

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	25MHz
Connectivity	SMBus (2-Wire/I ² C), UART/USART
Peripherals	POR, PWM, Temp Sensor, WDT
Number of I/O	8
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	10-VDFN Exposed Pad
Supplier Device Package	11-QFN (3x3)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f302-gm

13.4.Using the SMBus.....	115
13.4.1.SMBus Configuration Register.....	116
13.4.2.SMB0CN Control Register	119
13.4.3.Data Register	122
13.5.SMBus Transfer Modes.....	123
13.5.1.Master Transmitter Mode	123
13.5.2.Master Receiver Mode	124
13.5.3.Slave Receiver Mode	125
13.5.4.Slave Transmitter Mode	126
13.6.SMBus Status Decoding.....	127
14.UART0.....	131
14.1.Enhanced Baud Rate Generation.....	132
14.2.Operational Modes	133
14.2.1.8-Bit UART	133
14.2.2.9-Bit UART	134
14.3.Multiprocessor Communications	135
15.Timers.....	143
15.1.Timer 0 and Timer 1	143
15.1.1.Mode 0: 13-bit Counter/Timer	143
15.1.2.Mode 1: 16-bit Counter/Timer	145
15.1.3.Mode 2: 8-bit Counter/Timer with Auto-Reload.....	145
15.1.4.Mode 3: Two 8-bit Counter/Timers (Timer 0 Only).....	146
15.2.Timer 2	151
15.2.1.16-bit Timer with Auto-Reload.....	151
15.2.2.8-bit Timers with Auto-Reload.....	152
16.Programmable Counter Array	155
16.1.PCA Counter/Timer	156
16.2.Capture/Compare Modules	157
16.2.1.Edge-triggered Capture Mode.....	158
16.2.2.Software Timer (Compare) Mode.....	159
16.2.3.High Speed Output Mode.....	160
16.2.4.Frequency Output Mode	161
16.2.5.8-Bit Pulse Width Modulator Mode.....	162
16.2.6.16-Bit Pulse Width Modulator Mode.....	163
16.3.Watchdog Timer Mode	164
16.3.1.Watchdog Timer Operation	164
16.3.2.Watchdog Timer Usage	165
16.4.Register Descriptions for PCA.....	167
17.C2 Interface.....	173
17.1.C2 Interface Registers.....	173
17.2.C2 Pin Sharing	175
Document Change List.....	176
Contact Information.....	178

1.8. Comparator

C8051F300/1/2/3/4/5 devices include an on-chip voltage comparator that is enabled/disabled and configured via user software. All Port I/O pins may be configured as comparator inputs. Two comparator outputs may be routed to a Port pin if desired: a latched output and/or an unlatched (asynchronous) output. Comparator response time is programmable, allowing the user to select between high-speed and low-power modes. Positive and negative hysteresis is also configurable.

Comparator interrupts may be generated on rising, falling, or both edges. When in IDLE mode, these interrupts may be used as a “wake-up” source. The comparator may also be configured as a reset source.

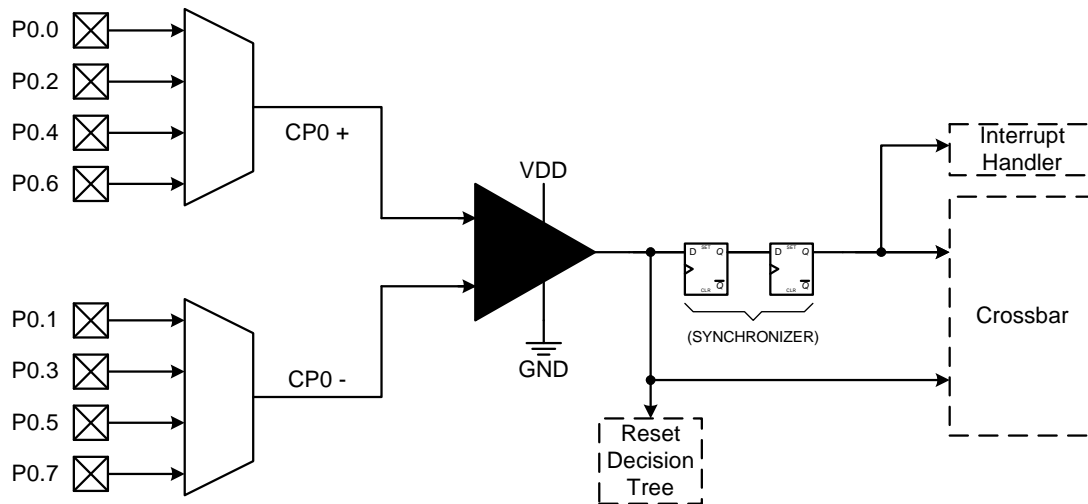


Figure 1.11. Comparator Block Diagram

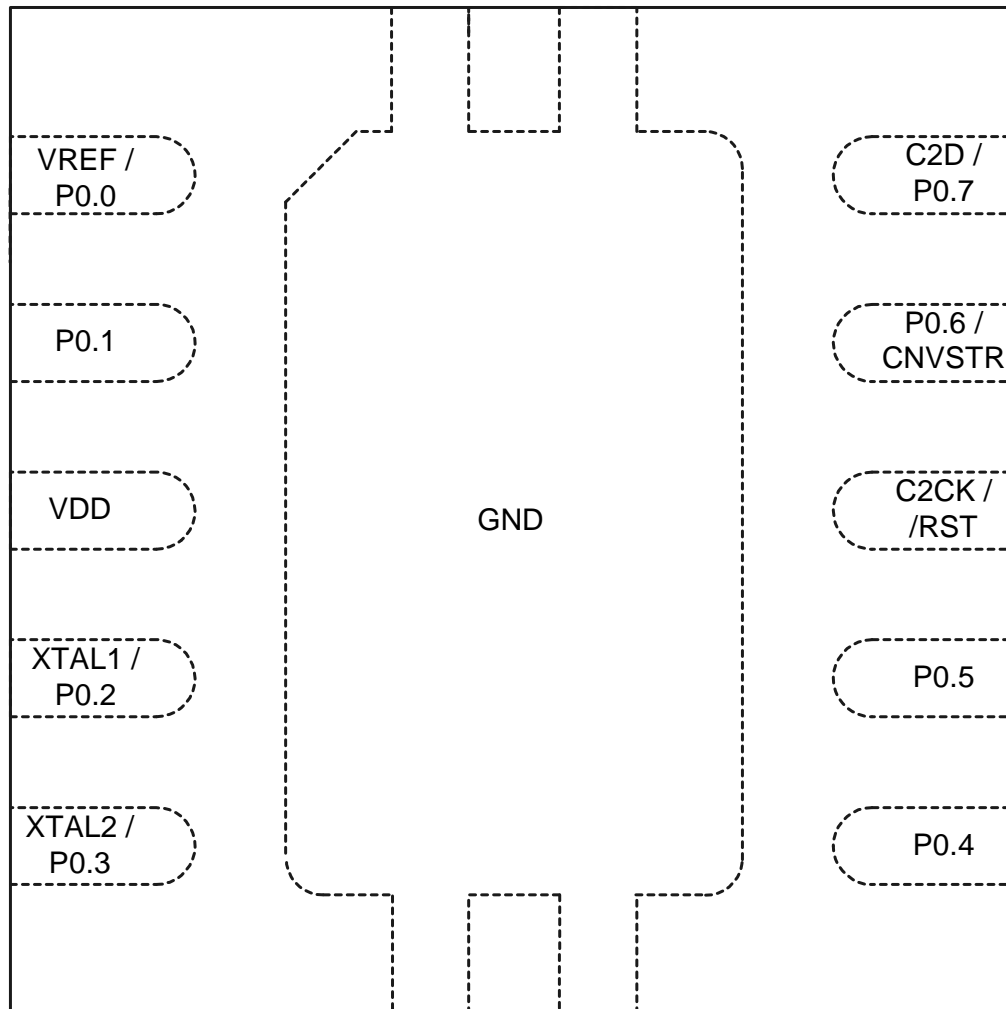


Figure 4.1. QFN-11 Pinout Diagram (Top View)

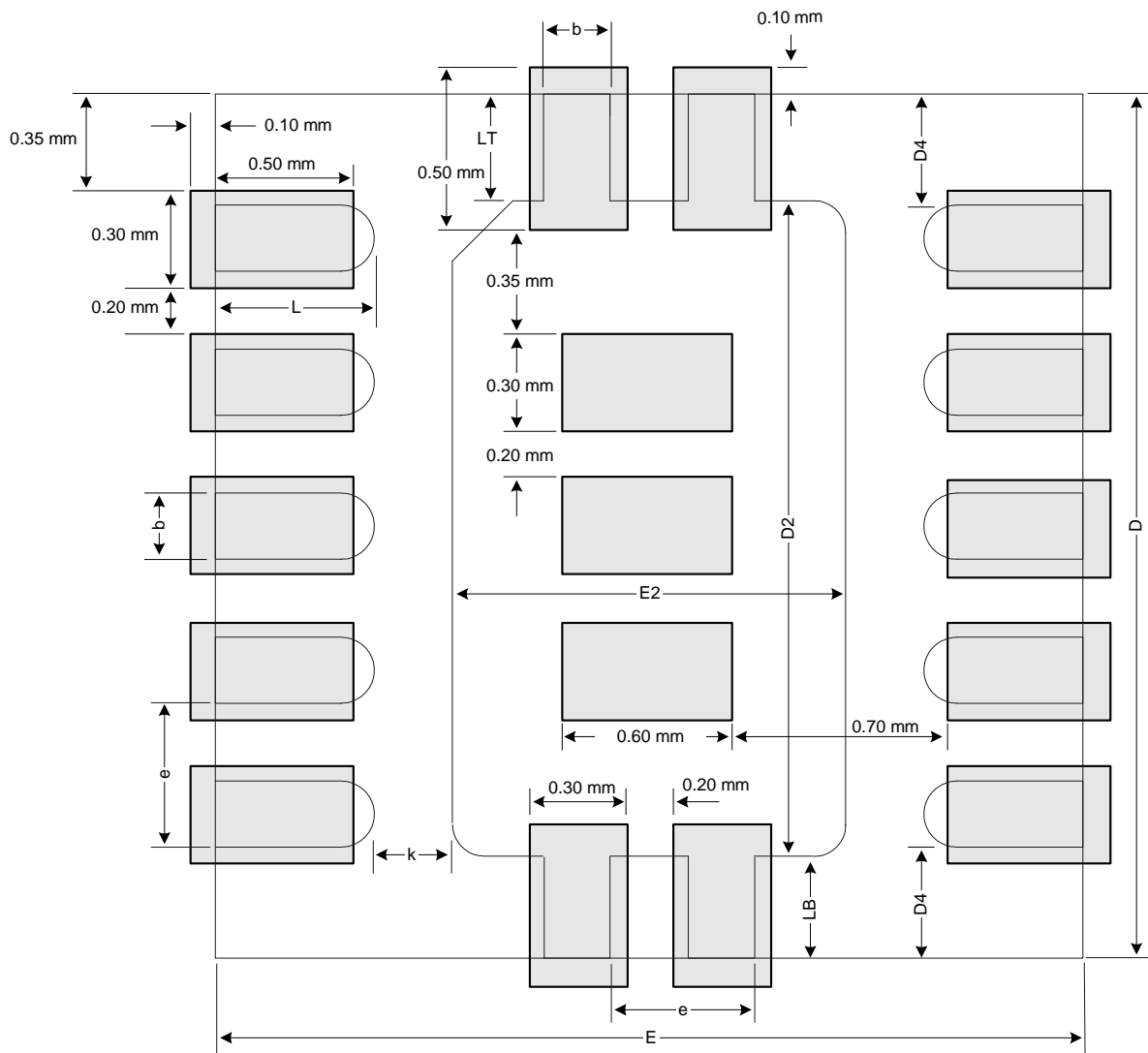


Figure 4.3. Typical QFN-11 Solder Paste Mask

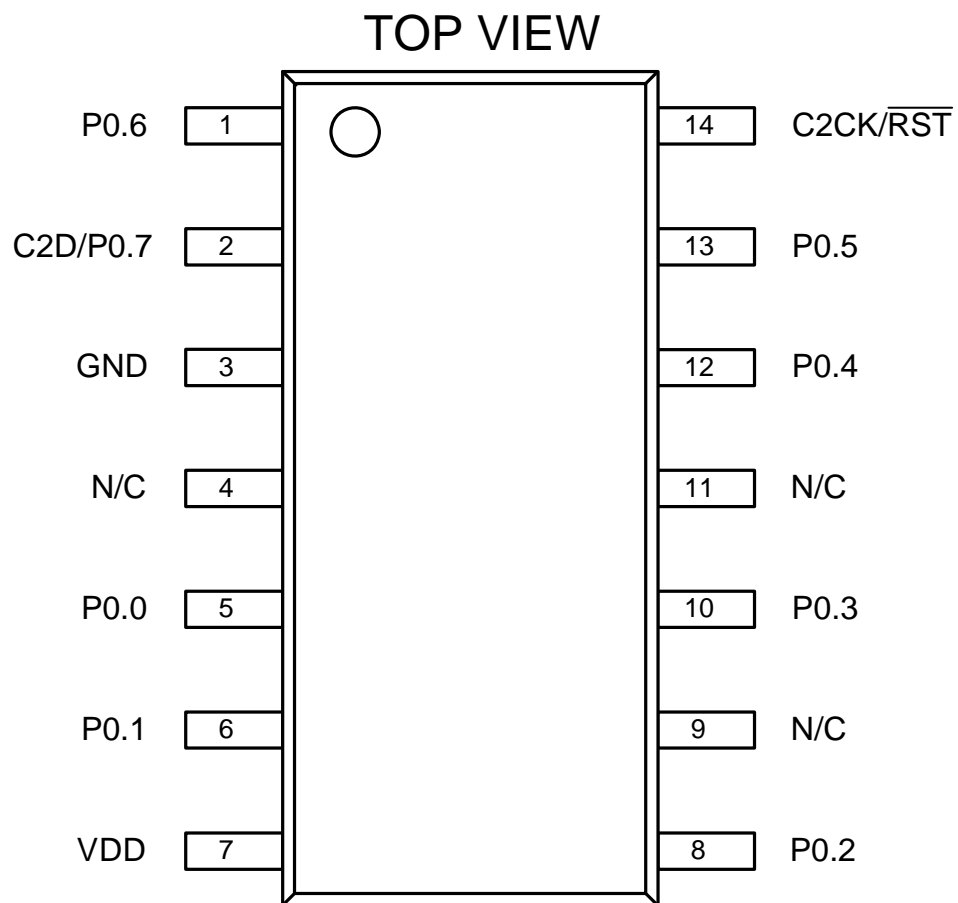


Figure 4.5. SOIC-14 Pinout Diagram (Top View)

5.4. Programmable Window Detector

The ADC Programmable Window Detector continuously compares the ADC0 output to user-programmed limits, and notifies the system when a desired condition is detected. This is especially effective in an interrupt-driven system, saving code space and CPU bandwidth while delivering faster system response times. The window detector interrupt flag (AD0WINT in register ADC0CN) can also be used in polled mode. The ADC0 Greater-Than (ADC0GT) and Less-Than (ADC0LT) registers hold the comparison values. Example comparisons for Single-ended and Differential modes are shown in Figure 5.6 and Figure 5.7, respectively. Notice that the window detector flag can be programmed to indicate when measured data is inside or outside of the user-programmed limits depending on the contents of the ADC0LT and ADC0GT registers.

5.4.1. Window Detector In Single-Ended Mode

Figure 5.6 shows two example window comparisons for Single-ended mode, with ADC0LT = 0x20 and ADC0GT = 0x10. Notice that in Single-ended mode, the codes vary from 0 to VREF x (255/256) and are represented as 8-bit unsigned integers. In the left example, an AD0WINT interrupt will be generated if the ADC0 conversion word (ADC0) is within the range defined by ADC0GT and ADC0LT (if $0x10 < \text{ADC0} < 0x20$). In the right example, an AD0WINT interrupt will be generated if ADC0 is outside of the range defined by ADC0GT and ADC0LT (if $\text{ADC0} < 0x10$ or $\text{ADC0} > 0x20$).

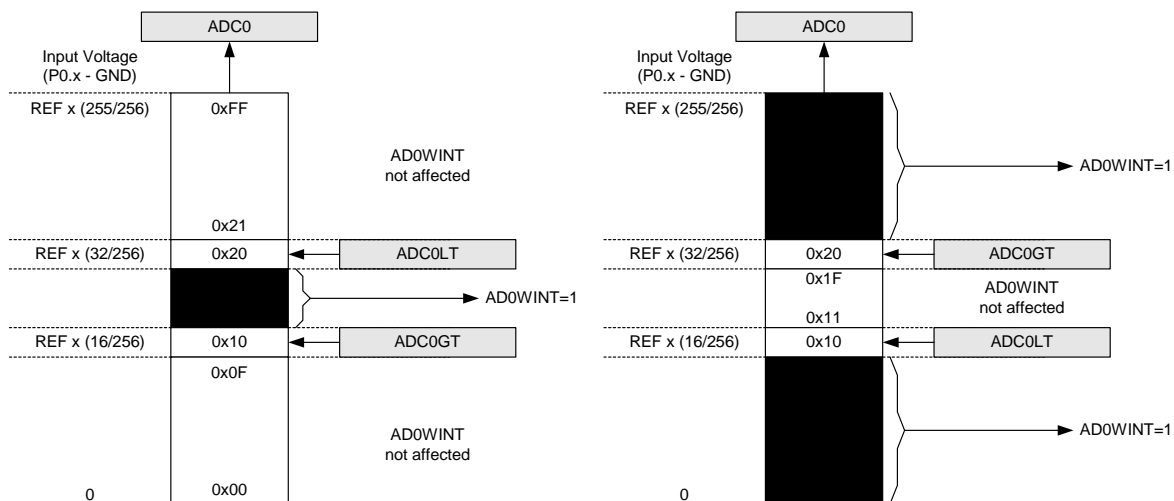


Figure 5.6. ADC Window Compare Examples, Single-Ended Mode

C8051F300/1/2/3/4/5

5.4.2. Window Detector In Differential Mode

Figure 5.7 shows two example window comparisons for differential mode, with $ADC0LT = 0x10 (+16d)$ and $ADC0GT = 0xFF (-1d)$. Notice that in Differential mode, the codes vary from $-VREF$ to $VREF \times (127/128)$ and are represented as 8-bit 2's complement signed integers. In the left example, an $AD0WINT$ interrupt will be generated if the $ADC0$ conversion word ($ADC0L$) is within the range defined by $ADC0GT$ and $ADC0LT$ (if $0xFF (-1d) < ADC0 < 0x10 (16d)$). In the right example, an $AD0WINT$ interrupt will be generated if $ADC0$ is outside of the range defined by $ADC0GT$ and $ADC0LT$ (if $ADC0 < 0xFF (-1d)$ or $ADC0 > 0x10 (+16d)$).

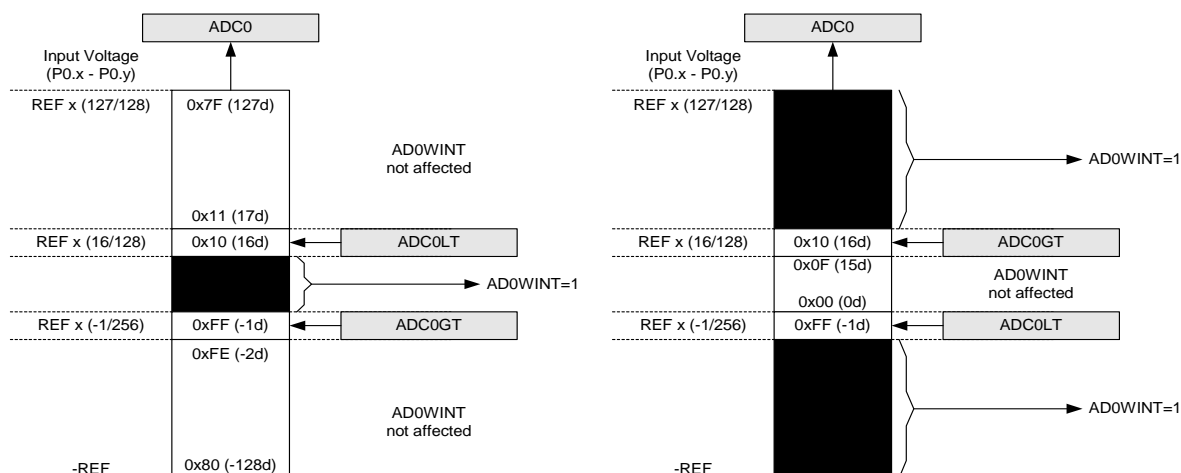


Figure 5.7. ADC Window Compare Examples, Differential Mode

SFR Definition 5.5. ADC0GT: ADC0 Greater-Than Data Byte (C8051F300/2)

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: 0xC4

Bits7–0: ADC0 Greater-Than Data Word.

SFR Definition 5.6. ADC0LT: ADC0 Less-Than Data Byte (C8051F300/2)

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: 0xC6

Bits7–0: ADC0 Less-Than Data Word.

Comparator0 interrupts can be generated on both rising-edge and falling-edge output transitions. (For Interrupt enable and priority control, see **Section “8.3. Interrupt Handler” on page 72**). The CP0FIF flag is set to logic 1 upon a Comparator0 falling-edge interrupt, and the CP0RIF flag is set to logic 1 upon the Comparator0 rising-edge interrupt. Once set, these bits remain set until cleared by software. The output state of Comparator0 can be obtained at any time by reading the CP0OUT bit. Comparator0 is enabled by setting the CP0EN bit to logic 1, and is disabled by clearing this bit to logic 0.

SFR Definition 7.1. CPT0CN: Comparator0 Control

R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
CP0EN	CP0OUT	CP0RIF	CP0FIF	CP0HYP1	CP0HYP0	CP0HYN1	CP0HYN0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
						(bit addressable)		0xF8
<p>Bit7: CP0EN: Comparator0 Enable Bit. 0: Comparator0 Disabled. 1: Comparator0 Enabled.</p> <p>Bit6: CP0OUT: Comparator0 Output State Flag. 0: Voltage on CP0+ < CP0−. 1: Voltage on CP0+ > CP0−.</p> <p>Bit5: CP0RIF: Comparator0 Rising-Edge Interrupt Flag. 0: No Comparator0 Rising Edge Interrupt has occurred since this flag was last cleared. 1: Comparator0 Rising Edge Interrupt has occurred.</p> <p>Bit4: CP0FIF: Comparator0 Falling-Edge Interrupt Flag. 0: No Comparator0 Falling-Edge Interrupt has occurred since this flag was last cleared. 1: Comparator0 Falling-Edge Interrupt has occurred.</p> <p>Bits3–2: CP0HYP1–0: Comparator0 Positive Hysteresis Control Bits. 00: Positive Hysteresis Disabled. 01: Positive Hysteresis = 5 mV. 10: Positive Hysteresis = 10 mV. 11: Positive Hysteresis = 20 mV.</p> <p>Bits1–0: CP0HYN1–0: Comparator0 Negative Hysteresis Control Bits. 00: Negative Hysteresis Disabled. 01: Negative Hysteresis = 5 mV. 10: Negative Hysteresis = 10 mV. 11: Negative Hysteresis = 20 mV.</p>								

C8051F300/1/2/3/4/5

NOTES:

C8051F300/1/2/3/4/5

Table 8.1. CIP-51 Instruction Set Summary (Continued)

Mnemonic	Description	Bytes	Clock Cycles
JNZ rel	Jump if A does not equal zero	2	2/3
CJNE A, direct, rel	Compare direct byte to A and jump if not equal	3	3/4
CJNE A, #data, rel	Compare immediate to A and jump if not equal	3	3/4
CJNE Rn, #data, rel	Compare immediate to Register and jump if not equal	3	3/4
CJNE @Ri, #data, rel	Compare immediate to indirect and jump if not equal	3	4/5
DJNZ Rn, rel	Decrement Register and jump if not zero	2	2/3
DJNZ direct, rel	Decrement direct byte and jump if not zero	3	3/4
NOP	No operation	1	1

Notes on Registers, Operands and Addressing Modes:

Rn - Register R0-R7 of the currently selected register bank.

@Ri - Data RAM location addressed indirectly through R0 or R1.

rel - 8-bit, signed (two's complement) offset relative to the first byte of the following instruction. Used by SJMP and all conditional jumps.

direct - 8-bit internal data location's address. This could be a direct-access Data RAM location (0x00-0x7F) or an SFR (0x80-0xFF).

#data - 8-bit constant

#data16 - 16-bit constant

bit - Direct-accessed bit in Data RAM or SFR

addr11 - 11-bit destination address used by ACALL and AJMP. The destination must be within the same 2K-byte page of program memory as the first byte of the following instruction.

addr16 - 16-bit destination address used by LCALL and LJMP. The destination may be anywhere within the 8K-byte program memory space.

There is one unused opcode (0xA5) that performs the same function as NOP.

All mnemonics copyrighted © Intel Corporation 1980.

Table 8.4. Interrupt Summary

Interrupt Source	Interrupt Vector	Priority Order	Pending Flag	Bit addressable?	Cleared by HW?	Enable Flag	Priority Control
Reset	0x0000	Top	None	N/A	N/A	Always Enabled	Always Highest
External Interrupt 0 (/INT0)	0x0003	0	IE0 (TCON.1)	Y	Y	EX0 (IE.0)	PX0 (IP.0)
Timer 0 Overflow	0x000B	1	TF0 (TCON.5)	Y	Y	ET0 (IE.1)	PT0 (IP.1)
External Interrupt 1 (/INT1)	0x0013	2	IE1 (TCON.3)	Y	Y	EX1 (IE.2)	PX1 (IP.2)
Timer 1 Overflow	0x001B	3	TF1 (TCON.7)	Y	Y	ET1 (IE.3)	PT1 (IP.3)
UART0	0x0023	4	RI0 (SCON0.0) TI0 (SCON0.1)	Y	N	ES0 (IE.4)	PS0 (IP.4)
Timer 2 Overflow	0x002B	5	TF2H (TMR2CN.7) TF2L (TMR2CN.6)	Y	N	ET2 (IE.5)	PT2 (IP.5)
SMBus Interface	0x0033	6	SI (SMB0CN.0)	Y	N	ESMB0 (EIE1.0)	PSMB0 (EIP1.0)
ADC0 Window Compare	0x003B	7	AD0WINT (ADC0CN.3)	Y	N	EWADC0 (EIE1.1)	PWADC0 (EIP1.1)
ADC0 Conversion Complete	0x0043	8	AD0INT (ADC0CN.5)	Y	N	EADC0C (EIE1.2)	PADC0C (EIP1.2)
Programmable Counter Array	0x004B	9	CF (PCA0CN.7) CCFn (PCA0CN.n)	Y	N	EPCA0 (EIE1.3)	PPCA0 (EIP1.3)
Comparator0 Falling Edge	0x0053	10	CP0FIF (CPT0CN.4)	N	N	ECP0F (EIE1.4)	PCP0F (EIP1.4)
Comparator0 Rising Edge	0x005B	11	CP0RIF (CPT0CN.5)	N	N	ECP0R (EIE1.5)	PCP0R (EIP1.5)

SFR Definition 8.8. IP: Interrupt Priority

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
—	—	PT2	PS0	PT1	PX1	PT0	PX0	11000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
						(bit addressable)		0xB8

Bits7–6: UNUSED. Read = 11b, Write = don't care.
 Bit5: PT2: Timer 2 Interrupt Priority Control.
 This bit sets the priority of the Timer 2 interrupt.
 0: Timer 2 interrupts set to low priority level.
 1: Timer 2 interrupts set to high priority level.
 Bit4: PS0: UART0 Interrupt Priority Control.
 This bit sets the priority of the UART0 interrupt.
 0: UART0 interrupts set to low priority level.
 1: UART0 interrupts set to high priority level.
 Bit3: PT1: Timer 1 Interrupt Priority Control.
 This bit sets the priority of the Timer 1 interrupt.
 0: Timer 1 interrupts set to low priority level.
 1: Timer 1 interrupts set to high priority level.
 Bit2: PX1: External Interrupt 1 Priority Control.
 This bit sets the priority of the External Interrupt 1 interrupt.
 0: External Interrupt 1 set to low priority level.
 1: External Interrupt 1 set to high priority level.
 Bit1: PT0: Timer 0 Interrupt Priority Control.
 This bit sets the priority of the Timer 0 interrupt.
 0: Timer 0 interrupts set to low priority level.
 1: Timer 0 interrupts set to high priority level.
 Bit0: PX0: External Interrupt 0 Priority Control.
 This bit sets the priority of the External Interrupt 0 interrupt.
 0: External Interrupt 0 set to low priority level.
 1: External Interrupt 0 set to high priority level.

C8051F300/1/2/3/4/5

10.1.3. Flash Write Procedure

Flash bytes are programmed by software with the following sequence:

- Step 1. Disable interrupts (recommended).
- Step 2. Erase the 512-byte Flash page containing the target location, as described in **Section 10.1.2**.
- Step 3. Set the PSWE bit in PSCTL.
- Step 4. Clear the PSEE bit in PSCTL.
- Step 5. Write the first key code to FLKEY: 0xA5.
- Step 6. Write the second key code to FLKEY: 0xF1.
- Step 7. Using the MOVX instruction, write a single data byte to the desired location within the 512-byte sector.

Steps 5–7 must be repeated for each byte to be written. After Flash writes are complete, PSWE should be cleared so that MOVX instructions do not target program memory. Writing to and erasing the Reserved area of Flash should be avoided.

Table 10.1. Flash Electrical Characteristics

Parameter	Conditions	Min	Typ	Max	Units
Flash Size	C8051F300/1/2/3	8192*			bytes
	C8051F304	4096			bytes
	C8051F305	2048			bytes
Endurance		20k	100k		Erase/Write
Erase Cycle Time	25 MHz System Clock	10	15	20	ms
Write Cycle Time	25 MHz System Clock	40	55	70	μs
SYSCCLK Frequency (Flash writes from application code)		100			kHz

*Note: 512 bytes at location 0x1E00 to 0x1FFF are reserved.

10.2. Non-Volatile Data Storage

The Flash memory can be used for non-volatile data storage as well as program code. This allows data such as calibration coefficients to be calculated and stored at run time. Data is written using the MOVX instruction and read using the MOVC instruction.

10.3. Security Options

The CIP-51 provides security options to protect the Flash memory from inadvertent modification by software as well as to prevent the viewing of proprietary program code and constants. The Program Store Write Enable (bit PSWE in register PSCTL) and the Program Store Erase Enable (bit PSEE in register PSCTL) bits protect the Flash memory from accidental modification by software. PSWE must be explicitly set to '1' before software can modify the Flash memory; both PSWE and PSEE must be set to '1' before software can erase Flash memory. Additional security features prevent proprietary program code and data constants from being read or altered across the C2 interface.

A security lock byte stored at the last byte of Flash user space protects the Flash program memory from being read or altered across the C2 interface. See Table 10.2 for the security byte description; see Figure 10.1 for a program memory map and the security byte locations for each device.

10.4. Flash Write and Erase Guidelines

Any system which contains routines which write or erase Flash memory from software involves some risk that the write or erase routines will execute unintentionally if the CPU is operating outside its specified operating range of V_{DD} , system clock frequency, or temperature. This accidental execution of Flash modifying code can result in alteration of Flash memory contents causing a system failure that is only recoverable by re-Flashing the code in the device.

The following guidelines are recommended for any system which contains routines which write or erase Flash from code.

10.4.1. V_{DD} Maintenance and the V_{DD} monitor

1. If the system power supply is subject to voltage or current "spikes," add sufficient transient protection devices to the power supply to ensure that the supply voltages listed in the Absolute Maximum Ratings table are not exceeded.
2. Make certain that the minimum V_{DD} rise time specification of 1 ms is met. If the system cannot meet this rise time specification, then add an external V_{DD} brownout circuit to the \overline{RST} pin of the device that holds the device in reset until V_{DD} reaches 2.7 V and re-asserts \overline{RST} if V_{DD} drops below 2.7 V.
3. Enable the on-chip V_{DD} monitor and enable the V_{DD} monitor as a reset source as early in code as possible. This should be the first set of instructions executed after the Reset Vector. For 'C'-based systems, this will involve modifying the startup code added by the 'C' compiler. See your compiler documentation for more details. Make certain that there are no delays in software between enabling the V_{DD} monitor and enabling the V_{DD} monitor as a reset source. Code examples showing this can be found in "AN201: Writing to Flash from Firmware", available from the Silicon Laboratories web site.
4. As an added precaution, explicitly enable the V_{DD} monitor and enable the V_{DD} monitor as a reset source inside the functions that write and erase Flash memory. The V_{DD} monitor enable instructions should be placed just after the instruction to set PSWE to a '1', but before the Flash write or erase operation instruction.
5. Make certain that all writes to the RSTSRC (Reset Sources) register use direct assignment operators and explicitly DO NOT use the bit-wise operators (such as AND or OR). For example, "RSTSRC = 0x02" is correct. "RSTSRC |= 0x02" is incorrect.
6. Make certain that all writes to the RSTSRC register explicitly set the PORSF bit to a '1'. Areas to check are initialization code which enables other reset sources, such as the Missing Clock Detector or Comparator, for example, and instructions which force a Software Reset. A global search on "RSTSRC" can quickly verify this.

10.4.2. PSWE Maintenance

7. Reduce the number of places in code where the PSWE bit (b0 in PSCTL) is set to a '1'. There should be exactly one routine in code that sets PSWE to a '1' to write Flash bytes and one routine in code that sets PSWE and PSEE both to a '1' to erase Flash pages.
8. Minimize the number of variable accesses while PSWE is set to a '1'. Handle pointer address updates and loop variable maintenance outside the "PSWE = 1; ... PSWE = 0;" area. Code examples showing this can be found in AN201, "Writing to Flash from Firmware", available from the Silicon Laboratories web site.
9. Disable interrupts prior to setting PSWE to a '1' and leave them disabled until after PSWE has been reset to '0'. Any interrupts posted during the Flash write or erase operation will be ser-

11.4. External Crystal Example

If a crystal or ceramic resonator is used as an external oscillator source for the MCU, the circuit should be configured as shown in Figure 11.1, Option 1. The External Oscillator Frequency Control value (XFCN) should be chosen from the Crystal column of the table in SFR Definition 11.3 (OSCXCN register). For example, an 11.0592 MHz crystal requires an XFCN setting of 111b.

When the crystal oscillator is first enabled, the oscillator amplitude detection circuit requires a settling time to achieve proper bias. Introducing a delay of 1 ms between enabling the oscillator and checking the XTLVLD bit will prevent a premature switch to the external oscillator as the system clock. Switching to the external oscillator before the crystal oscillator has stabilized can result in unpredictable behavior. The recommended procedure is:

- Step 1. Force the XTAL1 and XTAL2 pins low by writing 0's to the port latch.
- Step 2. Configure XTAL1 and XTAL2 as analog inputs.
- Step 3. Enable the external oscillator.
- Step 4. Wait at least 1 ms.
- Step 5. Poll for XTLVLD => '1'.
- Step 6. Switch the system clock to the external oscillator.

Note: Tuning-fork crystals may require additional settling time before XTLVLD returns a valid result.

The capacitors shown in the external crystal configuration provide the load capacitance required by the crystal for correct oscillation. These capacitors are “in series” as seen by the crystal and “in parallel” with the stray capacitance of the XTAL1 and XTAL2 pins.

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

For example, a tuning-fork crystal of 32.768 kHz with a recommended load capacitance of 12.5 pF should use the configuration shown in Figure 12.1, Option 1. The total value of the capacitors and the stray capacitance of the XTAL pins should equal 25 pF. With a stray capacitance of 3 pF per pin, the 22 pF capacitors yield an equivalent capacitance of 12.5 pF across the crystal, as shown in Figure 11.2.

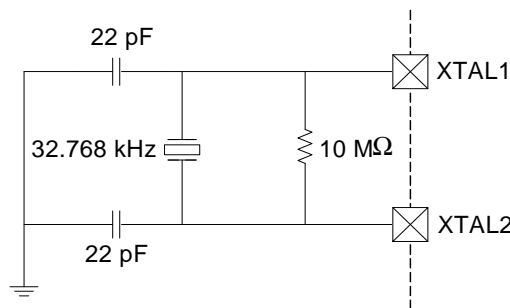


Figure 11.2. 32.768 kHz External Crystal Example

13.3.2. 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.

13.3.3. 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 2 is used to detect SCL low timeouts. Timer 2 is forced to reload when SCL is high, and allowed to count when SCL is low. With Timer 2 enabled and configured to overflow after 25 ms (and SMBTOE set), the Timer 2 interrupt service routine can be used to reset (disable and reenable) the SMBus in the event of an SCL low timeout. Timer 2 configuration details can be found in **Section “15.2. Timer 2” on page 151**.

13.3.4. 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.

14. UART0

UART0 is an asynchronous, full duplex serial port offering modes 1 and 3 of the standard 8051 UART. Enhanced baud rate support allows a wide range of clock sources to generate standard baud rates (details in **Section “14.1. Enhanced Baud Rate Generation” on page 132**). Received data buffering allows UART0 to start reception of a second incoming data byte before software has finished reading the previous data byte.

UART0 has two associated SFRs: Serial Control Register 0 (SCON0) and Serial Data Buffer 0 (SBUF0). The single SBUF0 location provides access to both transmit and receive registers. Reading SBUF0 accesses the buffered Receive register; writing SBUF0 accesses the Transmit register.

With UART0 interrupts enabled, an interrupt is generated each time a transmit is completed (TI0 is set in SCON0), or a data byte has been received (RI0 is set in SCON0). The UART0 interrupt flags are not cleared by hardware when the CPU vectors to the interrupt service routine. They must be cleared manually by software, allowing software to determine the cause of the UART0 interrupt (transmit complete or receive complete).

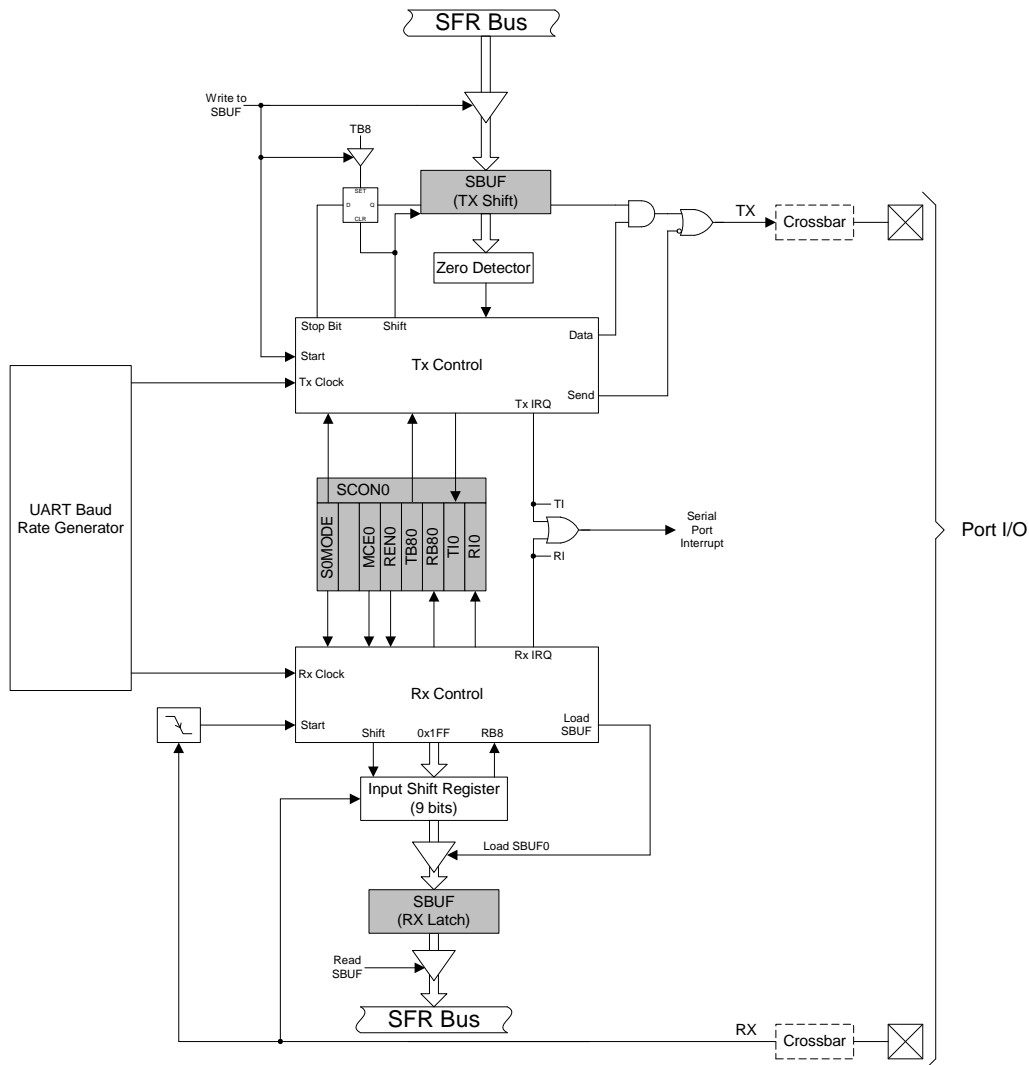


Figure 14.1. UART0 Block Diagram

C8051F300/1/2/3/4/5

SFR Definition 15.4. TL0: Timer 0 Low Byte

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: 0x8A

Bits 7–0: TL0: Timer 0 Low Byte.
The TL0 register is the low byte of the 16-bit Timer 0

SFR Definition 15.5. TL1: Timer 1 Low Byte

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: 0x8B

Bits 7–0: TL1: Timer 1 Low Byte.
The TL1 register is the low byte of the 16-bit Timer 1.

SFR Definition 15.6. TH0: Timer 0 High Byte

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: 0x8C

Bits 7–0: TH0: Timer 0 High Byte.
The TH0 register is the high byte of the 16-bit Timer 0.

SFR Definition 15.7. TH1: Timer 1 High Byte

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: 0x8D

Bits 7–0: TH1: Timer 1 High Byte.
The TH1 register is the high byte of the 16-bit Timer 1.

16.2.2. Software Timer (Compare) Mode

In Software Timer mode, the PCA counter/timer value is compared to the module's 16-bit capture/compare register (PCA0CPHn and PCA0CPLn). When a match occurs, the Capture/Compare Flag (CCFn) in PCA0CN is set to logic 1 and an interrupt request is generated if CCF interrupts are enabled. The CCFn bit is not automatically cleared by hardware when the CPU vectors to the interrupt service routine, and must be cleared by software. Setting the ECOMn and MATn bits in the PCA0CPMn register enables Software Timer 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'.

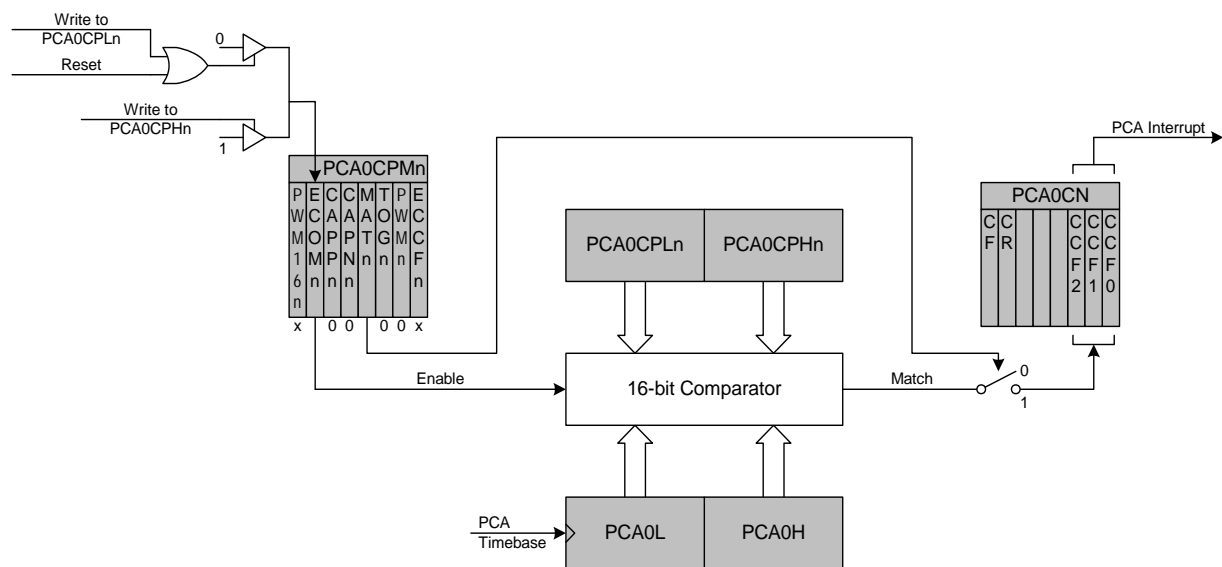


Figure 16.5. PCA Software Timer Mode Diagram

SFR Definition 16.6. PCA0CPLn: PCA Capture Module Low Byte

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: 0xFB, 0xE9, 0xEB
PCA0CPLn Address: PCA0CPL0 = 0xFB (n = 0) PCA0CPL1 = 0xE9 (n = 1) PCA0CPL2 = 0xEB (n = 2)								
Bits7–0: PCA0CPLn: PCA Capture Module Low Byte. The PCA0CPLn register holds the low byte (LSB) of the 16-bit capture Module n.								

SFR Definition 16.7. PCA0CPHn: PCA Capture Module High Byte

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: 0xFC, 0xEA, 0xEC
PCA0CPHn Address: PCA0CPH0 = 0xFC (n = 0) PCA0CPH1 = 0xEA (n = 1) PCA0CPH2 = 0xEC(n = 2)								
Bits7–0: PCA0CPHn: PCA Capture Module High Byte. The PCA0CPHn register holds the high byte (MSB) of the 16-bit capture Module n.								