** T2CSS: Timer 2 Capture Source Select.
This bit selects the source of a capture event when bit T2CE is set to '1'.
0: Capture source is USB SOF event.
1: Capture source is falling edge of Low-Frequency Oscillator.
- Bit0:** T2XCLK: Timer 2 External Clock Select.
This bit selects the external clock source for Timer 2. If Timer 2 is in 8-bit mode, this bit selects the external oscillator clock source for both timer bytes. However, the Timer 2 Clock Select bits (T2MH and T2ML in register CKCON) may still be used to select between the external clock and the system clock for either timer.
0: Timer 2 external clock selection is the system clock divided by 12.
1: Timer 2 external clock selection is the external clock divided by 8. Note that the external oscillator source divided by 8 is synchronized with the system clock.

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

21.3.2. 8-bit Timers with Auto-Reload

When T3SPLIT is '1' and T3CE = '0', Timer 3 operates as two 8-bit timers (TMR3H and TMR3L). Both 8-bit timers operate in auto-reload mode as shown in Figure 21.5. TMR3RLL holds the reload value for TMR3L; TMR3RLH holds the reload value for TMR3H. The TR3 bit in TMR3CN handles the run control for TMR3H. TMR3L is always running when configured for 8-bit Mode.

Each 8-bit timer may be configured to use SYSCLK, SYSCLK divided by 12, or the external oscillator clock source divided by 8. The Timer 3 Clock Select bits (T3MH and T3ML in CKCON) select either SYSCLK or the clock defined by the Timer 3 External Clock Select bit (T3XCLK in TMR3CN), as follows:

T3MH	T3XCLK	TMR3H Clock Source
0	0	SYSCLK / 12
0	1	External Clock / 8
1	X	SYSCLK

T3ML	T3XCLK	TMR3L Clock Source
0	0	SYSCLK / 12
0	1	External Clock / 8
1	X	SYSCLK

The TF3H bit is set when TMR3H overflows from 0xFF to 0x00; the TF3L bit is set when TMR3L overflows from 0xFF to 0x00. When Timer 3 interrupts are enabled, an interrupt is generated each time TMR3H overflows. If Timer 3 interrupts are enabled and TF3LEN (TMR3CN.5) is set, an interrupt is generated each time either TMR3L or TMR3H overflows. When TF3LEN is enabled, software must check the TF3H and TF3L flags to determine the source of the Timer 3 interrupt. The TF3H and TF3L interrupt flags are not cleared by hardware and must be manually cleared by software.

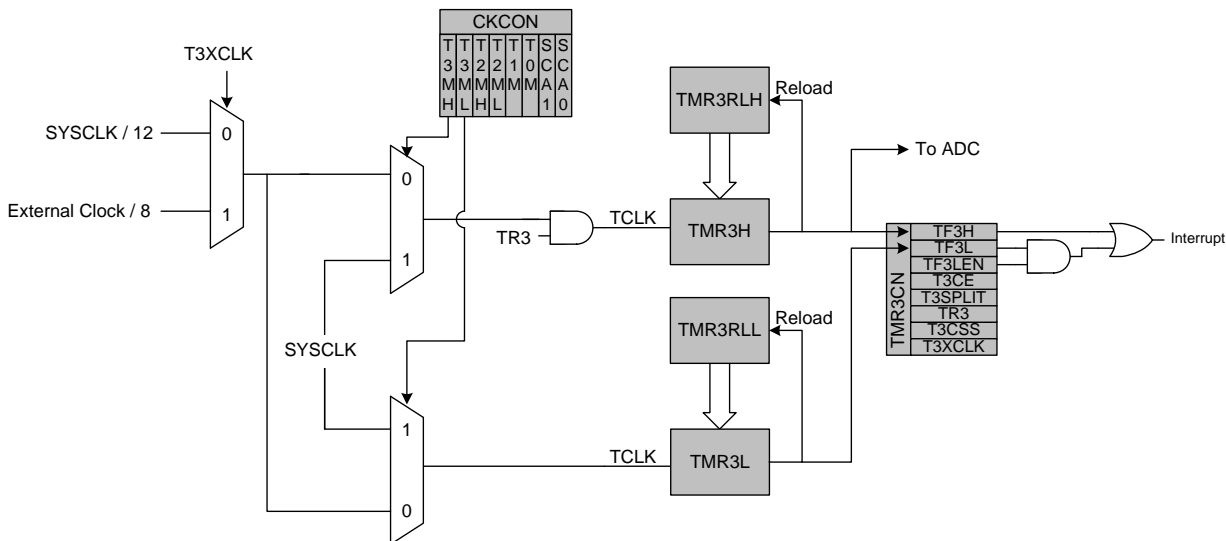


Figure 21.9. Timer 3 8-Bit Mode Block Diagram

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

22.1. PCA Counter/Timer

The 16-bit PCA counter/timer consists of two 8-bit SFRs: PCA0L and PCA0H. PCA0H is the high byte (MSB) of the 16-bit counter/timer and PCA0L is the low byte (LSB). Reading PCA0L automatically latches the value of PCA0H into a “snapshot” register; the following PCA0H read accesses this “snapshot” register. **Reading the PCA0L Register first guarantees an accurate reading of the entire 16-bit PCA0 counter.** Reading PCA0H or PCA0L does not disturb the counter operation. The CPS2-CPS0 bits in the PCA0MD register select the timebase for the counter/timer as shown in Table 22.1.

When the counter/timer overflows from 0xFFFF to 0x0000, the Counter Overflow Flag (CF) in PCA0MD is set to logic 1 and an interrupt request is generated if CF interrupts are enabled. Setting the ECF bit in PCA0MD to logic 1 enables the CF flag to generate an interrupt request. The CF bit is not automatically cleared by hardware when the CPU vectors to the interrupt service routine, and must be cleared by software (Note: PCA0 interrupts must be globally enabled before CF interrupts are recognized. PCA0 interrupts are globally enabled by setting the EA bit (IE.7) and the EPCA0 bit in EIE1 to logic 1). Clearing the CIDL bit in the PCA0MD register allows the PCA to continue normal operation while the CPU is in Idle mode.

Table 22.1. PCA Timebase Input Options

CPS2	CPS1	CPS0	Timebase
0	0	0	System clock divided by 12
0	0	1	System clock divided by 4
0	1	0	Timer 0 overflow
0	1	1	High-to-low transitions on ECI (max rate = system clock divided by 4)
1	0	0	System clock
1	0	1	External oscillator source divided by 8*

*Note: External oscillator source divided by 8 is synchronized with the system clock.

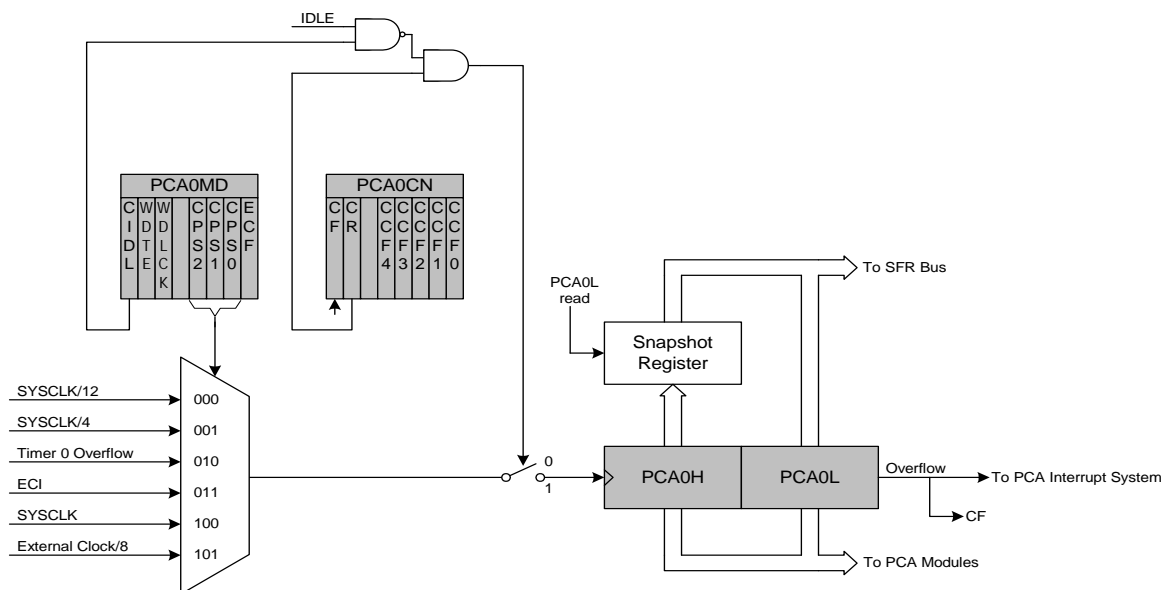


Figure 22.2. PCA Counter/Timer Block Diagram

C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D

22.2.3. High Speed Output Mode

In High Speed Output mode, a module's associated CEX_n pin is toggled each time a match occurs between the PCA Counter and the module's 16-bit capture/compare register (PCA0CPH_n and PCA0CPL_n). Setting the TOG_n, MAT_n, and ECOM_n bits in the PCA0CPM_n 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 PCA0CPL_n clears the ECOM_n bit to '0'; writing to PCA0CPH_n sets ECOM_n to '1'.

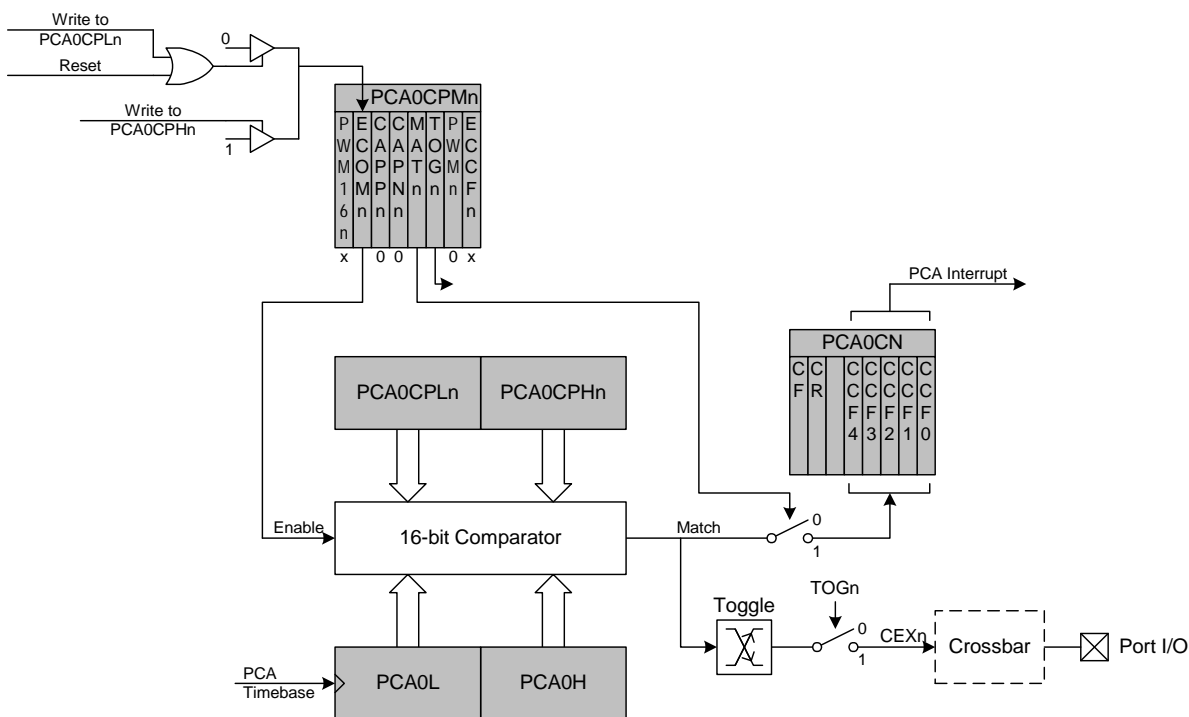


Figure 22.6. PCA High Speed Output Mode Diagram

$$Offset = (256 \times PCA0CPL4) + (256 - PCA0L)$$

Equation 22.4. Watchdog Timer Offset in PCA Clocks

The WDT reset is generated when PCA0L overflows while there is a match between PCA0CPH4 and PCA0H. Software may force a WDT reset by writing a '1' to the CCF4 flag (PCA0CN.4) while the WDT is enabled.

22.3.2. Watchdog Timer Usage

To configure the WDT, perform the following tasks:

1. Disable the WDT by writing a '0' to the WDTE bit.
2. Select the desired PCA clock source (with the CPS2-CPS0 bits).
3. Load PCA0CPL4 with the desired WDT update offset value.
4. Configure the PCA Idle mode (set CIDL if the WDT should be suspended while the CPU is in Idle mode).
5. Enable the WDT by setting the WDTE bit to '1'.
6. (optional) Lock the WDT (prevent WDT disable until the next system reset) by setting the WDLCK bit to '1'.
7. Write a value to PCA0CPH4 to reload the WDT.

The PCA clock source and Idle mode select cannot be changed while the WDT is enabled. The watchdog timer is enabled by setting the WDTE or WDLCK bits in the PCA0MD register. When WDLCK is set, the WDT cannot be disabled until the next system reset. If WDLCK is not set, the WDT is disabled by clearing the WDTE bit.

The WDT is enabled following any reset. The PCA0 counter clock defaults to the system clock divided by 12, PCA0L defaults to 0x00, and PCA0CPL4 defaults to 0x00. Using Equation 22.4, this results in a WDT timeout interval of 256 PCA clocks. Table 22.3 lists some example timeout intervals for typical system clocks.

Table 22.3. Watchdog Timer Timeout Intervals¹

System Clock (Hz)	PCA0CPL4	Timeout Interval (ms)
12,000,000	255	65.5
12,000,000	128	33.0
12,000,000	32	8.4
24,000,000	255	32.8
24,000,000	128	16.5
24,000,000	32	4.2
1,500,000 ²	255	524.3
1,500,000 ²	128	264.2
1,500,000 ²	32	67.6
32,768	255	24,000
32,768	128	12,093.75
32,768	32	3,093.75

Notes:

1. Assumes SYSCLK / 12 as the PCA clock source, and a PCA0L value of 0x00 at the update time.
2. System Clock reset frequency.

23. C2 Interface

C8051F34x devices include an on-chip Silicon Labs 2-Wire (C2) debug interface to allow Flash programming and in-system debugging with the production part installed in the end application. The C2 interface uses a clock signal (C2CK) and a bi-directional C2 data signal (C2D) to transfer information between the device and a host system. See the C2 Interface Specification for details on the C2 protocol.

23.1. C2 Interface Registers

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

C2 Register Definition 23.1. C2ADD: C2 Address

									Reset Value 00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0		

Bits7–0: The C2ADD register is accessed via the C2 interface to select the target Data register for C2 Data Read and Data Write commands.

Address	Description
0x00	Selects the Device ID register for Data Read instructions
0x01	Selects the Revision ID register for Data Read instructions
0x02	Selects the C2 Flash Programming Control register for Data Read/Write instructions
0xAD	Selects the C2 Flash Programming Data register for Data Read/Write instructions

C2 Register Definition 23.2. DEVICEID: C2 Device ID

									Reset Value 00001111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0		

This read-only register returns the 8-bit device ID: 0x0F (C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D).