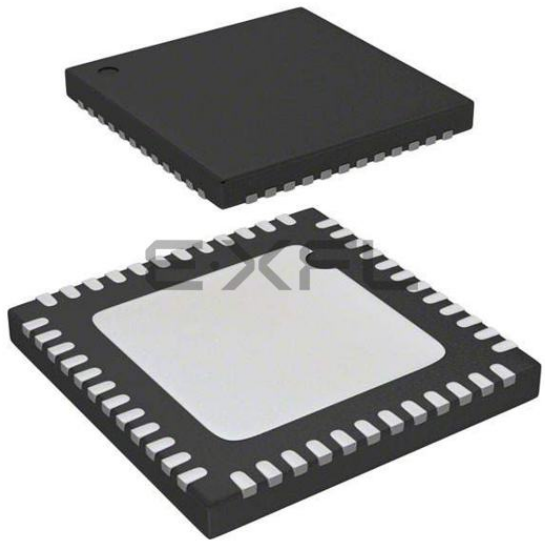

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Cap Sense, POR, PWM, WDT
Number of I/O	39
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-VFQFN Exposed Pad
Supplier Device Package	48-QFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f707-gm

Table 3.1. Pin Definitions for the C8051F70x/71x (Continued)

Name	TQFP64	TQFP48 QFN48	QFN32	QFN24	Type	Description
P3.0	29	—	15	—	D I/O or A In	Port 3.0. CS0 input pin 9.
P3.1	28	—	14	—	D I/O or A In	Port 3.1. CS0 input pin 10.
P3.2	27	—	13	—	D I/O or A In	Port 3.2. CS0 input pin 11.
P3.3	26	—	12	—	D I/O or A In	Port 3.3. CS0 input pin 12.
P3.4	23	19	11	—	D I/O or A In	Port 3.4. CS0 input pin 13.
P3.5	22	18	10	—	D I/O or A In	Port 3.5. CS0 input pin 14.
P3.6	21	17	9	—	D I/O or A In	Port 3.6. CS0 input pin 15.
P3.7	20	16	—	—	D I/O or A In	Port 3.7. CS0 input pin 16.
P4.0	19	15	—	9	D I/O or A In	Port 4.0. CS0 input pin 17.
P4.1	18	14	—	8	D I/O or A In	Port 4.1. CS0 input pin 18.
P4.2	17	13	—	7	D I/O or A In	Port 4.2. CS0 input pin 19.
P4.3	16	12	—	6	D I/O or A In	Port 4.3. CS0 input pin 20.
P4.4	15	—	—	5	D I/O or A In	Port 4.4. CS0 input pin 21.
P4.5	14	—	—	4	D I/O or A In	Port 4.5. CS0 input pin 22.
P4.6	13	—	—	3	D I/O or A In	Port 4.6. CS0 input pin 23.
P4.7	12	—	—	2	D I/O or A In	Port 4.7. CS0 input pin 24.
P5.0	11	11	8	-	D I/O or A In	Port 5.0. CS0 input pin 25.

13. Voltage Regulator (REG0)

C8051F70x/71x devices include an internal voltage regulator (REG0) to regulate the internal core supply to 1.8 V from a V_{DD} supply of 1.8 to 3.6 V. Two power-saving modes are built into the regulator to help reduce current consumption in low-power applications. These modes are accessed through the REG0CN register (SFR Definition 13.1). Electrical characteristics for the on-chip regulator are specified in Table 9.5 on page 50

If an external regulator is used to power the device, the internal regulator may be put into bypass mode using the BYPASS bit. **The internal regulator should never be placed in bypass mode unless an external 1.8 V regulator is used to supply V_{DD}. Doing so could cause permanent damage to the device.**

Under default conditions, when the device enters STOP mode the internal regulator will remain on. This allows any enabled reset source to generate a reset for the device and bring the device out of STOP mode. For additional power savings, the STOPCF bit can be used to shut down the regulator and the internal power network of the device when the part enters STOP mode. When STOPCF is set to 1, the RST pin or a full power cycle of the device are the only methods of generating a reset.

14. Comparator0

C8051F70x/71x devices include an on-chip programmable voltage comparator, Comparator0, shown in Figure 14.1.

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 mode. When assigned to a Port pin, the Comparator output may be configured as open drain or push-pull (see Section “28.4. Port I/O Initialization” on page 189). Comparator0 may also be used as a reset source (see Section “25.5. Comparator0 Reset” on page 167).

The Comparator0 inputs are selected by the comparator input multiplexer, as detailed in Section “14.1. Comparator Multiplexer” on page 78.

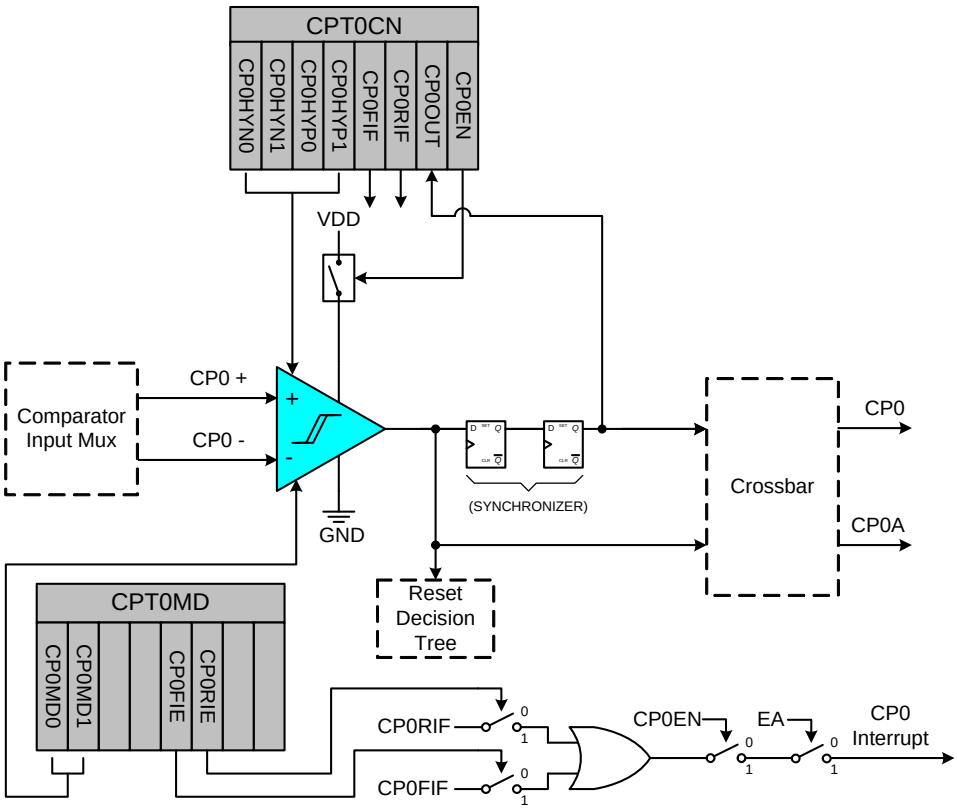

Figure 14.1. Comparator0 Functional Block Diagram

The Comparator output can be polled in software, used as an interrupt 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 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 the power supply to the comparator is turned off. See Section “28.3. Priority Crossbar Decoder” on page 185 for details on configuring Comparator outputs via the digital Crossbar. Comparator inputs can be externally driven from –0.25 V to (V_{DD}) + 0.25 V without damage or upset. The complete Comparator electrical specifications are given in Section “9. Electrical Characteristics” on page 47.

SFR Definition 15.12. CS0MX: Capacitive Sense Mux Channel Select

Bit	7	6	5	4	3	2	1	0
Name	CS0UC		CS0MX[5:0]					
Type	R/W	R/W	R/W					
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9C; SFR Page = 0

Bit	Name	Description
7	CS0UC	**CS0 Unconnected.** Disconnects CS0 from all port pins, regardless of the selected channel. 0: CS0 connected to port pins 1: CS0 disconnected from port pins
6	Reserved	Write = 0b
5:0	CS0MX[5:0]	**CS0 Mux Channel Select.** Selects one of the 38 input channels for Capacitive Sense conversion.

Value	64-pin	48-pin	32-pin	24-pin	Value	64-pin	48-pin	32-pin	24-pin
000000	P2.0	P2.0	P2.0	P2.0	**010011**	P4.3	P4.3	—	P4.3
000001	P2.1	P2.1	P2.1	P2.1	**010100**	P4.4	—	—	P4.4
000010	P2.2	P2.2	P2.2	P2.2	**010101**	P4.5	—	—	P4.5
000011	P2.3	P2.3	P2.3	P2.3	**010110**	P4.6	—	—	P4.6
000100	P2.4	P2.4	P2.4	P2.4	**010111**	P4.7	—	—	P4.7
000101	P2.5	P2.5	P2.5	P2.5	**011000**	P5.0	P5.0	P5.0	—
000110	P2.6	P2.6	P2.6	P2.6	**011001**	P5.1	P5.1	P5.1	—
000111	P2.7	P2.7	P2.7	P2.7	**011010**	P5.2	P5.2	P5.2	—
001000	P3.0	—	P3.0	—	**011011**	P5.3	P5.3	P5.3	—
001001	P3.1	—	P3.1	—	**011100**	P5.4	P5.4	P5.4	—
001010	P3.2	—	P3.2	—	**011101**	P5.5	P5.5	P5.5	—
001011	P3.3	—	P3.3	—	**011110**	P5.6	P5.6	P5.6	—
001100	P3.4	P3.4	P3.4	—	**011111**	P5.7	P5.7	P5.7	—
001101	P3.5	P3.5	P3.5	—	**100000**	P6.0	—	—	—
001110	P3.6	P3.6	P3.6	—	**100001**	P6.1	—	—	—
001111	P3.7	P3.7	—	—	**100010**	P6.2	—	—	—
010000	P4.0	P4.0	—	P4.0	**100011**	P6.3	P6.3	P6.3	—
010001	P4.1	P4.1	—	P4.1	**100100**	P6.4	P6.4	P6.4	P6.4
010010	P4.2	P4.2	—	P4.2	**100101**	P6.5	P6.5	P6.5	P6.5

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 (twos 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 2 kB 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 8 kB program memory space.
There is one unused opcode (0xA5) that performs the same function as NOP. All mnemonics copyrighted © Intel Corporation 1980.

SFR Definition 16.3. SP: Stack Pointer

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

SFR Address = 0x81; SFR Page = All Pages

Bit	Name	Function
7:0	SP[7:0]	**Stack Pointer.** The Stack Pointer holds the location of the top of the stack. The stack pointer is incremented before every PUSH operation. The SP register defaults to 0x07 after reset.

SFR Definition 16.4. ACC: Accumulator

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

SFR Address = 0xE0; SFR Page = All Pages; Bit-Addressable

Bit	Name	Function
7:0	ACC[7:0]	**Accumulator.** This register is the accumulator for arithmetic operations.

20. Special Function Registers

The direct-access data memory locations from 0x80 to 0xFF constitute the special function registers (SFRs). The SFRs provide control and data exchange with the C8051F70x/71x's resources and peripherals. The CIP-51 controller core duplicates the SFRs found in a typical 8051 implementation as well as implementing additional SFRs used to configure and access the sub-systems unique to the C8051F70x/71x. This allows the addition of new functionality while retaining compatibility with the MCS-51™ instruction set. Table 20.1 lists the SFRs implemented in the C8051F70x/71x device family.

The SFR registers are accessed anytime the direct addressing mode is used to access memory locations from 0x80 to 0xFF. SFRs with addresses ending in 0x0 or 0x8 (e.g., P0, TCON, SCON0, IE, etc.) are bit-addressable as well as byte-addressable. All other SFRs are byte-addressable only. Unoccupied addresses in the SFR space are reserved for future use. Accessing these areas will have an indeterminate effect and should be avoided. Refer to the corresponding pages of the data sheet, as indicated in Table 20.2, for a detailed description of each register.

22. Flash Memory

On-chip, re-programmable Flash memory is included for program code and non-volatile data storage. The Flash memory can be programmed in-system through the C2 interface or by software using the MOVX write instruction. Once cleared to logic 0, a Flash bit must be erased to set it back to logic 1. Flash bytes would typically be erased (set to 0xFF) before being reprogrammed. The write and erase operations are automatically timed by hardware for proper execution; data polling to determine the end of the write/erase operations is not required. Code execution is stalled during Flash write/erase operations. Refer to Table 9.6 for complete Flash memory electrical characteristics.

22.1. Programming The Flash Memory

The simplest means of programming the Flash memory is through the C2 interface using programming tools provided by Silicon Laboratories or a third party vendor. This is the only means for programming a non-initialized device. For details on the C2 commands to program Flash memory, see Section “35. C2 Interface” on page 301.

The Flash memory can be programmed by software using the MOVX write instruction with the address and data byte to be programmed provided as normal operands. Before programming Flash memory using MOVX, Flash programming operations must be enabled by: (1) setting the PSWE Program Store Write Enable bit (PSCTL.0) to logic 1 (this directs the MOVX writes to target Flash memory); and (2) Writing the Flash key codes in sequence to the Flash Lock register (FLKEY). The PSWE bit remains set until cleared by software.

Note: A minimum SYSCLK frequency is required for writing or erasing Flash memory, as detailed in Section “Table 9.6. Flash Electrical Characteristics” on page 50.

For detailed guidelines on programming Flash from firmware, please see Section “22.4. Flash Write and Erase Guidelines” on page 150.

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

22.1.1. Flash Lock and Key Functions

Flash writes and erases by user software are protected with a lock and key function. The Flash Lock and Key Register (FLKEY) must be written with the correct key codes, in sequence, before Flash operations may be performed. The key codes are: 0xA5, 0xF1. The timing does not matter, but the codes must be written in order. If the key codes are written out of order, or the wrong codes are written, Flash writes and erases will be disabled until the next system reset. Flash writes and erases will also be disabled if a Flash write or erase is attempted before the key codes have been written properly. The Flash lock resets after each write or erase; the key codes must be written again before a following Flash operation can be performed. The FLKEY register is detailed in SFR Definition 22.2.

22.1.2. Flash Erase Procedure

The Flash memory is organized in 512-byte pages. The erase operation applies to an entire page (setting all bytes in the page to 0xFF). To erase an entire 512-byte page, perform the following steps:

1. Save current interrupt state and disable interrupts.
2. Set the PSEE bit (register PSCTL).
3. Set the PSWE bit (register PSCTL).
4. Write the first key code to FLKEY: 0xA5.
5. Write the second key code to FLKEY: 0xF1.

22.4.3. System Clock

12. If operating from an external crystal, be advised that crystal performance is susceptible to electrical interference and is sensitive to layout and to changes in temperature. If the system is operating in an electrically noisy environment, use the internal oscillator or use an external CMOS clock.
13. If operating from the external oscillator, switch to the internal oscillator during Flash write or erase operations. The external oscillator can continue to run, and the CPU can switch back to the external oscillator after the Flash operation has completed.

Additional Flash recommendations and example code can be found in “AN201: Writing to Flash from Firmware,″ available from the Silicon Laboratories web site.

SFR Definition 28.17. P2: Port 2

Bit	7	6	5	4	3	2	1	0
Name	P2[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Address = 0xA0; SFR Page = All Pages; Bit Addressable

Bit	Name	Description	Write	Read
7:0	P2[7:0]	**Port 2 Data.** Sets the Port latch logic value or reads the Port pin logic state in Port cells configured for digital I/O.	0: Set output latch to logic LOW. 1: Set output latch to logic HIGH.	0: P2.n Port pin is logic LOW. 1: P2.n Port pin is logic HIGH.

SFR Definition 28.18. P2MDIN: Port 2 Input Mode

Bit	7	6	5	4	3	2	1	0
Name	P2MDIN[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Address = 0xF3; SFR Page = F

Bit	Name	Function
7:0	P2MDIN[7:0]	**Analog Configuration Bits for P2.7–P2.0 (respectively).** Port pins configured for analog mode have their weak pullup, digital driver, and digital receiver disabled. 0: Corresponding P2.n pin is configured for analog mode. 1: Corresponding P2.n pin is not configured for analog mode.

SFR Definition 28.19. P2MDOUT: Port 2 Output Mode

Bit	7	6	5	4	3	2	1	0
Name	P2MDOUT[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xA6; SFR Page = F

Bit	Name	Function
7:0	P2MDOUT[7:0]	**Output Configuration Bits for P2.7–P2.0 (respectively).** These bits are ignored if the corresponding bit in register P2MDIN is logic 0. 0: Corresponding P2.n Output is open-drain. 1: Corresponding P2.n Output is push-pull.

SFR Definition 28.20. P2SKIP: Port 2 Skip

Bit	7	6	5	4	3	2	1	0
Name	P2SKIP[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD6; SFR Page = F

Bit	Name	Function
7:0	P2SKIP[3:0]	**Port 2 Crossbar Skip Enable Bits.** These bits select Port 2 pins to be skipped by the Crossbar Decoder. Port pins used for analog, special functions or GPIO should be skipped by the Crossbar. 0: Corresponding P2.n pin is not skipped by the Crossbar. 1: Corresponding P2.n pin is skipped by the Crossbar.

SFR Definition 28.37. P6DRV: Port 6 Drive Strength

Bit	7	6	5	4	3	2	1	0
Name			P6DRV[5:0]					
Type	R	R	R/W					
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xC1; SFR Page = F

Bit	Name	Function
7:6	Unused	Read = 00b; Write = Don't Care
5:0	P6DRV[5:0]	**Drive Strength Configuration Bits for P6.5–P6.0 (respectively).** Configures digital I/O Port cells to high or low output drive strength. 0: Corresponding P6.n Output has low output drive strength. 1: Corresponding P6.n Output has high output drive strength.

SFR Definition 29.1. CRC0CN: CRC0 Control

Bit	7	6	5	4	3	2	1	0
Name				CRC0SEL	CRC0INIT	CRC0VAL	CRC0PNT[1:0]	
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x91; SFR Page = F

Bit	Name	Function
7:5	Unused	Read = 000b; Write = Don't Care.
4	CRC0SEL	**CRC0 Polynomial Select Bit.** This bit selects the CRC0 polynomial and result length (32-bit or 16-bit). 0: CRC0 uses the 32-bit polynomial 0x04C11DB7 for calculating the CRC result. 1: CRC0 uses the 16-bit polynomial 0x1021 for calculating the CRC result.
3	CRC0INIT	**CRC0 Result Initialization Bit.** Writing a 1 to this bit initializes the entire CRC result based on CRC0VAL.
2	CRC0VAL	**CRC0 Set Value Initialization Bit.** This bit selects the set value of the CRC result. 0: CRC result is set to 0x00000000 on write of 1 to CRC0INIT. 1: CRC result is set to 0xFFFFFFFF on write of 1 to CRC0INIT.
1:0	CRC0PNT[1:0]	**CRC0 Result Pointer.** Specifies the byte of the CRC result to be read/written on the next access to CRC0DAT. The value of these bits will auto-increment upon each read or write. For CRC0SEL = 0: 00: CRC0DAT accesses bits 7–0 of the 32-bit CRC result. 01: CRC0DAT accesses bits 15–8 of the 32-bit CRC result. 10: CRC0DAT accesses bits 23–16 of the 32-bit CRC result. 11: CRC0DAT accesses bits 31–24 of the 32-bit CRC result. For CRC0SEL = 1: 00: CRC0DAT accesses bits 7–0 of the 16-bit CRC result. 01: CRC0DAT accesses bits 15–8 of the 16-bit CRC result. 10: CRC0DAT accesses bits 7–0 of the 16-bit CRC result. 11: CRC0DAT accesses bits 15–8 of the 16-bit CRC result.

Table 30.5. SMBus Status Decoding: Hardware ACK Disabled (EHACK = 0) (Continued)

Mode	Values Read				Current SMbus State	Typical Response Options	Values to Write			Next Status Vector Expected
	Status Vector	ACKRQ	ARBLOST	ACK			STA	STO	ACK	
Bus Error Condition	0010	0	1	X	Lost arbitration while attempting a repeated START.	Abort failed transfer.	0	0	X	—
						Reschedule failed transfer.	1	0	X	1110
	0001	0	1	X	Lost arbitration due to a detected STOP.	Abort failed transfer.	0	0	X	—
						Reschedule failed transfer.	1	0	X	1110
	0000	1	1	X	Lost arbitration while transmitting a data byte as master.	Abort failed transfer.	0	0	0	—
						Reschedule failed transfer.	1	0	0	1110

Table 30.6. SMBus Status Decoding: Hardware ACK Enabled (EHACK = 1)

Mode	Values Read				Current SMbus State	Typical Response Options	Values to Write			Next Status Vector Expected
	Status Vector	ACKRQ	ARBLOST	ACK			STA	STO	ACK	
Master Transmitter	1110	0	0	X	A master START was generated.	Load slave address + R/W into SMB0DAT.	0	0	X	1100
	1100	0	0	0	A master data or address byte was transmitted; NACK received.	Set STA to restart transfer.	1	0	X	1110
						Abort transfer.	0	1	X	—
		0	0	1	A master data or address byte was transmitted; ACK received.	Load next data byte into SMB0DAT.	0	0	X	1100
						End transfer with STOP.	0	1	X	—
						End transfer with STOP and start another transfer.	1	1	X	—
						Send repeated START.	1	0	X	1110
						Switch to Master Receiver Mode (clear SI without writing new data to SMB0DAT). Set ACK for initial data byte.	0	0	1	1000

Table 30.6. SMBus Status Decoding: Hardware ACK Enabled (EHACK = 1) (Continued)

Mode	Values Read				Current SMbus State	Typical Response Options	Values to Write			Next Status Vector Expected
	Status Vector	ACKRQ	ARBLOST	ACK			STA	STO	ACK	
Master Receiver	1000	0	0	1	A master data byte was received; ACK sent.	Set ACK for next data byte; Read SMB0DAT.	0	0	1	1000
						Set NACK to indicate next data byte as the last data byte; Read SMB0DAT.	0	0	0	1000
						Initiate repeated START.	1	0	0	1110
						Switch to Master Transmitter Mode (write to SMB0DAT before clearing SI).	0	0	X	1100
		0	0	0	A master data byte was received; NACK sent (last byte).	Read SMB0DAT; send STOP.	0	1	0	—
						Read SMB0DAT; Send STOP followed by START.	1	1	0	1110
						Initiate repeated START.	1	0	0	1110
						Switch to Master Transmitter Mode (write to SMB0DAT before clearing SI).	0	0	X	1100
Slave Transmitter	0100	0	0	0	A slave byte was transmitted; NACK received.	No action required (expecting STOP condition).	0	0	X	0001
		0	0	1	A slave byte was transmitted; ACK received.	Load SMB0DAT with next data byte to transmit.	0	0	X	0100
		0	1	X	A Slave byte was transmitted; error detected.	No action required (expecting Master to end transfer).	0	0	X	0001
	0101	0	X	X	An illegal STOP or bus error was detected while a Slave Transmission was in progress.	Clear STO.	0	0	X	—

SFR Definition 33.13. TMR3CN: Timer 3 Control

Bit	7	6	5	4	3	2	1	0
Name	TF3H	TF3L	TF3LEN	TF3CEN	T3SPLIT	TR3		T3XCLK
Type	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x91; SFR Page = 0

Bit	Name	Function
7	TF3H	**Timer 3 High Byte Overflow Flag.** Set by hardware when the Timer 3 high byte overflows from 0xFF to 0x00. In 16 bit mode, this will occur when Timer 3 overflows from 0xFFFF to 0x0000. When the Timer 3 interrupt is enabled, setting this bit causes the CPU to vector to the Timer 3 interrupt service routine. This bit is not automatically cleared by hardware.
6	TF3L	**Timer 3 Low Byte Overflow Flag.** Set by hardware when the Timer 3 low byte overflows from 0xFF to 0x00. TF3L will be set when the low byte overflows regardless of the Timer 3 mode. This bit is not automatically cleared by hardware.
5	TF3LEN	**Timer 3 Low Byte Interrupt Enable.** When set to 1, this bit enables Timer 3 Low Byte interrupts. If Timer 3 interrupts are also enabled, an interrupt will be generated when the low byte of Timer 3 overflows.
4	TF3CEN	**Timer 3 Comparator Capture Enable.** When set to 1, this bit enables Timer 3 Comparator Capture Mode. If TF3CEN is set, on a rising edge of the Comparator0 output the current 16-bit timer value in TMR3H:TMR3L will be copied to TMR3RLH:TMR3RLL. If Timer 3 interrupts are also enabled, an interrupt will be generated on this event.
3	T3SPLIT	**Timer 3 Split Mode Enable.** When this bit is set, Timer 3 operates as two 8-bit timers with auto-reload. 0: Timer 3 operates in 16-bit auto-reload mode. 1: Timer 3 operates as two 8-bit auto-reload timers.
2	TR3	**Timer 3 Run Control.** Timer 3 is enabled by setting this bit to 1. In 8-bit mode, this bit enables/disables TMR3H only; TMR3L is always enabled in split mode.
1	Unused	Read = 0b; Write = Don't Care.
0	T3XCLK	**Timer 3 External Clock Select.** This bit selects the external clock source for Timer 3. If Timer 3 is in 8-bit mode, this bit selects the external oscillator clock source for both timer bytes. However, the Timer 3 Clock Select bits (T3MH and T3ML in register CKCON) may still be used to select between the external clock and the system clock for either timer. 0: System clock divided by 12. 1: External clock divided by 8 (synchronized with SYSCLK when not in suspend).

Table 34.2. PCA0CPM and PCA0PWM Bit Settings for PCA Modules

Operational Mode	PCA0CPMn								PCA0PWM				
Bit Number	7	6	5	4	3	2	1	0	7	6	5	4:2	1:0
Capture triggered by positive edge on CEXn	X	X	1	0	0	0	0	A	0	X	B	XXX	XX
Capture triggered by negative edge on CEXn	X	X	0	1	0	0	0	A	0	X	B	XXX	XX
Capture triggered by any transition on CEXn	X	X	1	1	0	0	0	A	0	X	B	XXX	XX
Software Timer	X	C	0	0	1	0	0	A	0	X	B	XXX	XX
High Speed Output	X	C	0	0	1	1	0	A	0	X	B	XXX	XX
Frequency Output	X	C	0	0	0	1	1	A	0	X	B	XXX	XX
8-Bit Pulse Width Modulator (Note 7)	0	C	0	0	E	0	1	A	0	X	B	XXX	00
9-Bit Pulse Width Modulator (Note 7)	0	C	0	0	E	0	1	A	D	X	B	XXX	01
10-Bit Pulse Width Modulator (Note 7)	0	C	0	0	E	0	1	A	D	X	B	XXX	10
11-Bit Pulse Width Modulator (Note 7)	0	C	0	0	E	0	1	A	D	X	B	XXX	11
16-Bit Pulse Width Modulator	1	C	0	0	E	0	1	A	0	X	B	XXX	XX

Notes:

1. X = Don't Care (no functional difference for individual module if 1 or 0).
2. A = Enable interrupts for this module (PCA interrupt triggered on CCFn set to 1).
3. B = Enable 8th, 9th, 10th or 11th bit overflow interrupt (Depends on setting of CLSEL[1:0]).
4. C = When set to 0, the digital comparator is off. For high speed and frequency output modes, the associated pin will not toggle. In any of the PWM modes, this generates a 0% duty cycle (output = 0).
5. D = Selects whether the Capture/Compare register (0) or the Auto-Reload register (1) for the associated channel is accessed via addresses PCA0CPHn and PCA0CPLn.
6. E = When set, a match event will cause the CCFn flag for the associated channel to be set.
7. All modules set to 8, 9, 10 or 11-bit PWM mode use the same cycle length setting.

34.3.3. 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). When a match occurs, the Capture/Compare Flag (CCFn) in PCA0CN is set to logic 1. An interrupt request is generated if the CCFn interrupt for that module is 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 TOGn, MATn, and ECOMn bits in the PCA0CPMn register enables the High-Speed Output mode. If ECOMn is cleared, the associated pin will retain its state, and not toggle on the next match event.

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.

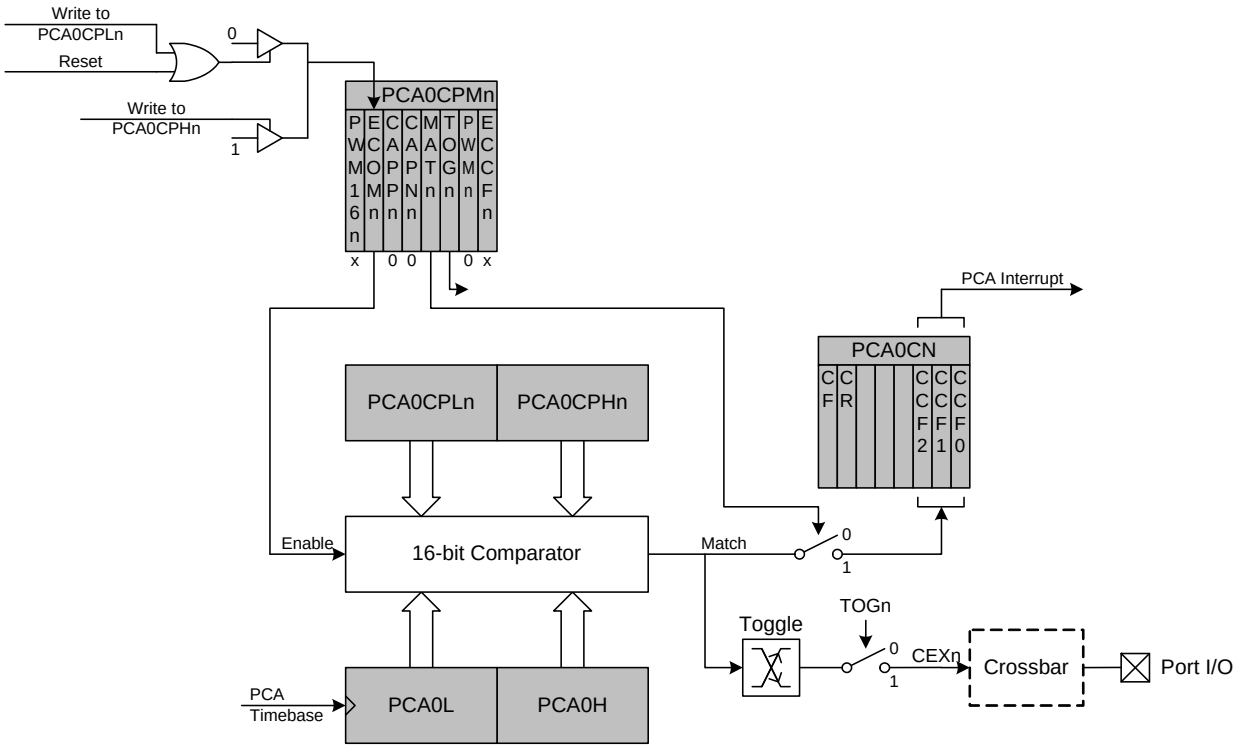

Figure 34.6. PCA High-Speed Output Mode Diagram

34.3.6. 16-Bit Pulse Width Modulator Mode

A PCA module may also be operated in 16-Bit PWM mode. 16-bit PWM mode is independent of the other (8/9/10/11-bit) PWM modes. In this mode, the 16-bit capture/compare module defines the number of PCA clocks for the low time of the PWM signal. When the PCA counter matches the module contents, the output on CEXn is asserted high; when the 16-bit counter overflows, CEXn is asserted low. To output a varying duty cycle, new value writes should be synchronized with PCA CCFn match interrupts. 16-Bit PWM Mode is enabled by setting the ECOMn, PWMn, and PWM16n bits in the PCA0CPMn register. For a varying duty cycle, match interrupts should be enabled (ECCFn = 1 AND MATn = 1) to help synchronize the capture/compare register writes. If the MATn bit is set to 1, the CCFn flag for the module will be set each time a 16-bit comparator match (rising edge) occurs. The CF flag in PCA0CN can be used to detect the overflow (falling edge). The duty cycle for 16-Bit PWM Mode is given by Equation 34.4.

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.

$$\text{Duty Cycle} = \frac{(65536 - PCA0CPn)}{65536}$$

Equation 34.4. 16-Bit PWM Duty Cycle

Using Equation 34.4, the largest duty cycle is 100% (PCA0CPn = 0), and the smallest duty cycle is 0.0015% (PCA0CPn = 0xFFFF). A 0% duty cycle may be generated by clearing the ECOMn bit to 0.

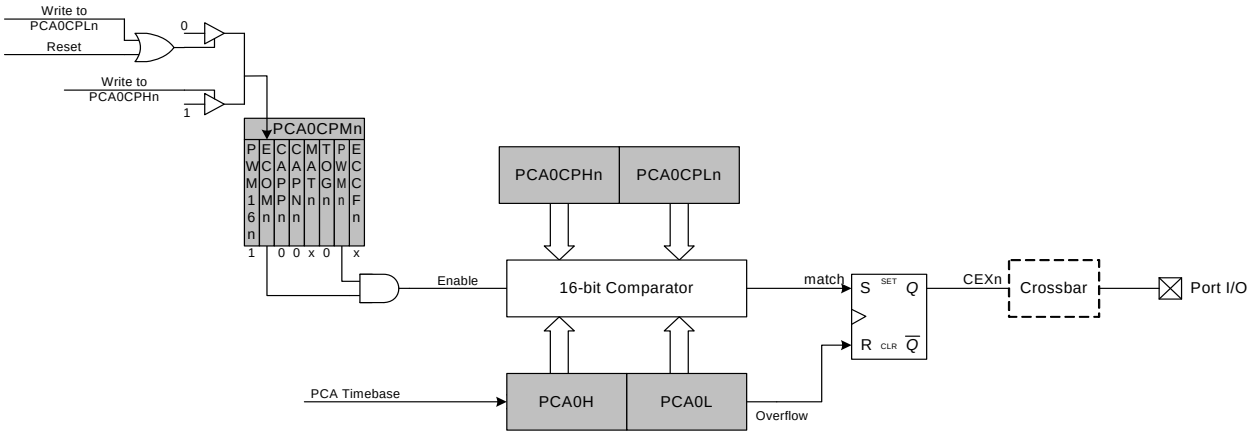

Figure 34.10. PCA 16-Bit PWM Mode

35. C2 Interface

C8051F70x/71x 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 operates using only two pins: a bi-directional data signal (C2D), and a clock input (C2CK). See the C2 Interface Specification for details on the C2 protocol.

35.1. 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 35.1. C2ADD: C2 Address

Bit	7	6	5	4	3	2	1	0
Name	C2ADD[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function		
7:0	C2ADD[7:0]	**C2 Address.** The C2ADD register is accessed via the C2 interface to select the target Data register for C2 Data Read and Data Write commands.		
		Address	**Name**	**Description**
		0x00	DEVICEID	Selects the Device ID Register (read only)
		0x01	REVID	Selects the Revision ID Register (read only)
		0x02	FPCTL	Selects the C2 Flash Programming Control Register
		0xBF	FPDAT	Selects the C2 Flash Data Register
		0x96	CRC0AUTO*	Selects the CRC0AUTO Register
		0x97	CRC0CNT*	Selects the CRC0CNT Register
		0x91	CRC0CN*	Selects the CRC0CN Register
		0xD9	CRC0DATA*	Selects the CRC0DATA Register
		0x95	CRC0FLIP*	Selects the CRC0FLIP Register
		0x94	CRC0IN*	Selects the CRC0IN Register

Note: CRC registers and functions are described in Section “29. Cyclic Redundancy Check Unit (CRC0)” on page 211.