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	Last Time Buy
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I²C), UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	8
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	14-SOIC (0.154", 3.90mm Width)
Supplier Device Package	14-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f304-gs

NOTES:

List of Figures

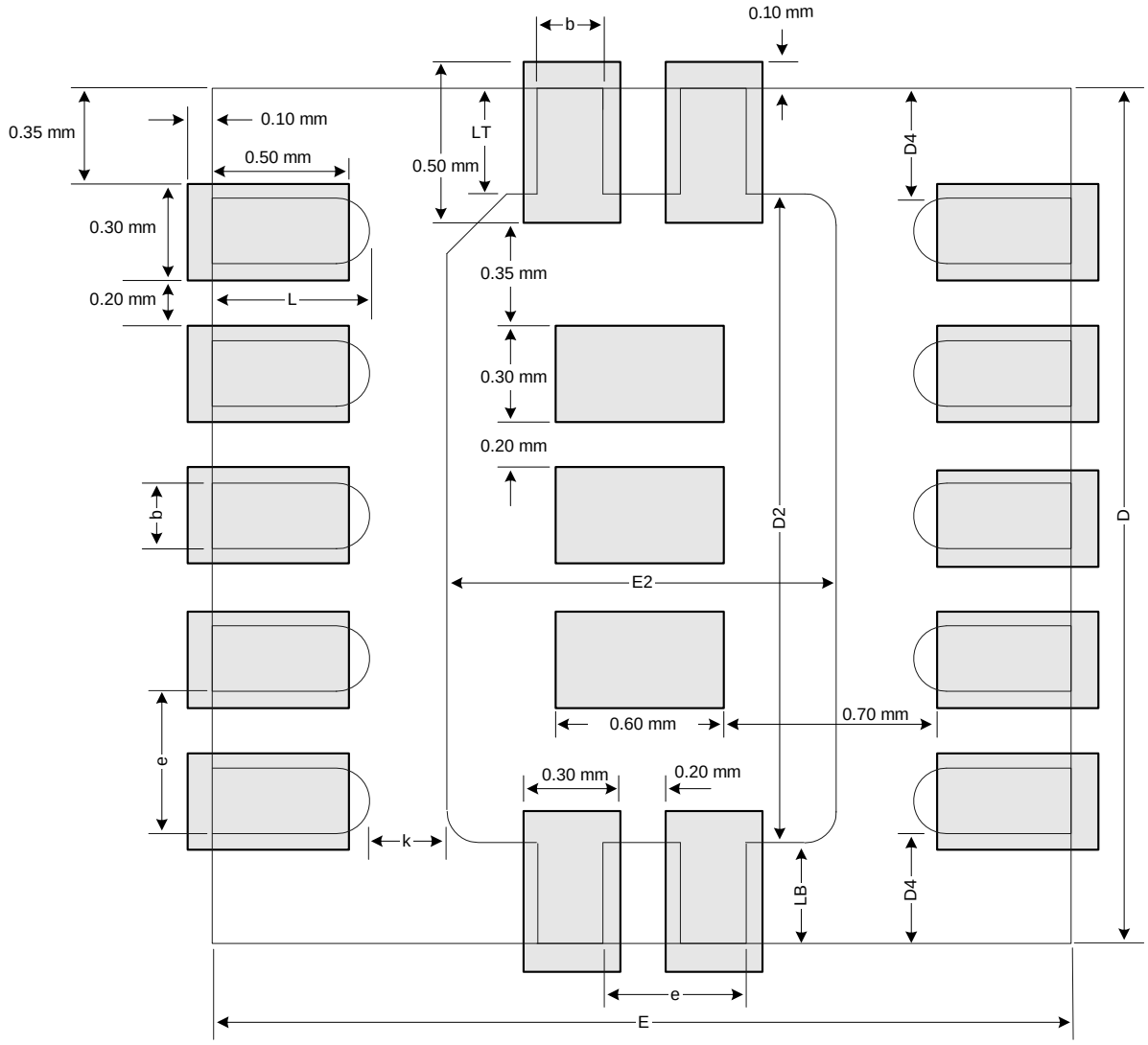

Figure 4.3. Typical QFN-11 Solder Paste Mask

NOTES:

SFR Definition 6.1. REF0CN: Reference Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
—	—	—	—	REFSL	TEMPE	BIASE	—	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xD1

Bits7–3: UNUSED. Read = 00000b; Write = don't care.
Bit3: REFSL: Voltage Reference Select.
This bit selects the source for the internal voltage reference.
0: VREF input pin used as voltage reference.
1: V_{DD} used as voltage reference.
Bit2: TEMPE: Temperature Sensor Enable Bit.
0: Internal Temperature Sensor off.
1: Internal Temperature Sensor on.
Bit1: BIASE: Internal Analog Bias Generator Enable Bit. (Must be '1' if using ADC).
0: Internal Bias Generator off.
1: Internal Bias Generator on.
Bit0: UNUSED. Read = 0b. Write = don't care.

Table 6.1. External Voltage Reference Circuit Electrical Characteristics

V_{DD} = 3.0 V; –40 to +85°C unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
Input Voltage Range		0	—	V_{DD}	V
Input Current	Sample Rate = 500 ksps; VREF = 3.0 V	—	12	—	µA

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

Bit7: CP0EN: Comparator0 Enable Bit.
0: Comparator0 Disabled.
1: Comparator0 Enabled.

Bit6: CP0OUT: Comparator0 Output State Flag.
0: Voltage on CP0+ < CP0–.
1: Voltage on CP0+ > CP0–.

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.

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.

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.

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.

Table 8.3. Special Function Registers* (Continued)

Register	Address	Description	Page No.
FLSCL	0xB6	Flash Scale	93
IE	0xA8	Interrupt Enable	75
IP	0xB8	Interrupt Priority	76
IT01CF	0xE4	INT0/INT1 Configuration Register	79
OSCICL	0xB3	Internal Oscillator Calibration	98
OSCICN	0xB2	Internal Oscillator Control	98
OSCXCN	0xB1	External Oscillator Control	100
P0	0x80	Port 0 Latch	109
P0MDIN	0xF1	Port 0 Input Mode Configuration	109
P0MDOUT	0xA4	Port 0 Output Mode Configuration	110
PCA0CN	0xD8	PCA Control	167
PCA0MD	0xD9	PCA Mode	168
PCA0CPH0	0xFC	PCA Capture 0 High	171
PCA0CPH1	0xEA	PCA Capture 1 High	171
PCA0CPH2	0xEC	PCA Capture 2 High	171
PCA0CPL0	0xFB	PCA Capture 0 Low	171
PCA0CPL1	0xE9	PCA Capture 1 Low	171
PCA0CPL2	0xEB	PCA Capture 2 Low	171
PCA0CPM0	0xDA	PCA Module 0 Mode Register	169
PCA0CPM1	0xDB	PCA Module 1 Mode Register	169
PCA0CPM2	0xDC	PCA Module 2 Mode Register	169
PCA0H	0xFA	PCA Counter High	170
PCA0L	0xF9	PCA Counter Low	170
PCON	0x87	Power Control	81
PSCTL	0x8F	Program Store R/W Control	92
PSW	0xD0	Program Status Word	70
REF0CN	0xD1	Voltage Reference Control	49
RSTSRC	0xEF	Reset Source Configuration/Status	87
SBUF0	0x99	UART 0 Data Buffer	137
SCON0	0x98	UART 0 Control	136
SMB0CF	0xC1	SMBus Configuration	118
SMB0CN	0xC0	SMBus Control	120
SMB0DAT	0xC2	SMBus Data	122
SP	0x81	Stack Pointer	69
TMR2CN	0xC8	Timer/Counter 2 Control	154
TCON	0x88	Timer/Counter Control	147
TH0	0x8C	Timer/Counter 0 High	150

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

Table 8.3. Special Function Registers* (Continued)

Register	Address	Description	Page No.
TH1	0x8D	Timer/Counter 1 High	150
TL0	0x8A	Timer/Counter 0 Low	150
TL1	0x8B	Timer/Counter 1 Low	150
TMOD	0x89	Timer/Counter Mode	148
TMR2RLH	0xCB	Timer/Counter 2 Reload High	154
TMR2RLL	0xCA	Timer/Counter 2 Reload Low	154
TMR2H	0xCD	Timer/Counter 2 High	154
TMR2L	0xCC	Timer/Counter 2 Low	154
XBR0	0xE1	Port I/O Crossbar Control 0	107
XBR1	0xE2	Port I/O Crossbar Control 1	107
XBR2	0xE3	Port I/O Crossbar Control 2	108
0x97, 0xAE, 0xAF, 0xB4, 0xB6, 0xBF, 0xCE, 0xD2, 0xD3, 0xD4, 0xD5, 0xD6, 0xD7, 0xDD, 0xDE, 0xDF, 0xF5		Reserved	
***Note:** SFRs are listed in alphabetical order. All undefined SFR locations are reserved			

8.2.7. Register Descriptions

Following are descriptions of SFRs related to the operation of the CIP-51 System Controller. Reserved bits should not be set to logic l. Future product versions may use these bits to implement new features in which case the reset value of the bit will be logic 0, selecting the feature's default state. Detailed descriptions of the remaining SFRs are included in the sections of the datasheet associated with their corresponding system function.

SFR Definition 8.1. DPL: Data Pointer 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: 0x82

Bits7–0: DPL: Data Pointer Low.
The DPL register is the low byte of the 16-bit DPTR. DPTR is used to access indirectly addressed Flash memory.

SFR Definition 8.9. EIE1: Extended Interrupt Enable 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
—	—	ECP0R	ECP0F	EPCA0	EADC0C	EWADC0	ESMB0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xE6

Bits7–6: UNUSED. Read = 00b. Write = don't care.

Bit5: ECP0R: Enable Comparator0 (CP0) Rising Edge Interrupt.
This bit sets the masking of the CP0 Rising Edge interrupt.
0: Disable CP0 Rising Edge interrupt.
1: Enable interrupt requests generated by the CP0RIF flag.

Bit4: ECP0F: Enable Comparator0 (CP0) Falling Edge Interrupt.
This bit sets the masking of the CP0 Falling Edge interrupt.
0: Disable CP0 Falling Edge interrupt.
1: Enable interrupt requests generated by the CP0FIF flag.

Bit3: EPCA0: Enable Programmable Counter Array (PCA0) Interrupt.
This bit sets the masking of the PCA0 interrupts.
0: Disable all PCA0 interrupts.
1: Enable interrupt requests generated by PCA0.

Bit2: EADC0C: Enable ADC0 Conversion Complete Interrupt.
This bit sets the masking of the ADC0 Conversion Complete interrupt.
0: Disable ADC0 Conversion Complete interrupt.
1: Enable interrupt requests generated by the AD0INT flag.

Bit1: EWADC0: Enable Window Comparison ADC0 Interrupt.
This bit sets the masking of ADC0 Window Comparison interrupt.
0: Disable ADC0 Window Comparison interrupt.
1: Enable interrupt requests generated by ADC0 Window Compare flag.

Bit0: ESMB0: Enable SMBus Interrupt.
This bit sets the masking of the SMBus interrupt.
0: Disable all SMBus interrupts.
1: Enable interrupt requests generated by the SI flag.

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-

viced in priority order after the Flash operation has been completed and interrupts have been re-enabled by software.

10. Make certain that the Flash write and erase pointer variables are not located in XRAM. See your compiler documentation for instructions regarding how to explicitly locate variables in different memory areas.
11. Add address bounds checking to the routines that write or erase Flash memory to ensure that a routine called with an illegal address does not result in modification of the Flash.

10.4.3. 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 12.3. XBR2: Port I/O Crossbar Register 2

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
WEAKPUD	XBARE	—	—	—	T1E	T0E	ECIE	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
								0xE3

Bit7: WEAKPUD: Port I/O Weak Pull-up Disable.
0: Weak Pull-ups enabled (except for Ports whose I/O are configured as push-pull).
1: Weak Pull-ups disabled.
Bit6: XBARE: Crossbar Enable.
0: Crossbar disabled.
1: Crossbar enabled.
Bits5–3: UNUSED: Read = 000b. Write = don’t care.
Bit2: T1E: T1 Enable.
0: T1 unavailable at Port pin.
1: T1 routed to Port pin.
Bit1: T0E: T0 Enable.
0: T0 unavailable at Port pin.
1: T0 routed to Port pin.
Bit0: ECIE: PCA0 Counter Input Enable.
0: ECI unavailable at Port pin.
1: ECI routed to Port pin.

12.3. General Purpose Port I/O

Port pins that remain unassigned by the Crossbar and are not used by analog peripherals can be used for general purpose I/O. Port0 is accessed through a corresponding special function register (SFR) that is both byte addressable and bit addressable. When writing to a Port, the value written to the SFR is latched to maintain the output data value at each pin. When reading, the logic levels of the Port's input pins are returned regardless of the XBRn settings (i.e., even when the pin is assigned to another signal by the Crossbar, the Port register can always read its corresponding Port I/O pin). The exception to this is the execution of the read-modify-write instructions. The read-modify-write instructions when operating on a Port SFR are the following: ANL, ORL, XRL, JBC, CPL, INC, DEC, DJNZ and MOV, CLR or SET, when the destination is an individual bit in a Port SFR. For these instructions, the value of the register (not the pin) is read, modified, and written back to the SFR.

13. SMBus

The SMBus I/O interface is a two-wire bidirectional serial bus. The SMBus is compliant with the System Management Bus Specification, version 1.1, and compatible with the I^2C serial bus. Reads and writes to the interface by the system controller are byte oriented with the SMBus interface autonomously controlling the serial transfer of the data. Data can be transferred at up to 1/20th of the system clock operating as master or slave (this can be faster than allowed by the SMBus specification, depending on the system clock used). A method of extending the clock-low duration is available to accommodate devices with different speed capabilities on the same bus.

The SMBus interface may operate as a master and/or slave, and may function on a bus with multiple masters. The SMBus provides control of SDA (serial data), SCL (serial clock) generation and synchronization, arbitration logic, and START/STOP control and generation. Three SFRs are associated with the SMBus: SMB0CF configures the SMBus; SMB0CN controls the status of the SMBus; and SMB0DAT is the data register, used for both transmitting and receiving SMBus data and slave addresses.

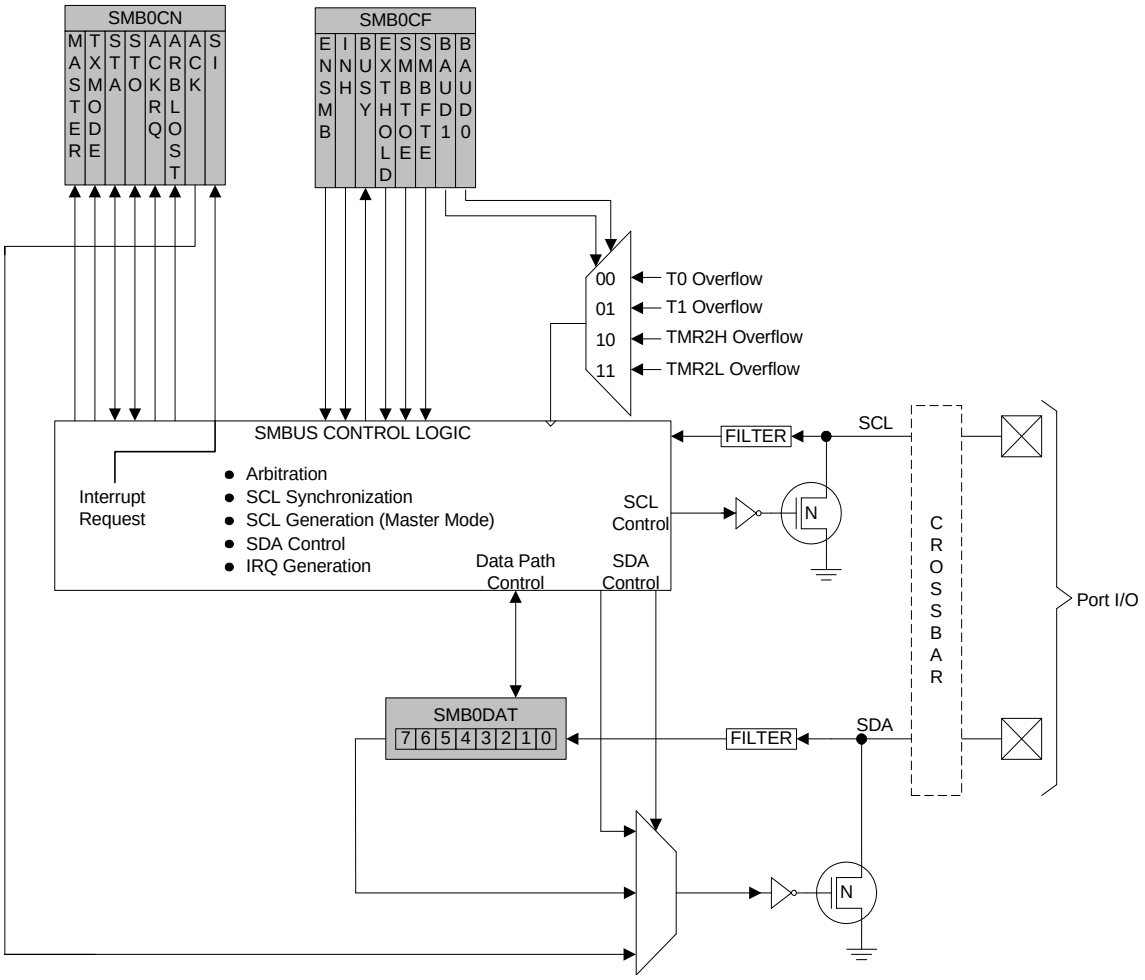

Figure 13.1. SMBus Block Diagram

Table 13.4. SMBus Status Decoding (Continued)

Mode	Values Read				Current SMbus State	Typical Response Options	Values Written		
	Status Vector	ACKRQ	ARBLOST	ACK			STA	STO	ACK
SLAVE RECEIVER	0010	1	0	X	A slave address was received; ACK requested.	Acknowledge received address (received slave address match, R/W bit = READ).	0	0	1
						Do not acknowledge received address.	0	0	0
						Acknowledge received address, and switch to transmitter mode (received slave address match, R/W bit = WRITE); see **Section 13.5.4** for procedure.	0	0	1
		1	1	X	Lost arbitration as master; slave address received; ACK requested.	Acknowledge received address (received slave address match, R/W bit = READ).	0	0	1
						Do not acknowledge received address.	0	0	0
						Acknowledge received address, and switch to transmitter mode (received slave address match, R/W bit = WRITE); see **Section 13.5.4** for procedure.	0	0	1
						Reschedule failed transfer; do not acknowledge received address	1	0	0
	0010	0	1	X	Lost arbitration while attempting a repeated START.	Abort failed transfer.	0	0	X
						Reschedule failed transfer.	1	0	X
	0001	1	1	X	Lost arbitration while attempting a STOP.	No action required (transfer complete/aborted).	0	0	0
		0	0	X	A STOP was detected while addressed as a Slave Transmitter or Slave Receiver.	Clear STO.	0	0	X
		0	1	X	Lost arbitration due to a detected STOP.	Abort transfer.	0	0	X
						Reschedule failed transfer.	1	0	X
	0000	1	0	X	A slave byte was received; ACK requested.	Acknowledge received byte; Read SMB0DAT.	0	0	1
						Do not acknowledge received byte.	0	0	0
		1	1	X	Lost arbitration while transmitting a data byte as master.	Abort failed transfer.	0	0	0
						Reschedule failed transfer.	1	0	0

Table 14.1. Timer Settings for Standard Baud Rates Using The Internal 24.5 MHz Oscillator

	Frequency: 24.5 MHz						
	Target Baud Rate (bps)	Baud Rate % Error	Oscillator Divide Factor	Timer Clock Source	SCA1–SCA0 (pre-scale select)[1]	T1M[1]	Timer 1 Reload Value (hex)
SYSCLK from Internal Osc.	230400	–0.32%	106	SYSCLK	XX[2]	1	0xCB
	115200	–0.32%	212	SYSCLK	XX[2]	1	0x96
	57600	0.15%	426	SYSCLK	XX[2]	1	0x2B
	28800	–0.32%	848	SYSCLK / 4	01	0	0x96
	14400	0.15%	1704	SYSCLK / 12	00	0	0xB9
	9600	–0.32%	2544	SYSCLK / 12	00	0	0x96
	2400	–0.32%	10176	SYSCLK / 48	10	0	0x96
	1200	0.15%	20448	SYSCLK / 48	10	0	0x2B

Notes:
1. SCA1-SCA0 and T1M bit definitions can be found in **Section 15.1**.
2. X = Don't care.

Table 14.2. Timer Settings for Standard Baud Rates Using an External 25 MHz Oscillator

	Frequency: 25.0 MHz						
	Target Baud Rate (bps)	Baud Rate % Error	Oscillator Divide Factor	Timer Clock Source	SCA1–SCA0 (pre-scale select)[1]	T1M[1]	Timer 1 Reload Value (hex)
SYSCLK from External Osc.	230400	–0.47%	108	SYSCLK	XX[2]	1	0xCA
	115200	0.45%	218	SYSCLK	XX[2]	1	0x93
	57600	–0.01%	434	SYSCLK	XX[2]	1	0x27
	28800	0.45%	872	SYSCLK / 4	01	0	0x93
	14400	–0.01%	1736	SYSCLK / 4	01	0	0x27
	9600	0.15%	2608	EXTCLK / 8	11	0	0x5D
	2400	0.45%	10464	SYSCLK / 48	10	0	0x93
	1200	–0.01%	20832	SYSCLK / 48	10	0	0x27
SYSCLK from Internal Osc.	57600	–0.47%	432	EXTCLK / 8	11	0	0xE5
	28800	–0.47%	864	EXTCLK / 8	11	0	0xCA
	14400	0.45%	1744	EXTCLK / 8	11	0	0x93
	9600	0.15%	2608	EXTCLK / 8	11	0	0x5D

Notes:
1. SCA1–SCA0 and T1M bit definitions can be found in **Section 15.1**.
2. X = Don't care

15. Timers

Each MCU includes 3 counter/timers: two are 16-bit counter/timers compatible with those found in the standard 8051, and one is a 16-bit auto-reload timer for use with the ADC, SMBus, or for general purpose use. These timers can be used to measure time intervals, count external events and generate periodic interrupt requests. Timer 0 and Timer 1 are nearly identical and have four primary modes of operation. Timer 2 offers 16-bit and split 8-bit timer functionality with auto-reload.

Timer 0 and Timer 1 Modes:	Timer 2 Modes:
13-bit counter/timer	16-bit timer with auto-reload
16-bit counter/timer	
8-bit counter/timer with auto-reload	Two 8-bit timers with auto-reload
Two 8-bit counter/timers (Timer 0 only)	

Timers 0 and 1 may be clocked by one of five sources, determined by the Timer Mode Select bits (T1M–T0M) and the Clock Scale bits (SCA1–SCA0). The Clock Scale bits define a pre-scaled clock from which Timer 0 and/or Timer 1 may be clocked (See SFR Definition 15.3 for pre-scaled clock selection).

Timer 0/1 may then be configured to use this pre-scaled clock signal or the system clock. Timer 2 may be clocked by the system clock, the system clock divided by 12, or the external oscillator clock source divided by 8.

Timer 0 and Timer 1 may also be operated as counters. When functioning as a counter, a counter/timer register is incremented on each high-to-low transition at the selected input pin. Events with a frequency of up to one-fourth the system clock's frequency can be counted. The input signal need not be periodic, but it should be held at a given level for at least two full system clock cycles to ensure the level is properly sampled.

15.1. Timer 0 and Timer 1

Each timer is implemented as 16-bit register accessed as two separate bytes: a low byte (TL0 or TL1) and a high byte (TH0 or TH1). The Counter/Timer Control register (TCON) is used to enable Timer 0 and Timer 1 as well as indicate their status. Timer 0 interrupts can be enabled by setting the ET0 bit in the IE register (**Section “8.3.5. Interrupt Register Descriptions” on page 75**); Timer 1 interrupts can be enabled by setting the ET1 bit in the IE register (**Section 8.3.5**). Both counter/timers operate in one of four primary modes selected by setting the Mode Select bits T1M1–T0M0 in the Counter/Timer Mode register (TMOD). Each timer can be configured independently. Each operating mode is described below.

15.1.1. Mode 0: 13-bit Counter/Timer

Timer 0 and Timer 1 operate as 13-bit counter/timers in Mode 0. The following describes the configuration and operation of Timer 0. However, both timers operate identically, and Timer 1 is configured in the same manner as described for Timer 0.

The TH0 register holds the eight MSBs of the 13-bit counter/timer. TL0 holds the five LSBs in bit positions TL0.4-TL0.0. The three upper bits of TL0 (TL0.7-TL0.5) are indeterminate and should be masked out or ignored when reading. As the 13-bit timer register increments and overflows from 0x1FFF (all ones) to 0x0000, the timer overflow flag TF0 (TCON.5) is set and an interrupt will occur if Timer 0 interrupts are enabled.

15.1.4. Mode 3: Two 8-bit Counter/Timers (Timer 0 Only)

In Mode 3, Timer 0 is configured as two separate 8-bit counter/timers held in TL0 and TH0. The counter/timer in TL0 is controlled using the Timer 0 control/status bits in TCON and TMOD: TR0, C/T0, GATE0 and TF0. TL0 can use either the system clock or an external input signal as its timebase. The TH0 register is restricted to a timer function sourced by the system clock or prescaled clock. TH0 is enabled using the Timer 1 run control bit TR1. TH0 sets the Timer 1 overflow flag TF1 on overflow and thus controls the Timer 1 interrupt.

Timer 1 is inactive in Mode 3. When Timer 0 is operating in Mode 3, Timer 1 can be operated in Modes 0, 1 or 2, but cannot be clocked by external signals nor set the TF1 flag and generate an interrupt. However, the Timer 1 overflow can be used to generate baud rates for the SMBus and/or UART, and/or initiate ADC conversions. While Timer 0 is operating in Mode 3, Timer 1 run control is handled through its mode settings. To run Timer 1 while Timer 0 is in Mode 3, set the Timer 1 Mode as 0, 1, or 2. To disable Timer 1, configure it for Mode 3.

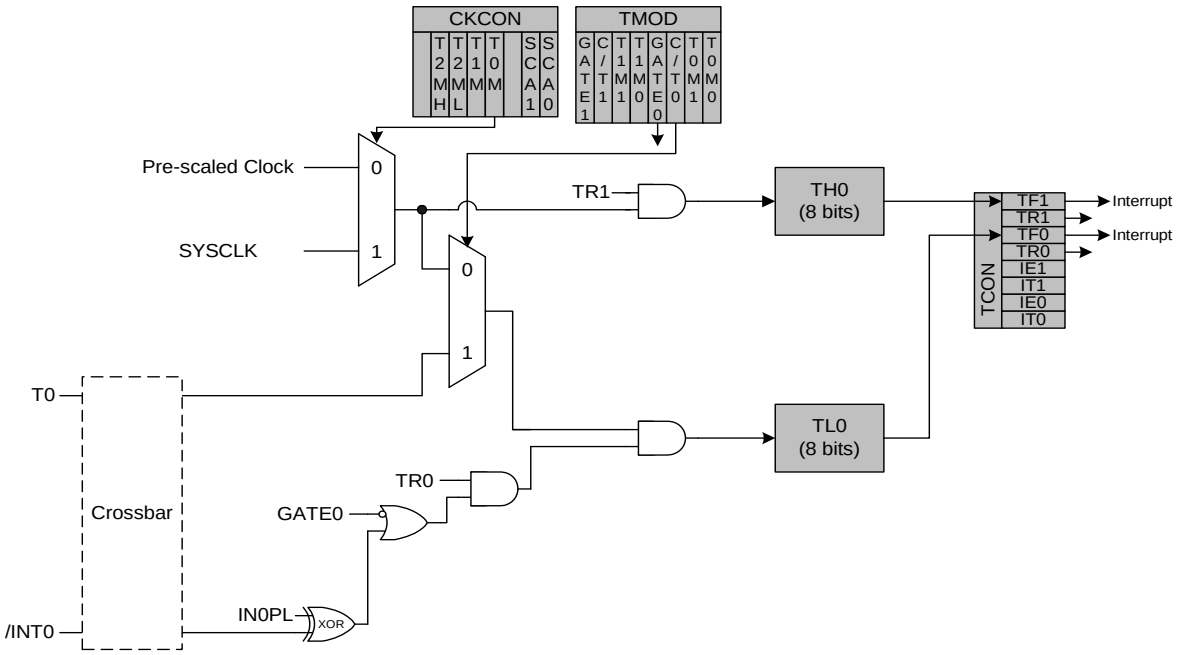

Figure 15.3. T0 Mode 3 Block Diagram

16.2. Capture/Compare Modules

Each module can be configured to operate independently in one of six operation modes: Edge-triggered Capture, Software Timer, High Speed Output, Frequency Output, 8-bit Pulse Width Modulator, or 16-bit Pulse Width Modulator. Each module has Special Function Registers (SFRs) associated with it in the CIP-51 system controller. These registers are used to exchange data with a module and configure the module's mode of operation.

Table 16.2 summarizes the bit settings in the PCA0CPMn registers used to select the PCA capture/compare module's operating modes. Setting the ECCFn bit in a PCA0CPMn register enables the module's CCFn interrupt. Note: PCA0 interrupts must be globally enabled before individual CCFn interrupts are recognized. PCA0 interrupts are globally enabled by setting the EA bit and the EPCA0 bit to logic 1. See Figure 16.3 for details on the PCA interrupt configuration.

Table 16.2. PCA0CPM Register Settings for PCA Capture/Compare Modules

PWM16	ECOM	CAPP	CAPN	MAT	TOG	PWM	ECCF	Operation Mode
X*	X*	1	0	0	0	0	X*	Capture triggered by positive edge on CEXn
X*	X*	0	1	0	0	0	X*	Capture triggered by negative edge on CEXn
X*	X*	1	1	0	0	0	X*	Capture triggered by transition on CEXn
X*	1	0	0	1	0	0	X*	Software Timer
X*	1	0	0	1	1	0	X*	High Speed Output
X*	1	0	0	X*	1	1	X*	Frequency Output
0	1	0	0	X*	0	1	X*	8-bit Pulse Width Modulator
1	1	0	0	X*	0	1	X*	16-bit Pulse Width Modulator

***Note:** X = Don't Care

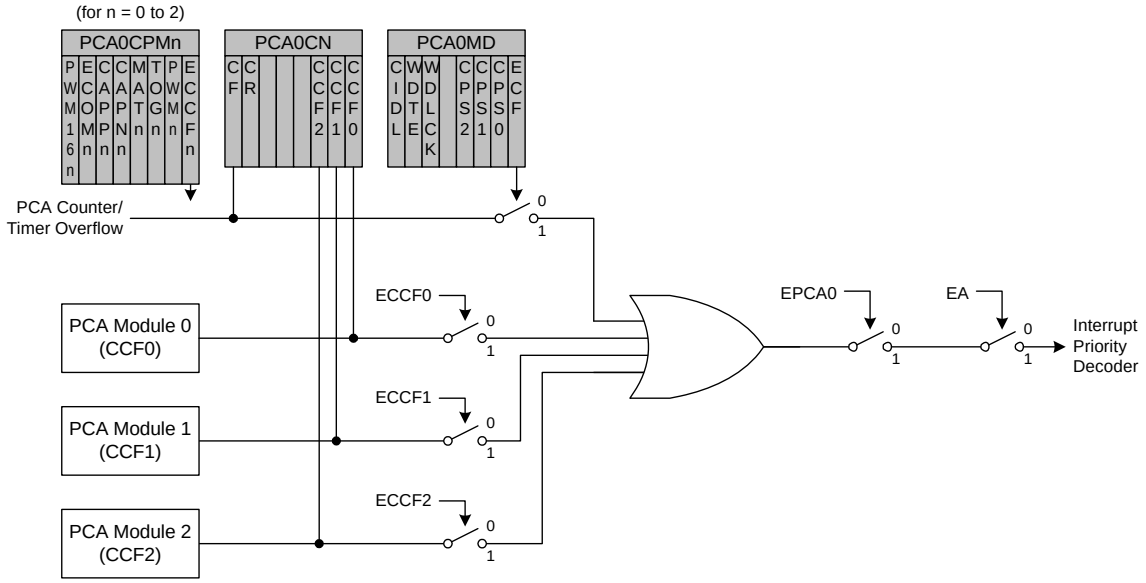

Figure 16.3. PCA Interrupt Block Diagram

16.2.1. Edge-triggered Capture Mode

In this mode, a valid transition on the CEXn pin causes the PCA to capture the value of the PCA counter/timer and copy it into the corresponding module's 16-bit capture/compare register (PCA0CPLn and PCA0CPHn). The CAPPn and CAPNn bits in the PCA0CPMn register are used to select the type of transition that triggers the capture: low-to-high transition (positive edge), high-to-low transition (negative edge), or either transition (positive or negative edge). When a capture 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. If both CAPPn and CAPNn bits are set to logic 1, then the state of the Port pin associated with CEXn can be read directly to determine whether a rising-edge or falling-edge caused the capture.

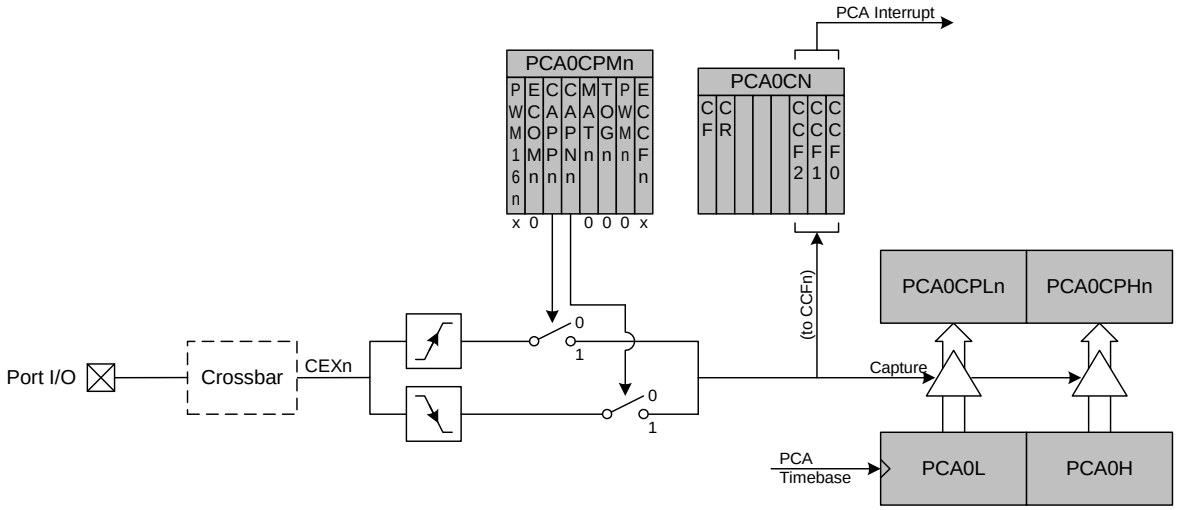

Figure 16.4. PCA Capture Mode Diagram

Note: The CEXn input signal must remain high or low for at least 2 system clock cycles to be recognized by the hardware.