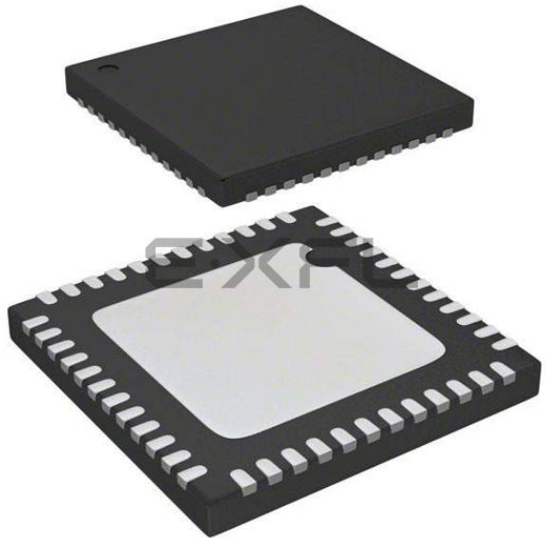

Welcome to E-XFL.COM

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, Temp Sensor, 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	A/D 12x10b
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/c8051f706-gmr

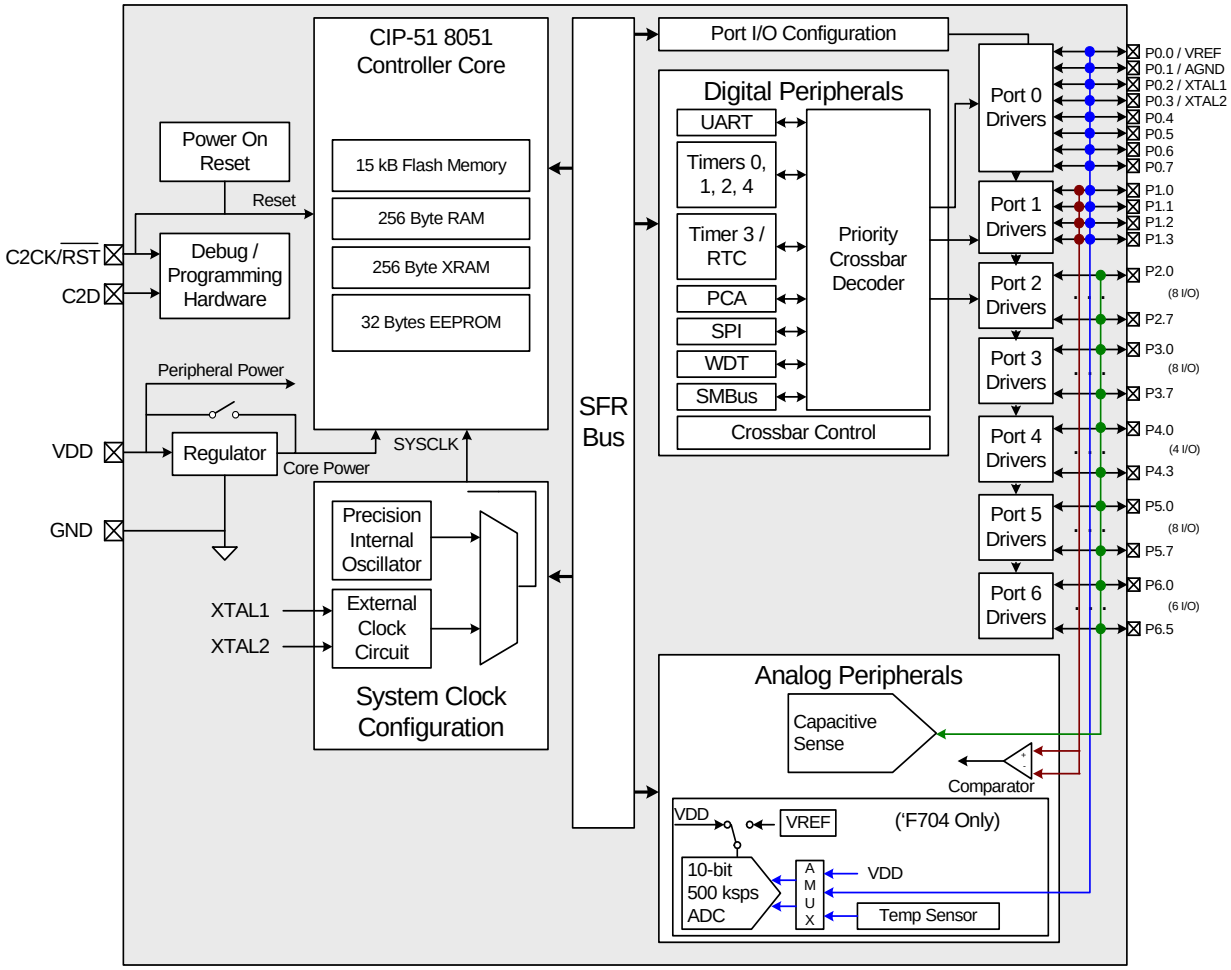

Figure 1.3. C8051F704/5 Block Diagram

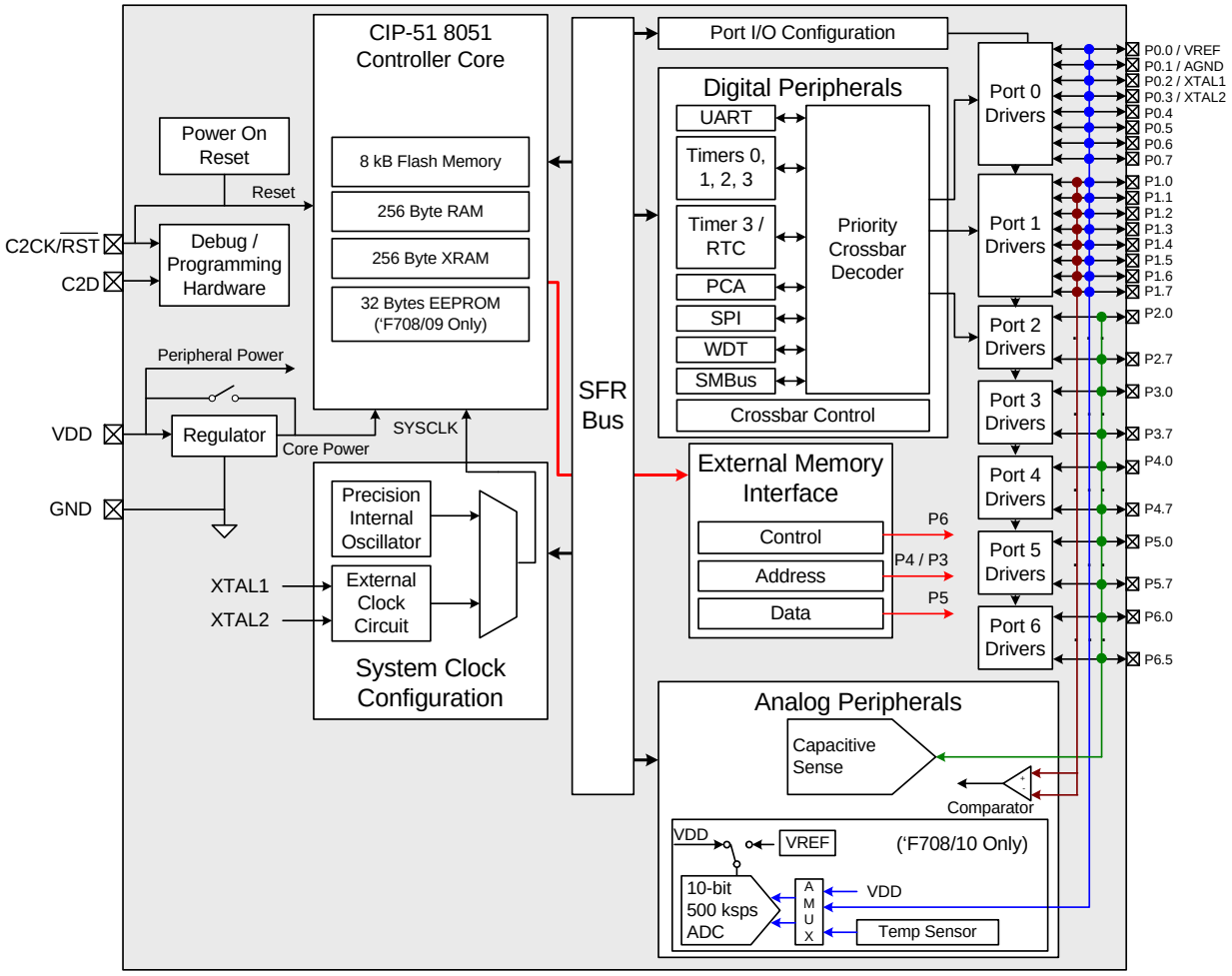

Figure 1.5. C8051F708/09/10/11 Block Diagram

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

Name	TQFP64	TQFP48 QFN48	QFN32	QFN24	Type	Description
P5.1	10	10	7	—	D I/O or A In	Port 5.0. CS0 input pin 26.
P5.2	7	7	6	—	D I/O or A In	Port 5.2. CS0 input pin 27
P5.3	6	6	5	—	D I/O or A In	Port 5.3. CS0 input pin 28.
P5.4	5	5	4	—	D I/O or A In	Port 5.4. CS0 input pin 29.
P5.5	4	4	3	—	D I/O or A In	Port 5.5. CS0 input pin 30.
P5.6	3	3	2	—	D I/O or A In	Port 5.6. CS0 input pin 31.
P5.7	2	2	1	—	D I/O or A In	Port 5.7. CS0 input pin 32.
P6.0	1	—	—	—	D I/O	Port 6.0. CS0 input pin 33.
P6.1	64	—	—	—	D I/O	Port 6.1. CS0 input pin 34.
P6.2	63	—	—	—	D I/O	Port 6.2. CS0 input pin 35.
P6.3	62	1	32	—	D I/O	Port 6.3. CS0 input pin 36.
P6.4	61	48	31	1	D I/O	Port 6.4. CS0 input pin 37.
P6.5	60	47	30	24	D I/O	Port 6.5. CS0 input pin 38.

9.2. Electrical Characteristics

Table 9.2. Global Electrical Characteristics

–40 to +85 °C, 25 MHz system clock unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
Supply Voltage[1]	Regulator in Normal Mode Regulator in Bypass Mode	1.8 1.7	3.0 1.8	3.6 1.9	V V
Digital Supply Current with CPU Active (Normal Mode[2,3])	V_{DD} = 1.8 V, Clock = 25 MHz V_{DD} = 1.8 V, Clock = 1 MHz V_{DD} = 1.8 V, Clock = 32 kHz V_{DD} = 3.0 V, Clock = 25 MHz V_{DD} = 3.0 V, Clock = 1 MHz V_{DD} = 3.0 V, Clock = 32 kHz	— — — — — —	5.0 1.2 175 5.5 1.3 190	6.5 — — 7.0 — —	mA mA µA mA mA µA
Digital Supply Current with CPU Inactive (Idle Mode[2,3])	V_{DD} = 1.8 V, Clock = 25 MHz V_{DD} = 1.8 V, Clock = 1 MHz V_{DD} = 1.8 V, Clock = 32 kHz V_{DD} = 3.0 V, Clock = 25 MHz V_{DD} = 3.0 V, Clock = 1 MHz V_{DD} = 3.0 V, Clock = 32 kHz	— — — — — —	2.5 180 90 3.2 200 110	4.0 — — 4.5 — —	mA µA µA mA µA µA
Digital Supply Current (shutdown)[3]	Stop/suspend mode, Reg On, 25 °C	—	80	90	µA
	Stop/suspend mode, Reg Bypass, 25 °C	—	2	4	µA
Digital Supply RAM Data Retention Voltage		—	1.3	—	V
Specified Operating Temperature Range		–40	—	+85	°C
SYSCLK (system clock frequency)	See Note 3.	0	—	25	MHz
Tsysl (SYSCLK low time)		18	—	—	ns
Tsysh (SYSCLK high time)		18	—	—	ns

Notes:

1. Analog performance is not guaranteed when V_{DD} is below 1.8 V.
2. Includes bias current for internal voltage regulator.
3. SYSCLK must be at least 32 kHz to enable debugging.

10.1. Output Code Formatting

The ADC measures the input voltage with reference to GND. The registers ADC0H and ADC0L contain the high and low bytes of the output conversion code from the ADC at the completion of each conversion. Data can be right-justified or left-justified, depending on the setting of the AD0LJST bit. Conversion codes are represented as 10-bit unsigned integers. Inputs are measured from 0 to VREF x 1023/1024. Example codes are shown below for both right-justified and left-justified data. Unused bits in the ADC0H and ADC0L registers are set to 0.

Input Voltage	Right-Justified ADC0H:ADC0L (AD0LJST = 0)	Left-Justified ADC0H:ADC0L (AD0LJST = 1)
VREF x 1023/1024	0x03FF	0xFFC0
VREF x 512/1024	0x0200	0x8000
VREF x 256/1024	0x0100	0x4000
0	0x0000	0x0000

10.2. 8-Bit Mode

Setting the ADC08BE bit in register ADC0CF to 1 will put the ADC in 8-bit mode. In 8-bit mode, only the 8 MSBs of data are converted, and the ADC0H register holds the results. The AD0LJST bit is ignored for 8-bit mode. 8-bit conversions take two fewer SAR clock cycles than 10-bit conversions, so the conversion is completed faster, and a 500 ksps sampling rate can be achieved with a slower SAR clock.

10.3. Modes of Operation

ADC0 has a maximum conversion speed of 500 ksps. The ADC0 conversion clock is a divided version of the system clock, determined by the AD0SC bits in the ADC0CF register.

10.3.1. Starting a Conversion

A conversion can be initiated in one of six ways, depending on the programmed states of the ADC0 Start of Conversion Mode bits (AD0CM2–0) in register ADC0CN. Conversions may be initiated by one of the following:

1. Writing a 1 to the AD0BUSY bit of register ADC0CN
2. A Timer 0 overflow (i.e., timed continuous conversions)
3. A Timer 2 overflow
4. A Timer 1 overflow
5. A rising edge on the CNVSTR input signal
6. A Timer 3 overflow

Writing a 1 to AD0BUSY provides software control of ADC0 whereby conversions are performed "on-demand". During conversion, the AD0BUSY bit is set to logic 1 and reset to logic 0 when the conversion is complete. The falling edge of AD0BUSY triggers an interrupt (when enabled) and sets the ADC0 interrupt flag (AD0INT). When polling for ADC conversion completions, the ADC0 interrupt flag (AD0INT) should be used. Converted data is available in the ADC0 data registers, ADC0H:ADC0L, when bit AD0INT is logic 1. When Timer 2 or Timer 3 overflows are used as the conversion source, Low Byte overflows are used if Timer 2/3 is in 8-bit mode; High byte overflows are used if Timer 2/3 is in 16-bit mode. See Section “33. Timers” on page 262 for timer configuration.

Important Note About Using CNVSTR: The CNVSTR input pin also functions as a Port I/O pin. When the CNVSTR input is used as the ADC0 conversion source, the associated pin should be skipped by the Digital Crossbar. See Section “28. Port Input/Output” on page 180 for details on Port I/O configuration.

11. Temperature Sensor

An on-chip temperature sensor is included on the C8051F700/2/4/6/8 and C8051F710/2/4/6 which can be directly accessed via the ADC multiplexer in single-ended configuration. To use the ADC to measure the temperature sensor, the ADC mux channel should be configured to connect to the temperature sensor. The temperature sensor transfer function is shown in Figure 11.1. The output voltage (V_{TEMP}) is the positive ADC input when the ADC multiplexer is set correctly. The TEMPE bit in register REF0CN enables/disables the temperature sensor, as described in SFR Definition 12.1. While disabled, the temperature sensor defaults to a high impedance state and any ADC measurements performed on the sensor will result in meaningless data. Refer to Table 9.12 for the slope and offset parameters of the temperature sensor.

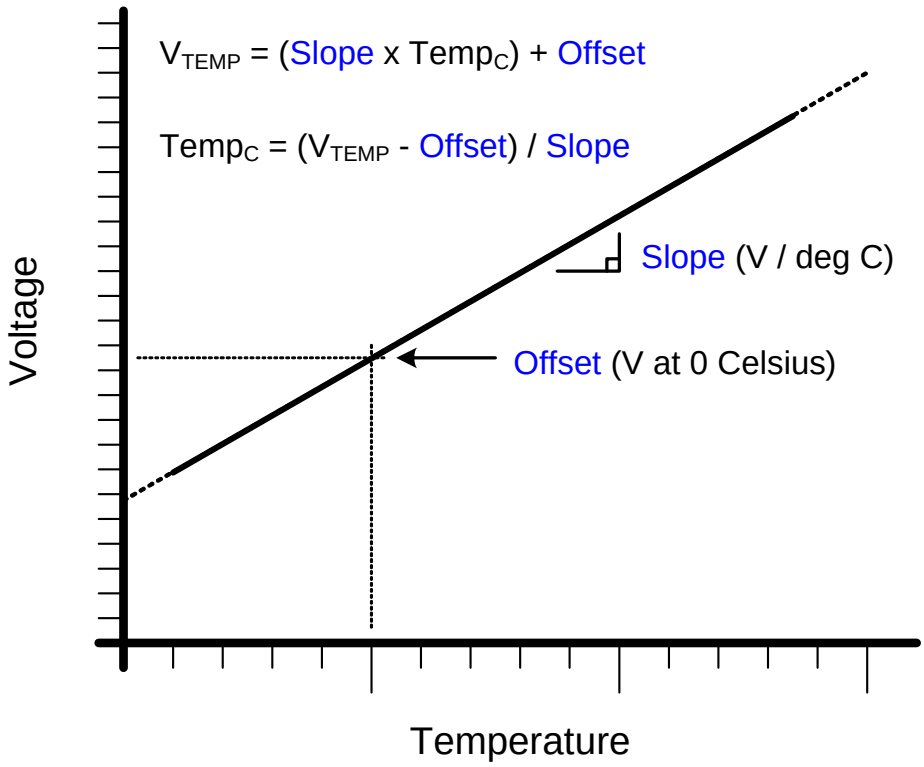

Figure 11.1. Temperature Sensor Transfer Function

11.1. Calibration

The uncalibrated temperature sensor output is extremely linear and suitable for relative temperature measurements (see Table 5.1 for linearity specifications). For absolute temperature measurements, offset and/or gain calibration is recommended. Typically a 1-point (offset) calibration includes the following steps:

1. Control/measure the ambient temperature (this temperature must be known).
2. Power the device, and delay for a few seconds to allow for self-heating.
3. Perform an ADC conversion with the temperature sensor selected as the ADC's input.
4. Calculate the offset characteristics, and store this value in non-volatile memory for use with subsequent temperature sensor measurements.

Figure 5.3 shows the typical temperature sensor error assuming a 1-point calibration at 0 °C.

SFR Definition 13.1. REG0CN: Voltage Regulator Control

Bit	7	6	5	4	3	2	1	0
Name	STOPCF	BYPASS						
Type	R/W	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 = 0xB9; SFR Page = F

Bit	Name	Function
7	STOPCF	**Stop Mode Configuration.** This bit configures the regulator's behavior when the device enters STOP mode. 0: Regulator is still active in STOP mode. Any enabled reset source will reset the device. 1: Regulator is shut down in STOP mode. Only the $\overline{\text{RST}}$ pin or power cycle can reset the device.
6	BYPASS	**Bypass Internal Regulator.** This bit places the regulator in bypass mode, allowing the core to run directly from the V_{DD} supply pin. 0: Normal Mode—Regulator is on and regulates V_{DD} down to the core voltage. 1: Bypass Mode—Regulator is in bypass mode, and the microcontroller core operates directly from the V_{DD} supply voltage. **IMPORTANT: Bypass mode is for use with an external regulator as the supply voltage only. Never place the regulator in bypass mode when the V_{DD} supply voltage is greater than the specifications given in Table 9.1 on page 47. Doing so may cause permanent damage to the device.**
5:0	Reserved	Reserved. Must Write 000000b.

SFR Definition 15.3. CS0DH: Capacitive Sense Data High Byte

Bit	7	6	5	4	3	2	1	0
Name	CS0DH[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAA; SFR Page = 0

Bit	Name	Description
7:0	CS0DH	**CS0 Data High Byte.** Stores the high byte of the last completed 16-bit Capacitive Sense conversion.

SFR Definition 15.4. CS0DL: Capacitive Sense Data Low Byte

Bit	7	6	5	4	3	2	1	0
Name	CS0DL[7:0]							
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xA9; SFR Page = 0

Bit	Name	Description
7:0	CS0DL	**CS0 Data Low Byte.** Stores the low byte of the last completed 16-bit Capacitive Sense conversion.

Table 20.1. Special Function Register (SFR) Memory Map

Addr	SFR Page	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
F8	0 F	**SPI0CN**	PCA0L P0DRV	PCA0H P1DRV	PCA0CPL0 P2DRV	PCA0CPH0 P3DRV	P4DRV	P5DRV	**VDM0CN**
F0	0 F	**B**	P0MDIN	P1MDIN	P0MAT P2MDIN	P0MASK P3MDIN	P4MDIN	P5MDIN	P6MDIN
E8	0 F	**ADC0CN**	PCA0CPL1	PCA0CPH1	PCA0CPL2	PCA0CPH2 DERIVID	PCA0MD	EMI0TC	**RSTSRC**
E0	0 F	**ACC**	P1MAT XBR0	P1MASK XBR1	**WDTCN**	IT01CF		**EIE1**	**EIE2**
D8	0 F	**PCA0CN**	CRC0DATA	PCA0CPM0	PCA0CPM1	PCA0CPM2			
D0	0 F	**PSW**	**EEDATA**	REF0CN		P0SKIP	P1SKIP	P2SKIP	
C8	0 F	**TMR2CN**		TMR2RLL	TMR2RLH	TMR2L	TMR2H	EIP1	EIP2
C0	0 F	**SMB0CN**	SMB0CF P6DRV	SMB0DAT	ADC0GTL	ADC0GTH HWID	ADC0LTL EECNTL	ADC0LTH EEKEY	EMI0CF
B8	0 F	**IP**	REG0CN	SMB0ADR	ADC0MX SMB0ADM	ADC0CF	ADC0L CLKSEL	ADC0H CS0MD2	OSCICL
B0	0 F	**P3**		**P6**	**P5**		OSCXCN	**EEADDR**	**FLKEY**
A8	0 F	**IE**	CS0DL OSCICN	CS0DH EMI0CN		**P4**	CS0MD1 REVID		P3MDOUT
A0	0 F	**P2**	SPI0CFG PCA0PWM	SPI0CKR	SPI0DAT	P0MDOUT	P1MDOUT	P2MDOUT	**SFRPAGE**
98	0 F	**SCON0**	**SBUF0**	CS0CN P4MDOUT	CPT0CN P5MDOUT	CS0MX P6MDOUT	CPT0MD	CS0CF	CPT0MX CS0PM
90	0 F	**P1**	TMR3CN CRC0CN	TMR3RLL CS0SS	TMR3RLH CS0SE	TMR3L CRC0IN	TMR3H CRC0FLIP	CS0THL CRC0AUTO	CS0THH CRC0CNT
88	0 F	**TCON**	**TMOD**	**TL0**	**TL1**	**TH0**	**TH1**	**CKCON**	**PSCTL**
80	0 F	**P0**	**SP**	**DPL**	**DPH**				**PCON**
		0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)

Notes:

1. SFR addresses ending in 0x0 or 0x8 (leftmost column) are bit-addressable.
2. SFRs indicated with bold lettering and shaded cells are available on both SFR Page 0 and F.

21.2. Interrupt Register Descriptions

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

SFR Definition 21.1. IE: Interrupt Enable

Bit	7	6	5	4	3	2	1	0
Name	EA	ESPI0	ET2	ES0	ET1	EX1	ET0	EX0
Type	R/W	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 = 0xA8; SFR Page = All Pages; Bit-Addressable

Bit	Name	Function
7	EA	**Enable All Interrupts.** Globally enables/disables all interrupts. It overrides individual interrupt mask settings. 0: Disable all interrupt sources. 1: Enable each interrupt according to its individual mask setting.
6	ESPI0	**Enable Serial Peripheral Interface (SPI0) Interrupt.** This bit sets the masking of the SPI0 interrupts. 0: Disable all SPI0 interrupts. 1: Enable interrupt requests generated by SPI0.
5	ET2	**Enable Timer 2 Interrupt.** This bit sets the masking of the Timer 2 interrupt. 0: Disable Timer 2 interrupt. 1: Enable interrupt requests generated by the TF2L or TF2H flags.
4	ES0	**Enable UART0 Interrupt.** This bit sets the masking of the UART0 interrupt. 0: Disable UART0 interrupt. 1: Enable UART0 interrupt.
3	ET1	**Enable Timer 1 Interrupt.** This bit sets the masking of the Timer 1 interrupt. 0: Disable all Timer 1 interrupt. 1: Enable interrupt requests generated by the TF1 flag.
2	EX1	**Enable External Interrupt 1.** This bit sets the masking of External Interrupt 1. 0: Disable external interrupt 1. 1: Enable interrupt requests generated by the $\overline{\text{INT1}}$ input.
1	ET0	**Enable Timer 0 Interrupt.** This bit sets the masking of the Timer 0 interrupt. 0: Disable all Timer 0 interrupt. 1: Enable interrupt requests generated by the TF0 flag.
0	EX0	**Enable External Interrupt 0.** This bit sets the masking of External Interrupt 0. 0: Disable external interrupt 0. 1: Enable interrupt requests generated by the $\overline{\text{INT0}}$ input.

SFR Definition 21.5. EIP1: Extended Interrupt Priority 1

Bit	7	6	5	4	3	2	1	0
Name	PT3	Reserved	PCP0	PPCA0	PADC0	PWADC0	PMAT	PSMB0
Type	R/W	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 = 0xCE; SFR Page = F

Bit	Name	Function
7	PT3	**Timer 3 Interrupt Priority Control.** This bit sets the priority of the Timer 3 interrupt. 0: Timer 3 interrupt set to low priority level. 1: Timer 3 interrupt set to high priority level.
6	Reserved	Must write 0b.
5	PCP0	**Comparator0 (CP0) Interrupt Priority Control.** This bit sets the priority of the CP0 rising edge or falling edge interrupt. 0: CP0 interrupt set to low priority level. 1: CP0 interrupt set to high priority level.
4	PPCA0	**Programmable Counter Array (PCA0) Interrupt Priority Control.** This bit sets the priority of the PCA0 interrupt. 0: PCA0 interrupt set to low priority level. 1: PCA0 interrupt set to high priority level.
3	PADC0	**ADC0 Conversion Complete Interrupt Priority Control.** This bit sets the priority of the ADC0 Conversion Complete interrupt. 0: ADC0 Conversion Complete interrupt set to low priority level. 1: ADC0 Conversion Complete interrupt set to high priority level.
2	PWADC0	**ADC0 Window Comparator Interrupt Priority Control.** This bit sets the priority of the ADC0 Window interrupt. 0: ADC0 Window interrupt set to low priority level. 1: ADC0 Window interrupt set to high priority level.
1	PMAT	**Port Match Interrupt Priority Control.** This bit sets the priority of the Port Match Event interrupt. 0: Port Match interrupt set to low priority level. 1: Port Match interrupt set to high priority level.
0	PSMB0	**SMBus (SMB0) Interrupt Priority Control.** This bit sets the priority of the SMB0 interrupt. 0: SMB0 interrupt set to low priority level. 1: SMB0 interrupt set to high priority level.

27.2. Programmable Internal High-Frequency (H-F) Oscillator

All C8051F70x/71x devices include a programmable internal high-frequency oscillator that defaults as the system clock after a system reset. The internal oscillator period can be adjusted via the OSCICL register as defined by SFR Definition 27.2.

On C8051F70x/71x devices, OSCICL is factory calibrated to obtain a 24.5 MHz base frequency.

The internal oscillator output frequency may be divided by 1, 2, 4, or 8, as defined by the IFCN bits in register OSCICN. The divide value defaults to 8 following a reset.

The precision oscillator supports a spread spectrum mode which modulates the output frequency in order to reduce the EMI generated by the system. When enabled (SSE = 1), the oscillator output frequency is modulated by a stepped triangle wave whose frequency is equal to the oscillator frequency divided by 384 (63.8 kHz using the factory calibration). The maximum deviation from the center frequency is ±0.75%. The output frequency updates occur every 32 cycles and the step size is typically 0.25% of the center frequency.

SFR Definition 27.2. OSCICL: Internal H-F Oscillator Calibration

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

SFR Address = 0xBF; SFR Page = F

Bit	Name	Function
6:0	OSCICL[7:0]	**Internal Oscillator Calibration Bits.** These bits determine the internal oscillator period. When set to 00000000b, the H-F oscillator operates at its fastest setting. When set to 11111111b, the H-F oscillator operates at its slowest setting. The reset value is factory calibrated to generate an internal oscillator frequency of 24.5 MHz.

28.5. Port Match

Port match functionality allows system events to be triggered by a logic value change on P0 or P1. A software controlled value stored in the PnMATCH registers specifies the expected or normal logic values of P0 and P1. A Port mismatch event occurs if the logic levels of the Port's input pins no longer match the software controlled value. This allows Software to be notified if a certain change or pattern occurs on P0 or P1 input pins regardless of the XBRn settings.

The PnMASK registers can be used to individually select which P0 and P1 pins should be compared against the PnMATCH registers. A Port mismatch event is generated if (P0 & P0MASK) does not equal (P0MATCH & P0MASK) or if (P1 & P1MASK) does not equal (P1MATCH & P1MASK).

A Port mismatch event may be used to generate an interrupt or wake the device from a low power mode, such as IDLE or SUSPEND. See the Interrupts and Power Options chapters for more details on interrupt and wake-up sources.

SFR Definition 28.3. P0MASK: Port 0 Mask Register

Bit	7	6	5	4	3	2	1	0
Name	P0MASK[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF4; SFR Page = 0

Bit	Name	Function
7:0	P0MASK[7:0]	**Port 0 Mask Value.** Selects P0 pins to be compared to the corresponding bits in P0MAT. 0: P0.n pin logic value is ignored and cannot cause a Port Mismatch event. 1: P0.n pin logic value is compared to P0MAT.n.

30.5.3. Write Sequence (Slave)

During a write sequence, an SMBus master writes data to a slave device. The slave in this transfer will be a receiver during the address byte, and a receiver during all data bytes. When slave events are enabled (INH = 0), the interface enters Slave Receiver Mode when a START followed by a slave address and direction bit (WRITE in this case) is received. If hardware ACK generation is disabled, upon entering Slave Receiver Mode, an interrupt is generated and the ACKRQ bit is set. The software must respond to the received slave address with an ACK, or ignore the received slave address with a NACK. If hardware ACK generation is enabled, the hardware will apply the ACK for a slave address which matches the criteria set up by SMB0ADR and SMB0ADM. The interrupt will occur after the ACK cycle.

If the received slave address is ignored (by software or hardware), slave interrupts will be inhibited until the next START is detected. If the received slave address is acknowledged, zero or more data bytes are received.

If hardware ACK generation is disabled, the ACKRQ is set to 1 and an interrupt is generated after each received byte. Software must write the ACK bit at that time to ACK or NACK the received byte.

With hardware ACK generation enabled, the SMBus hardware will automatically generate the ACK/NACK, and then post the interrupt. **It is important to note that the appropriate ACK or NACK value should be set up by the software prior to receiving the byte when hardware ACK generation is enabled.**

The interface exits Slave Receiver Mode after receiving a STOP. The interface will switch to Slave Transmitter Mode if SMB0DAT is written while an active Slave Receiver. Figure 30.7 shows a typical slave write sequence. Two received data bytes are shown, though any number of bytes may be received. Notice that the ‘data byte transferred’ interrupts occur at different places in the sequence, depending on whether hardware ACK generation is enabled. The interrupt occurs **before** the ACK with hardware ACK generation disabled, and **after** the ACK when hardware ACK generation is enabled.

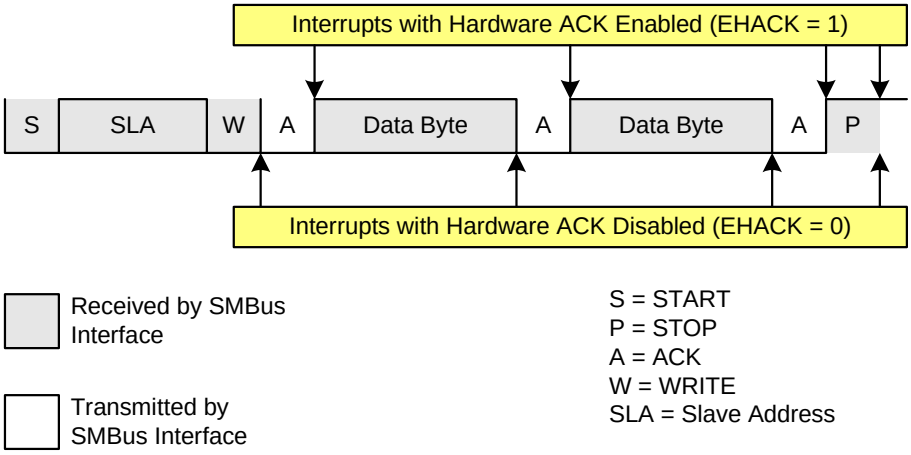

Figure 30.7. Typical Slave Write Sequence

31. Enhanced Serial Peripheral Interface (SPI0)

The Enhanced Serial Peripheral Interface (SPI0) provides access to a flexible, full-duplex synchronous serial bus. SPI0 can operate as a master or slave device in both 3-wire or 4-wire modes, and supports multiple masters and slaves on a single SPI bus. The slave-select (NSS) signal can be configured as an input to select SPI0 in slave mode, or to disable Master Mode operation in a multi-master environment, avoiding contention on the SPI bus when more than one master attempts simultaneous data transfers. NSS can also be configured as a chip-select output in master mode, or disabled for 3-wire operation. Additional general purpose port I/O pins can be used to select multiple slave devices in master mode.

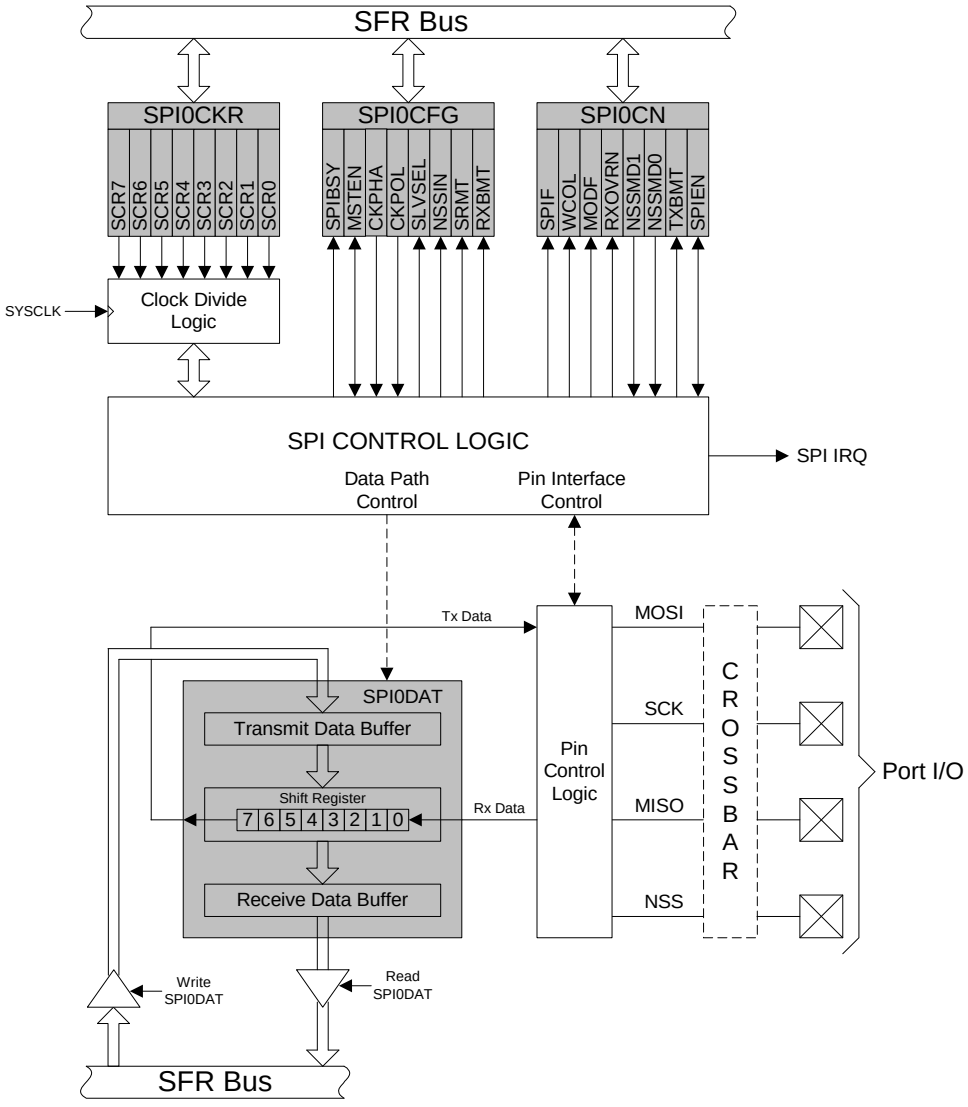

Figure 31.1. SPI Block Diagram

SFR Definition 33.1. CKCON: Clock Control

Bit	7	6	5	4	3	2	1	0
Name	T3MH	T3ML	T2MH	T2ML	T1M	T0M	SCA[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 = 0x8E; SFR Page = All Pages

Bit	Name	Function
7	T3MH	**Timer 3 High Byte Clock Select.** Selects the clock supplied to the Timer 3 high byte (split 8-bit timer mode only). 0: Timer 3 high byte uses the clock defined by the T3XCLK bit in TMR3CN. 1: Timer 3 high byte uses the system clock.
6	T3ML	**Timer 3 Low Byte Clock Select.** Selects the clock supplied to Timer 3. Selects the clock supplied to the lower 8-bit timer in split 8-bit timer mode. 0: Timer 3 low byte uses the clock defined by the T3XCLK bit in TMR3CN. 1: Timer 3 low byte uses the system clock.
5	T2MH	**Timer 2 High Byte Clock Select.** Selects the clock supplied to the Timer 2 high byte (split 8-bit timer mode only). 0: Timer 2 high byte uses the clock defined by the T2XCLK bit in TMR2CN. 1: Timer 2 high byte uses the system clock.
4	T2ML	**Timer 2 Low Byte Clock Select.** 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.
3	T1	**Timer 1 Clock Select.** Selects the clock source supplied to Timer 1. Ignored when C/T1 is set to 1. 0: Timer 1 uses the clock defined by the prescale bits SCA[1:0]. 1: Timer 1 uses the system clock.
2	T0	**Timer 0 Clock Select.** Selects the clock source supplied to Timer 0. Ignored when C/T0 is set to 1. 0: Counter/Timer 0 uses the clock defined by the prescale bits SCA[1:0]. 1: Counter/Timer 0 uses the system clock.
1:0	SCA[1:0]	**Timer 0/1 Prescale Bits.** These bits control the Timer 0/1 Clock Prescaler: 00: System clock divided by 12 01: System clock divided by 4 10: System clock divided by 48 11: External clock divided by 8 (synchronized with the system clock)

SFR Definition 33.2. TCON: Timer Control

Bit	7	6	5	4	3	2	1	0
Name	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
Type	R/W	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 = 0x88; SFR Page = All Pages; Bit-Addressable

Bit	Name	Function
7	TF1	**Timer 1 Overflow Flag.** Set to 1 by hardware when Timer 1 overflows. This flag can be cleared by software but is automatically cleared when the CPU vectors to the Timer 1 interrupt service routine.
6	TR1	**Timer 1 Run Control.** Timer 1 is enabled by setting this bit to 1.
5	TF0	**Timer 0 Overflow Flag.** Set to 1 by hardware when Timer 0 overflows. This flag can be cleared by software but is automatically cleared when the CPU vectors to the Timer 0 interrupt service routine.
4	TR0	**Timer 0 Run Control.** Timer 0 is enabled by setting this bit to 1.
3	IE1	**External Interrupt 1.** This flag is set by hardware when an edge/level of type defined by IT1 is detected. It can be cleared by software but is automatically cleared when the CPU vectors to the External Interrupt 1 service routine in edge-triggered mode.
2	IT1	**Interrupt 1 Type Select.** This bit selects whether the configured /INT1 interrupt will be edge or level sensitive. /INT1 is configured active low or high by the IN1PL bit in the IT01CF register (see SFR Definition 21.7). 0: /INT1 is level triggered. 1: /INT1 is edge triggered.
1	IE0	**External Interrupt 0.** This flag is set by hardware when an edge/level of type defined by IT1 is detected. It can be cleared by software but is automatically cleared when the CPU vectors to the External Interrupt 0 service routine in edge-triggered mode.
0	IT0	**Interrupt 0 Type Select.** This bit selects whether the configured $\overline{\text{INT0}}$ interrupt will be edge or level sensitive. $\overline{\text{INT0}}$ is configured active low or high by the IN0PL bit in register IT01CF (see SFR Definition 21.7). 0: $\overline{\text{INT0}}$ is level triggered. 1: $\overline{\text{INT0}}$ is edge triggered.

33.3.3. Comparator 0 Capture Mode

The capture mode in Timer 3 allows Comparator 0 rising edges to be captured with the timer clocking from the system clock or the system clock divided by 12. Timer 3 capture mode is enabled by setting TF3CEN to 1 and T3SPLIT to 0.

When capture mode is enabled, a capture event will be generated on every Comparator 0 rising edge. When the capture event occurs, the contents of Timer 3 (TMR3H:TMR3L) are loaded into the Timer 3 reload registers (TMR3RLH:TMR3RLL) and the TF3H flag is set (triggering an interrupt if Timer 3 interrupts are enabled). By recording the difference between two successive timer capture values, the Comparator 0 period can be determined with respect to the Timer 3 clock. The Timer 3 clock should be much faster than the capture clock to achieve an accurate reading.

This mode allows software to determine the time between consecutive Comparator 0 rising edges, which can be used for detecting changes in the capacitance of a capacitive switch, or measuring the frequency of a low-level analog signal.

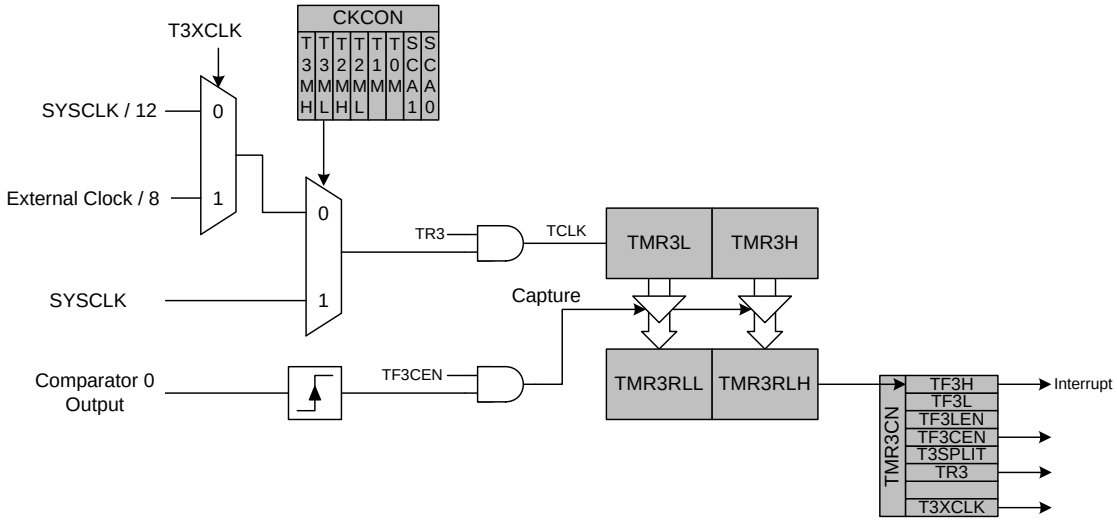

Figure 33.9. Timer 3 Capture Mode Block Diagram

34.2. PCA0 Interrupt Sources

Figure 34.3 shows a diagram of the PCA interrupt tree. There are eight independent event flags that can be used to generate a PCA0 interrupt. They are: the main PCA counter overflow flag (CF), which is set upon a 16-bit overflow of the PCA0 counter, an intermediate overflow flag (COVF), which can be set on an overflow from the 8th, 9th, 10th, or 11th bit of the PCA0 counter, and the individual flags for each PCA channel (CCF0, CCF1, and CCF2), which are set according to the operation mode of that module. These event flags are always set when the trigger condition occurs. Each of these flags can be individually selected to generate a PCA0 interrupt, using the corresponding interrupt enable flag (ECF for CF, ECOV for COVF, and ECCFn for each CCFn). PCA0 interrupts must be globally enabled before any individual interrupt sources are recognized by the processor. PCA0 interrupts are globally enabled by setting the EA bit and the EPCA0 bit to logic 1.

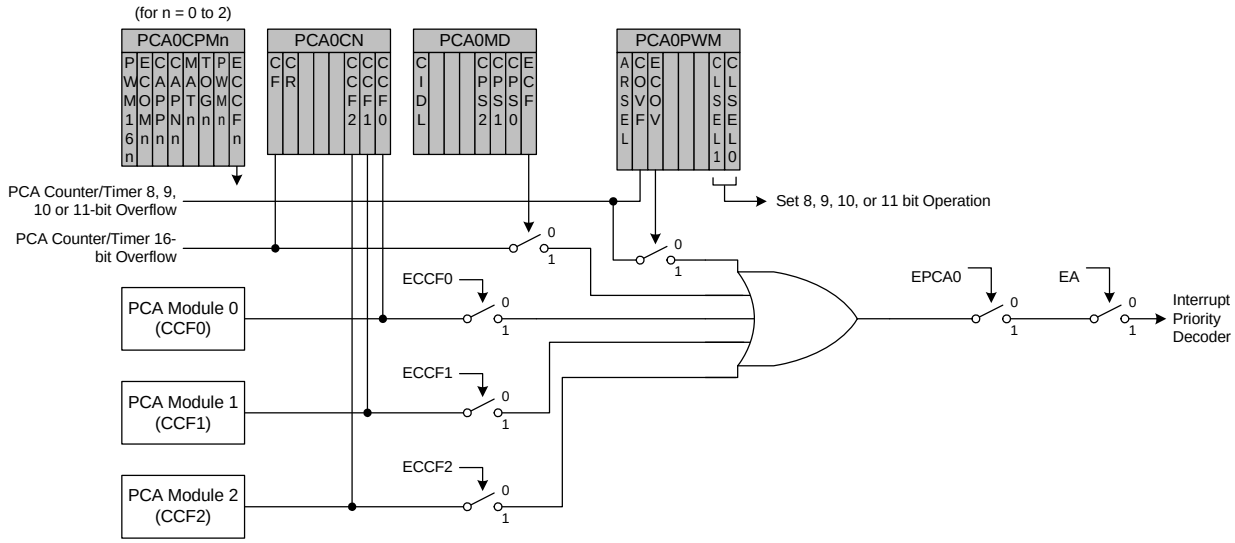

Figure 34.3. PCA Interrupt Block Diagram

34.3. 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 to 11-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 34.2 summarizes the bit settings in the PCA0CPMn and PCA0PWM registers used to select the PCA capture/compare module's operating mode. All modules set to use 8, 9, 10, or 11-bit PWM mode must use the same cycle length (8-11 bits). Setting the ECCFn bit in a PCA0CPMn register enables the module's CCFn interrupt.

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.