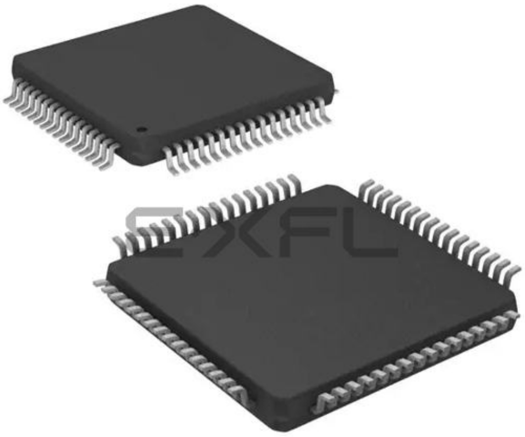


Welcome to E-XFL.COM

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	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	32
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1.25K x 8
Voltage - Supply (Vcc/Vdd)	2.8V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f018r

1.7. Analog to Digital Converter

The C8051F018/9 have an on-chip 10-bit SAR ADC with a 9-channel input multiplexer. With a maximum throughput of 100ksps, the ADC offers true 10-bit accuracy with an INL of ±1LSB. The ADC has a maximum throughput of 100ksps. There is also an on-board 15ppm voltage reference, or an external reference may be used via the VREF pin.

The ADC is under full control of the CIP-51 microcontroller via the Special Function Registers. One input channel is tied to an internal temperature sensor, while the other eight channels are available externally. Each pair of the eight external input channels can be configured as either two single-ended inputs or a single differential input. The system controller can also put the ADC into shutdown to save power.

Conversions can be started in four ways; a software command, an overflow on Timer 2, an overflow on Timer 3, or an external signal input. This flexibility allows the start of conversion to be triggered by software events, external HW signals, or convert continuously. A completed conversion causes an interrupt, or a status bit can be polled in software to determine the end of conversion. The resulting 10-bit data word is latched into two SFRs upon completion of a conversion. The data can be right or left justified in these registers under software control.

Compare registers for the ADC data can be configured to interrupt the controller when ADC data is within a specified window. The ADC can monitor a key voltage continuously in background mode, but not interrupt the controller unless the converted data is within the specified window.

Figure 1.9. ADC Diagram

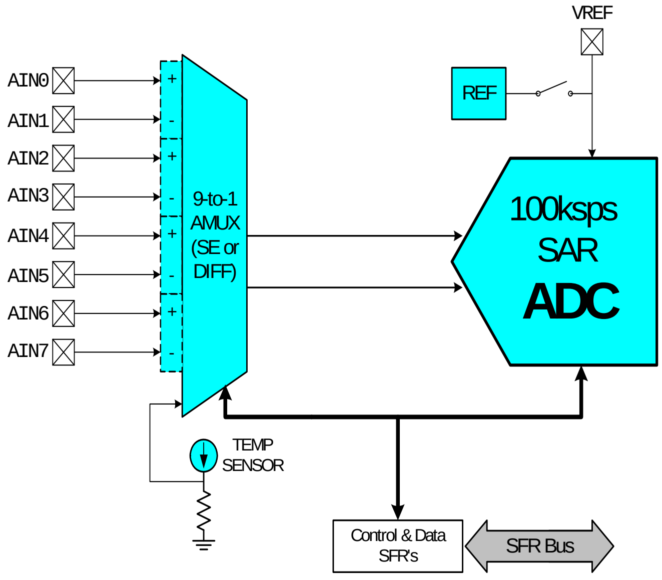
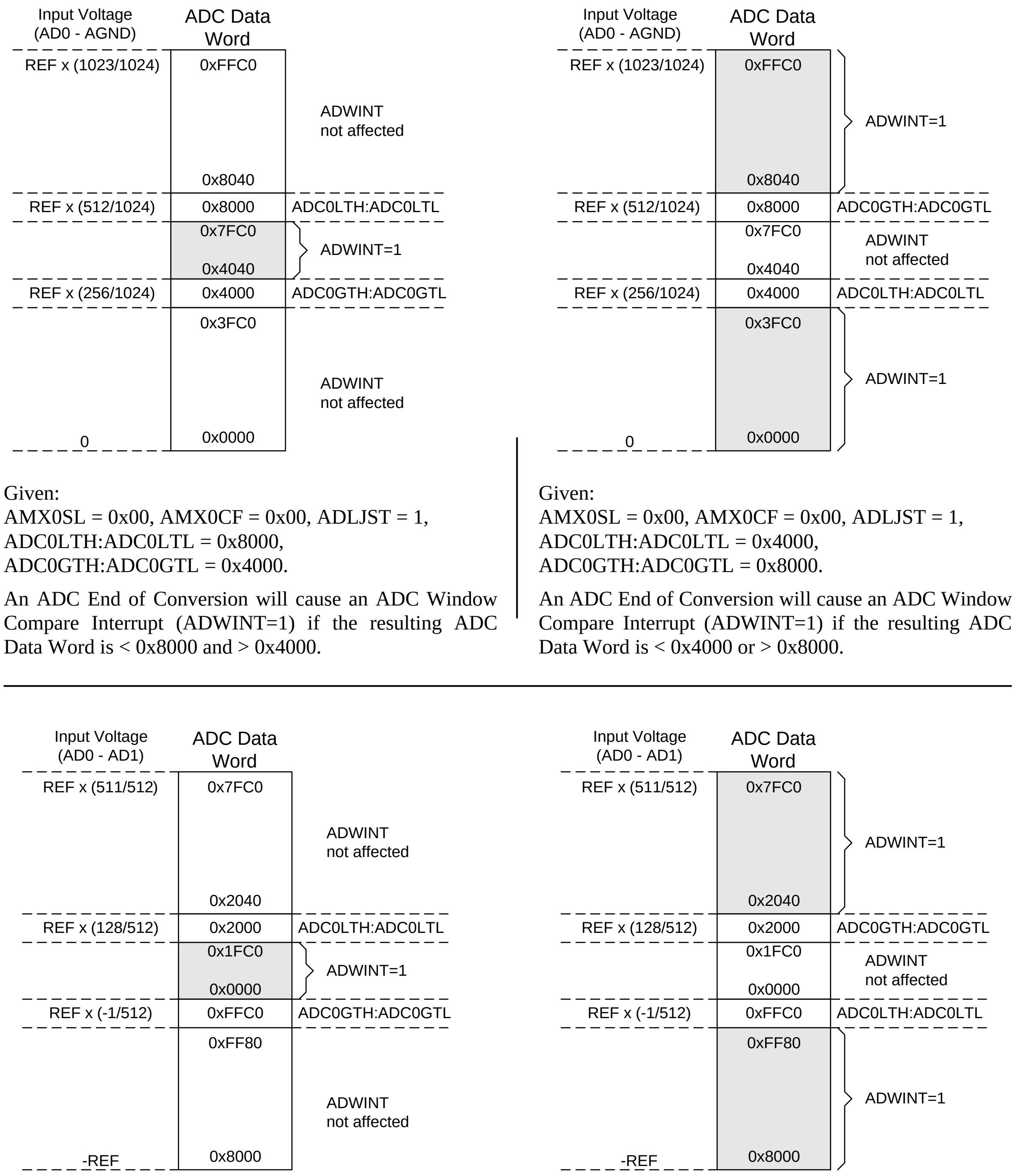


Figure 5.15. 10-Bit ADC Window Interrupt Examples, Left Justified Data



Given:
AMX0SL = 0x00, AMX0CF = 0x00, ADLJST = 1,
ADC0LTH:ADC0LTL = 0x8000,
ADC0GTH:ADC0GTL = 0x4000.

An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0x8000 and > 0x4000.

Given:
AMX0SL = 0x00, AMX0CF = 0x00, ADLJST = 1,
ADC0LTH:ADC0LTL = 0x4000,
ADC0GTH:ADC0GTL = 0x8000.

An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0x4000 or > 0x8000.

Given:

AMX0SL = 0x00, AMX0CF = 0x01, ADLJST = 1,
ADC0LTH:ADC0LTL = 0x2000,
ADC0GTH:ADC0GTL = 0xFFC0.

An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0x2000 and > 0xFFC0. (Two's Complement math.)

Given:

AMX0SL = 0x00, AMX0CF = 0x01, ADLJST = 1,
ADC0LTH:ADC0LTH = 0xFFC0,
ADC0GTH:ADC0GTL = 0x2000.

An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0xFFC0 or > 0x2000. (Two's Complement math.)

Table 6.1. Comparator Electrical Characteristics

VDD = 3.0V, AV+ = 3.0V, -40°C to +85°C unless otherwise specified.

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Response Time1	(CP+) – (CP-) = 100mV (Note 1)		4		µs
Response Time2	(CP+) – (CP-) = 10mV (Note 1)		12		µs
Common Mode Rejection Ratio			1.5	4	mV/V
Positive Hysteresis1	CPnHYP1-0 = 00		0	1	mV
Positive Hysteresis2	CPnHYP1-0 = 01	2	4.5	7	mV
Positive Hysteresis3	CPnHYP1-0 = 10	4	9	13	mV
Positive Hysteresis4	CPnHYP1-0 = 11	10	17	25	mV
Negative Hysteresis1	CPnHYN1-0 = 00		0	1	mV
Negative Hysteresis2	CPnHYN1-0 = 01	2	4.5	7	mV
Negative Hysteresis3	CPnHYN1-0 = 10	4	9	13	mV
Negative Hysteresis4	CPnHYN1-0 = 11	10	17	25	mV
Inverting or Non-inverting Input Voltage Range		-0.25		(AV+) + 0.25	V
Input Capacitance			7		pF
Input Bias Current		-5	0.001	+5	nA
Input Offset Voltage		-10		+10	mV
POWER SUPPLY					
Power-up Time	CPnEN from 0 to 1		20		µs
Power Supply Rejection			0.1	1	mV/V
Supply Current	Operating Mode (each comparator) at DC		1.5	10	µA

Note 1: CPnHYP1-0 = CPnHYN1-0 = 00.

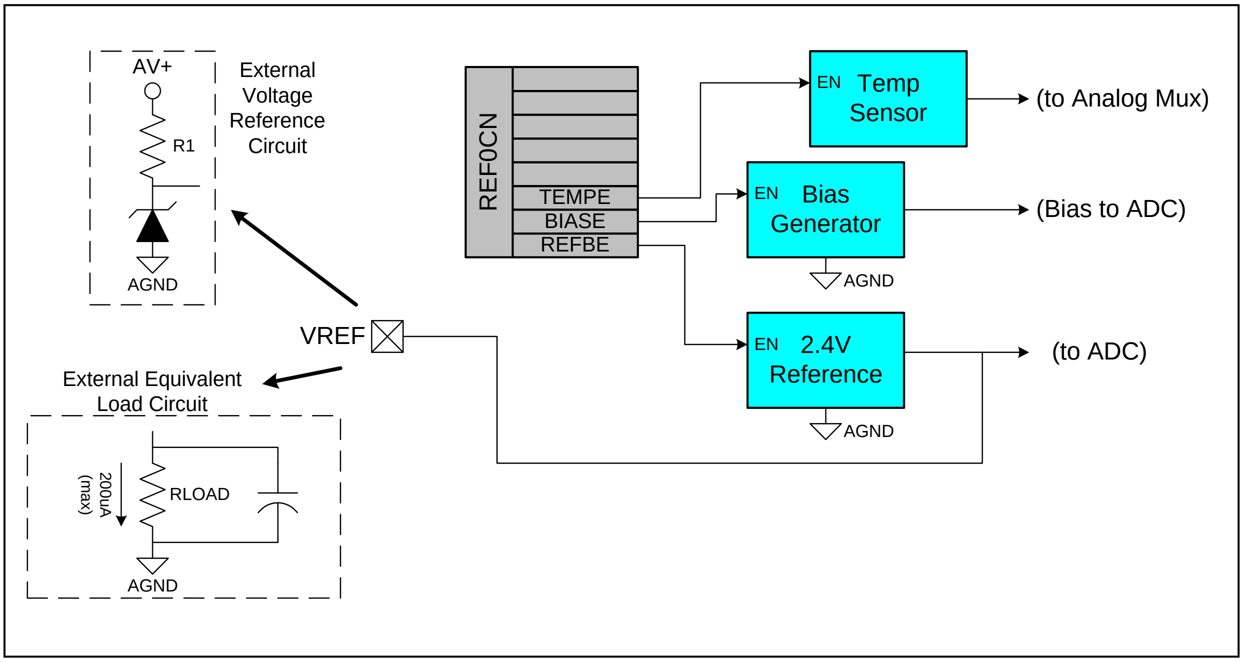
7. VOLTAGE REFERENCE

The voltage reference circuit consists of a 1.2V, 15ppm/°C (typical) bandgap voltage reference generator and a gain-of-two output buffer amplifier. The reference voltage on VREF can be connected to external devices in the system, as long as the maximum load seen by the VREF pin is less than 200µA to AGND (see Figure 7.1).

If a different reference voltage is required, an external reference can be connected to the VREF pin and the internal bandgap and buffer amplifier disabled in software. The external reference voltage must still be less than AV+ - 0.3V. The Reference Control Register, REF0CN (defined in Figure 7.2), provides the means to enable or disable the bandgap and buffer amplifier. The BIASE bit in REF0CN enables the bias circuitry for the ADC while the REFBE bit enables the bandgap reference and buffer amplifier which drive the VREF pin. When disabled, the supply current drawn by the bandgap and buffer amplifier falls to less than 1µA (typical) and the output of the buffer amplifier enters a high impedance state. If the internal bandgap is used as the reference voltage generator, BIASE and REFBE must both be set to 1. If an external reference is used, REFBE must be set to 0 and BIASE must be set to 1. If the ADC is not being used, both of these bits can be set to 0 to conserve power. The electrical specifications for the Voltage Reference are given in Table 7.1.

The temperature sensor connects to the highest order input of the A/D converter's input multiplexer. The TEMPE bit within REF0CN enables and disables the temperature sensor. While disabled, the temperature sensor defaults to a high impedance state and any A/D measurements performed on the sensor while disabled result in meaningless data.

Figure 7.1. Voltage Reference Functional Block Diagram



8.3. 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 CIP-51's resources and peripherals. The CIP-51 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 MCU. This allows the addition of new functionality while retaining compatibility with the MCS-51™ instruction set. Table 8.3 lists the SFRs implemented in the CIP-51 System Controller.

The SFR registers are accessed any time 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, P1, SCON, 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 datasheet, as indicated in Table 8.3, for a detailed description of each register.

Table 8.2. Special Function Register Memory Map

F8	SPI0CN	PCA0H	PCA0CPH0	PCA0CPH1	PCA0CPH2	PCA0CPH3	PCA0CPH4	WDTCN
F0	B						EIP1	EIP2
E8	ADC0CN	PCA0L	PCA0CPL0	PCA0CPL1	PCA0CPL2	PCA0CPL3	PCA0CPL4	RSTSRC
E0	ACC	XBR0	XBR1	XBR2			EIE1	EIE2
D8	PCA0CN	PCA0MD	PCA0CPM0	PCA0CPM1	PCA0CPM2	PCA0CPM3	PCA0CPM4	
D0	PSW	REF0CN						
C8	T2CON		RCAP2L	RCAP2H	TL2	TH2		SMB0CR
C0	SMB0CN	SMB0STA	SMB0DAT	SMB0ADR	ADC0GTL	ADC0GTH	ADC0LTL	ADC0LTH
B8	IP		AMX0CF	AMX0SL	ADC0CF		ADC0L	ADC0H
B0	P3	OSCXCN	OSCICN				FLSCL	FLACL
A8	IE					PRT1IF		EMI0CN
A0	P2				PRT0CF	PRT1CF	PRT2CF	PRT3CF
98	SCON	SBUF	SPI0CFG	SPI0DAT		SPI0CKR	CPT0CN	CPT1CN
90	P1	TMR3CN	TMR3RLL	TMR3RLH	TMR3L	TMR3H		
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON	PSCTL
80	P0	SP	DPL	DPH				PCON
	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)

Bit Addressable

Table 8.3. Special Function Registers

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

Address	Register	Description	Page No.
0xE0	ACC	Accumulator	58
0xBC	ADC0CF	ADC Configuration	28
0xE8	ADC0CN	ADC Control	31
0xC5	ADC0GTH	ADC Greater-Than Data Word (High Byte)	33
0xC4	ADC0GTL	ADC Greater-Than Data Word (Low Byte)	33
0xBF	ADC0H	ADC Data Word (High Byte)	32
0xBE	ADC0L	ADC Data Word (Low Byte)	32
0xC7	ADC0LTH	ADC Less-Than Data Word (High Byte)	33
0xC6	ADC0LTL	ADC Less-Than Data Word (Low Byte)	33

8.3.1. 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.

Figure 8.3. SP: Stack Pointer

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
								0x81

Bits 7-0: SP: 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.

Figure 8.4. 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

Bits 7-0: DPL: Data Pointer Low.
The DPL register is the low byte of the 16-bit DPTR. DPTR is used to access indirectly addressed RAM and Flash Memory.

Figure 8.5. DPH: Data Pointer 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:
								0x83

Bits 7-0: DPH: Data Pointer High.
The DPH register is the high byte of the 16-bit DPTR. DPTR is used to access indirectly addressed RAM and Flash Memory.

Figure 8.10. IP: Interrupt Priority

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	-	PT2	PS	PT1	PX1	PT0	PX0	00000000
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 interrupts.
0: Timer 2 interrupts set to low priority level.
1: Timer 2 interrupts set to high priority level.

Bit4: PS: Serial Port (UART) Interrupt Priority Control.
This bit sets the priority of the Serial Port (UART) interrupts.
0: UART interrupts set to low priority level.
1: UART interrupts set to high priority level.

Bit3: PT1: Timer 1 Interrupt Priority Control.
This bit sets the priority of the Timer 1 interrupts.
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 interrupts.
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 interrupts.
0: Timer 0 interrupt set to low priority level.
1: Timer 0 interrupt set to high priority level.

Bit0: PX0: External Interrupt 0 Priority Control.
This bit sets the priority of the External Interrupt 0 interrupts.
0: External Interrupt 0 set to low priority level.
1: External Interrupt 0 set to high priority level.

Figure 9.4. FLSCL: Flash Memory Timing Prescaler

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
FOSE	FRAE	-	-	FLASCL				10001111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xB6

Bit7: FOSE: Flash One-Shot Timer Enable
0: Flash One-shot timer disabled.
1: Flash One-shot timer enabled

Bit6: FRAE: Flash Read Always Enable
0: Flash reads per one-shot timer
1: Flash always in read mode

Bits5-4: UNUSED. Read = 00b, Write = don’t care.

Bits3-0: FLASCL: Flash Memory Timing Prescaler.
This register specifies the prescaler value for a given system clock required to generate the correct timing for Flash write/erase operations. If the prescaler is set to 1111b, Flash write/erase operations are disabled.
0000: System Clock < 50kHz
0001: 50kHz ≤ System Clock < 100kHz
0010: 100kHz ≤ System Clock < 200kHz
0011: 200kHz ≤ System Clock < 400kHz
0100: 400kHz ≤ System Clock < 800kHz
0101: 800kHz ≤ System Clock < 1.6MHz
0110: 1.6MHz ≤ System Clock < 3.2MHz
0111: 3.2MHz ≤ System Clock < 6.4MHz
1000: 6.4MHz ≤ System Clock < 12.8MHz
1001: 12.8MHz ≤ System Clock < 25.6MHz
1010: 25.6MHz ≤ System Clock < 51.2MHz *
1011, 1100, 1101, 1110: Reserved Values
1111: Flash Memory Write/Erase Disabled

The prescaler value is the smallest value satisfying the following equation:
$\text{FLASCL} > \log_2(\text{System Clock} / 50\text{kHz})$

* For test purposes. The C8051F018/9 is not guaranteed for operation over 25MHz.

Figure 11.4. RSTSRC: Reset Source Register

R	R/W	R/W	R/W	R	R	R/W	R	Reset Value
JTAGRST	CNVRSEF	C0RSEF	SWRSEF	WDTRSF	MCDRSF	PORSF	PINRSF	xxxxxxxx
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xEF

(Note: Do not use read-modify-write operations on this register.)

Bit7: JTAGRST. JTAG Reset Flag.
0: JTAG is not currently in reset state.
1: JTAG is in reset state.

Bit6: CNVRSEF: Convert Start Reset Source Enable and Flag
Write
0: CNVSTR is not a reset source
1: CNVSTR is a reset source (active low)
Read
0: Source of prior reset was not from CNVSTR
1: Source of prior reset was from CNVSTR

Bit5: C0RSEF: Comparator 0 Reset Enable and Flag
Write
0: Comparator 0 is not a reset source
1: Comparator 0 is a reset source (active low)
Read
Note: The value read from C0RSEF is not defined if Comparator 0 has not been enabled as a reset source.
0: Source of prior reset was not from Comparator 0
1: Source of prior reset was from Comparator 0

Bit4: SWRSF: Software Reset Force and Flag
Write
0: No Effect
1: Forces an internal reset. /RST pin is not affected.
Read
0: Prior reset source was not from write to the SWRSF bit.
1: Prior reset source was from write to the SWRSF bit.

Bit3: WDTRSF: Watchdog Timer Reset Flag
0: Source of prior reset was not from WDT timeout.
1: Source of prior reset was from WDT timeout.

Bit2: MCDRSF: Missing Clock Detector Flag
0: Source of prior reset was not from Missing Clock Detector timeout.
1: Source of prior reset was from Missing Clock Detector timeout.

Bit1: PORSF: Power-On Reset Force and Flag
Write
0: No effect
1: Forces a Power-On Reset. /RST is driven low.
Read
0: Source of prior reset was not from POR.
1: Source of prior reset was from POR.

Bit0: PINRSF: HW Pin Reset Flag
0: Source of prior reset was not from /RST pin.
1: Source of prior reset was from /RST pin.

Figure 13.13. P3: Port3 Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0	11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							(bit addressable)	0xB0

Bits7-0: P3.[7:0]
(Write)
0: Logic Low Output.
1: Logic High Output (high-impedance if corresponding PRT3CF.n bit = 0)
(Read)
0: P3.n is logic low.
1: P3.n is logic high.

Figure 13.14. PRT3CF: Port3 Configuration Register

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:
								0xA7

Bits7-0: PRT3CF.[7:0]: Output Configuration Bits for P3.7-P3.0 (respectively)
0: Corresponding P3.n Output mode is Open-Drain.
1: Corresponding P3.n Output mode is Push-Pull.

Table 13.2. Port I/O DC Electrical Characteristics

VDD = 2.8 to 3.6V, -40°C to +85°C unless otherwise specified.

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Output High Voltage	I_{OH} = -10uA, Port I/O push-pull I_{OH} = -3mA, Port I/O push-pull I_{OH} = -10mA, Port I/O push-pull	VDD – 0.1 VDD – 0.7	VDD – 0.8		V
Output Low Voltage	I_{OL} = 10uA I_{OL} = 8.5mA I_{OL} = 25mA		1.0	0.1 0.6	V
Input High Voltage		0.7 x VDD			V
Input Low Voltage				0.3 x VDD	V
Input Leakage Current	DGND < Port Pin < VDD, Pin Tri-state Weak Pull-up Off Weak Pull-up On		30	±1	µA
Capacitive Loading			5		pF

14.2.4. Slave Receiver Mode

Serial data is received on SDA while the serial clock is received on SCL. First, a byte is received that contains an address and data direction bit. In this case the data direction bit (R/W) will be logic 0 to indicate a “WRITE” operation. If the received address matches the slave’s assigned address (or a general call address is received) one or more bytes of serial data are received from the master. After each byte is received, an acknowledge bit is transmitted by the slave. The master outputs START and STOP conditions to indicate the beginning and end of the serial transfer.

14.3. Arbitration

A master may start a transfer only if the bus is free. The bus is free after a STOP condition or after the SCL and SDA lines remains high for a specified time. Two or more master devices may attempt to generate a START condition at the same time. Since the devices that generated the START condition may not be aware that other masters are contending for the bus, an arbitration scheme is employed. The master devices continue to transmit until one of the masters transmits a HIGH level, while the other(s) master transmits a LOW level on SDA. The first master(s) transmitting the HIGH level on SDA looses the arbitration and is required to give up the bus.

14.4. 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 can hold the SCL line LOW to extend the clock low period, effectively decreasing the serial clock frequency.

14.5. Timeouts

14.5.1. 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 25ms as a “timeout” condition. Devices that have detected the timeout condition must reset the communication no later than 10ms after detecting the timeout condition.

One of the MCU’s general-purpose timers, operating in 16-bit auto-reload mode, can be used to monitor the SCL line for this timeout condition. Timer 3 is specifically designed for this purpose. (Refer to the Timer 3 Section 17.3. for detailed information on Timer 3 operation.)

14.5.2. SCL High (SMBus Free) Timeout

The SMBus specification stipulates that if a device holds the SCL and SDA lines high for more that 50usec, the bus is designated as free. The SMB0CR register is used to detect this condition when the FTE bit in SMB0CN is set.

14.6. SMBus Special Function Registers

The SMBus serial interface is accessed and controlled through five SFRs: SMB0CN Control Register, SMB0CR Clock Rate Register, SMB0ADR Address Register, SMB0DAT Data Register and SMB0STA Status Register. The system device may have one or more SMBus serial interfaces implemented. The five special function registers related to the operation of the SMBus interface are described in the following section.

14.6.1. Control Register

The SMBus Control register SMB0CN is used to configure and control the SMBus interface. All of the bits in the register can be read or written by software. Two of the control bits are also affected by the SMBus hardware. The Serial Interrupt flag (SI, SMB0CN.3) is set to logic 1 by the hardware when a valid serial interrupt condition occurs. It can only be cleared by software. The Stop flag (STO, SMB0CN.4) is cleared to logic 0 by hardware when a STOP condition is present on the bus.

Setting the ENSMB flag to logic 1 enables the SMBus interface. Clearing the ENSMB flag to logic 0 disables the SMBus interface and removes it from the bus. Momentarily clearing the ENSMB flag and then resetting it to logic 1 will reset a SMBus communication. However, ENSMB should not be used to temporarily remove a device from the bus since the bus state information will be lost. Instead, the Assert Acknowledge (AA) flag should be used to temporarily remove the device from the bus (see description of AA flag below).

Setting the Start flag (STA, SMB0CN.5) to logic 1 will put the SMBus in a master mode. If the bus is free, the SMBus hardware will generate a START condition. If the bus is not free, the SMBus hardware waits for a STOP condition to free the bus and then generates a START condition after a 5μs delay per the SMB0CR value. (In accordance with the SMBus protocol, the SMBus interface also considers the bus free if the bus is idle for 50μs and no STOP condition was recognized.) If STA is set to logic 1 while the SMBus is in master mode and one or more bytes have been transferred, a repeated START condition will be generated. To ensure proper operation, the STO flag should be explicitly cleared before setting STA to a logic 1.

When the Stop flag (STO, SMB0CN.4) is set to logic 1 while the SMBus interface is in master mode, the hardware generates a STOP condition on the SMBus. In a slave mode, the STO flag may be used to recover from an error condition. In this case, a STOP condition is not generated on the SMBus, but the SMBus hardware behaves as if a STOP condition has been received and enters the “not addressed” slave receiver mode. The SMBus hardware automatically clears the STO flag to logic 0 when a STOP condition is detected on the bus.

The Serial Interrupt flag (SI, SMB0CN.3) is set to logic 1 by hardware when the SMBus interface enters one of 27 possible states. If interrupts are enabled for the SMBus interface, an interrupt request is generated when the SI flag is set. The SI flag must be cleared by software. While SI is set to logic 1, the clock-low period of the serial clock will be stretched and the serial transfer is suspended.

The Assert Acknowledge flag (AA, SMB0CN.2) is used to set the level of the SDA line during the acknowledge clock cycle on the SCL line. Setting the AA flag to logic 1 will cause an ACKNOWLEDGE (low level on SDA) to be sent during the acknowledge cycle if the device has been addressed. Setting the AA flag to logic 0 will cause a NOT ACKNOWLEDGE (high level on SDA) to be sent during acknowledge cycle. After the transmission of a byte in slave mode, the slave can be temporarily removed from the bus by clearing the AA flag. The slave’s own address and general call address will be ignored. To resume operation on the bus, the AA flag must be reset to logic 1 to allow the slave’s address to be recognized.

Setting the SMBus Free Timer Enable bit (FTE, SMB0CN.1) to logic 1 enables the SMBus Free Timeout feature. If SCL and SDA remain high for the SMBus Free Timeout given in the SMBus Clock Rate Register (Figure 14.5), the bus will be considered free and a Start will be generated if pending. The bus free period should be greater than 50μs.

Setting the SMBus timeout enable bit (TOE, SMB0CN.0) to logic 1 enables Timer 3 to count up when the SCL line is low and Timer 3 is enabled. If Timer 3 overflows, a Timer 3 interrupt will be generated, which will alert the CPU that a SMBus SCL low timeout has occurred.

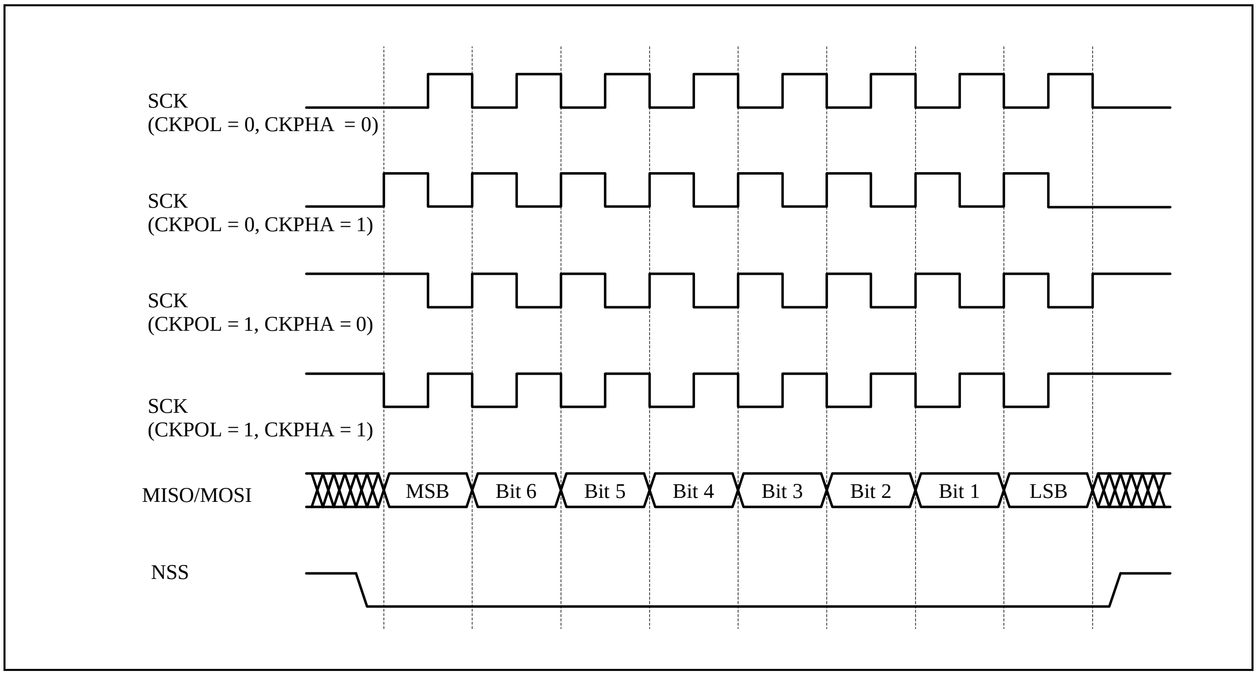
in an “off-line” state. In a multiple-master environment, the system controller should check the state of the SLVSEL flag (SPI0CN.2) to ensure the bus is free before setting the MSTEN bit and initiating a data transfer.

15.3. Serial Clock Timing

As shown in Figure 15.4, four combinations of serial clock phase and polarity can be selected using the clock control bits in the SPI Configuration Register (SPI0CFG). The CKPHA bit (SPI0CFG.7) selects one of two clock phases (edge used to latch the data). The CKPOL bit (SPI0CFG.6) selects between an active-high or active-low clock. Both master and slave devices must be configured to use the same clock phase and polarity. Note: the SPI should be disabled (by clearing the SPIEN bit, SPI0CN.0) while changing the clock phase and polarity.

The SPI Clock Rate Register (SPI0CKR) as shown in Figure 15.7 controls the master mode serial clock frequency. This register is ignored when operating in slave mode.

Figure 15.4. Data/Clock Timing Diagram



16.1.3. Mode 2: 9-Bit UART, Fixed Baud Rate

Mode 2 provides asynchronous, full-duplex communication using a total of eleven bits per data byte: a start bit, 8 data bits (LSB first), a programmable ninth data bit, and a stop bit (see timing diagram in Figure 16.6). On transmit, the ninth data bit is determined by the value in TB8 (SCON.3). It can be assigned the value of the parity flag P in the PSW or used in multiprocessor communications. On receive, the ninth data bit goes into RB8 (SCON.2) and the stop bit is ignored.

Data transmission begins when an instruction writes a data byte to the SBUF register. The TI Transmit Interrupt Flag (SCON.1) is set at the end of the transmission (the beginning of the stop-bit time). Data reception can begin any time after the REN Receive Enable bit (SCON.4) is set to logic 1. After the stop bit is received, the data byte will be loaded into the SBUF receive register if the following conditions are met: RI must be logic 0, and if SM2 is logic 1, the 9th bit must be logic 1.

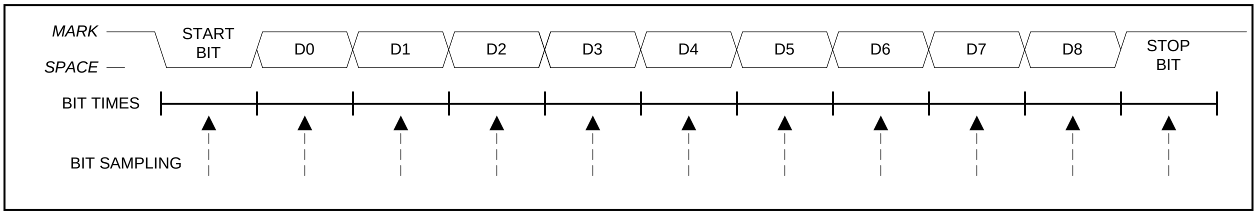
If these conditions are met, the eight bits of data are stored in SBUF, the ninth bit is stored in RB8 and the RI flag is set. If these conditions are not met, SBUF and RB8 will not be loaded and the RI flag will not be set. An interrupt will occur if enabled when either TI or RI are set.

The baud rate in Mode 2 is a direct function of the system clock frequency as follows:

$$\textit{Mode 2 Baud Rate} = 2^{SMOD} * (SYSCLK / 64).$$

The SMOD bit (PCON.7) selects whether to divide SYSCLK by 32 or 64. In the formula, 2 is raised to the power SMOD, resulting in a baud rate of either 1/32 or 1/64 of the system clock frequency. On reset, the SMOD bit is logic 0, thus selecting the lower speed baud rate by default.

Figure 16.6. UART Modes 2 and 3 Timing Diagram



16.1.4. Mode 3: 9-Bit UART, Variable Baud Rate

Mode 3 is the same as Mode 2 in all respects except the baud rate is variable. The baud rate is determined in the same manner as for Mode 1. Mode 3 operation transmits 11 bits: a start bit, 8 data bits (LSB first), a programmable ninth data bit, and a stop bit. Timer 1 or Timer 2 overflows generate the baud rate just as with Mode 1. In summary, Mode 3 transmits using the same protocol as Mode 2 but with Mode 1 baud rate generation.

Figure 17.4. TCON: Timer Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0 (bit addressable)	SFR Address: 0x88

Bit7: TF1: Timer 1 Overflow Flag.
Set 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.
0: No Timer 1 overflow detected.
1: Timer 1 has overflowed.

Bit6: TR1: Timer 1 Run Control.
0: Timer 1 disabled.
1: Timer 1 enabled.

Bit5: TF0: Timer 0 Overflow Flag.
Set 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.
0: No Timer 0 overflow detected.
1: Timer 0 has overflowed.

Bit4: TR0: Timer 0 Run Control.
0: Timer 0 disabled.
1: Timer 0 enabled.

Bit3: 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 if IT1 = 1. This flag is the inverse of the /INT1 input signal's logic level when IT1 = 0.

Bit2: IT1: Interrupt 1 Type Select.
This bit selects whether the configured /INT1 signal will detect falling edge or active-low level-sensitive interrupts.
0: /INT1 is level triggered.
1: /INT1 is edge triggered.

Bit1: IE0: External Interrupt 0.
This flag is set by hardware when an edge/level of type defined by IT0 is detected. It can be cleared by software but is automatically cleared when the CPU vectors to the External Interrupt 0 service routine if IT0 = 1. This flag is the inverse of the /INT0 input signal's logic level when IT0 = 0.

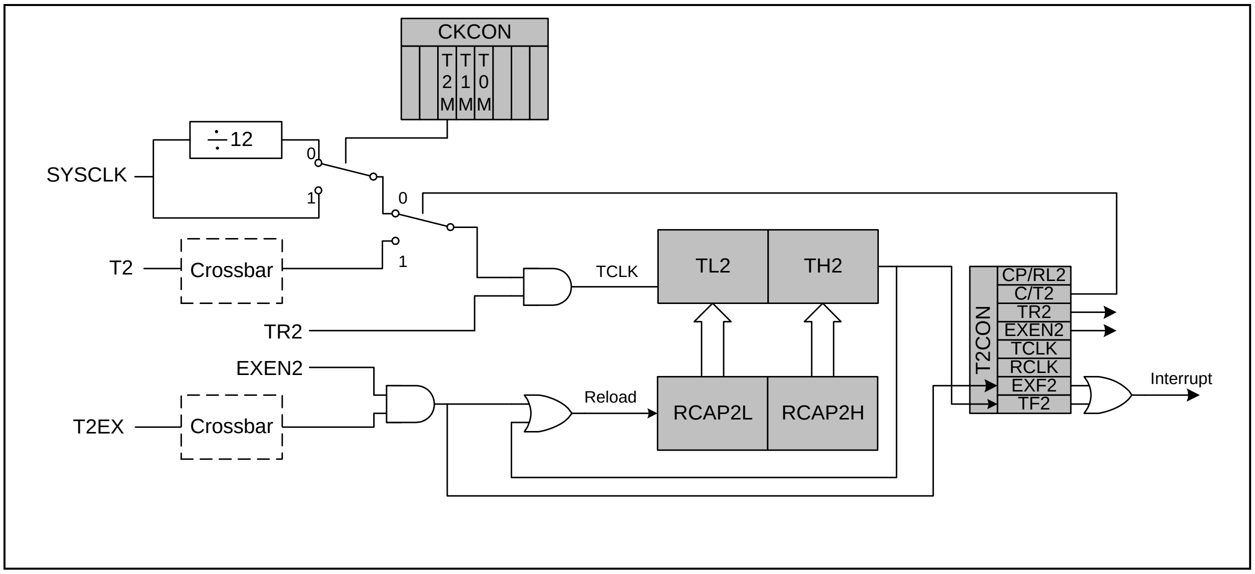
Bit0: IT0: Interrupt 0 Type Select.
This bit selects whether the configured /INT0 signal will detect falling edge or active-low level-sensitive interrupts.
0: /INT0 is level triggered.
1: /INT0 is edge triggered.

17.2.2. Mode 1: 16-bit Counter/Timer with Auto-Reload

The Counter/Timer with Auto-Reload mode sets the TF2 timer overflow flag when the counter/timer register overflows from 0xFFFF to 0x0000. An interrupt is generated if enabled. On overflow, the 16-bit value held in the two capture registers (RCAP2H, RCAP2L) is automatically loaded into the counter/timer register and the timer is restarted.

Counter/Timer with Auto-Reload mode is selected by clearing the CP/RL2 bit. Setting TR2 to logic 1 enables and starts the timer. Timer 2 can use either the system clock or transitions on an external input pin as its clock source, as specified by the C/T2 bit. If EXEN2 is set to logic 1, a high-to-low transition on T2EX will also cause Timer 2 to be reloaded. If EXEN2 is cleared, transitions on T2EX will be ignored.

Figure 17.12. T2 Mode 1 Block Diagram



17.3. Timer 3

Timer 3 is a 16-bit timer formed by the two 8-bit SFRs, TMR3L (low byte) and TMR3H (high byte). The input for Timer 3 is the system clock (divided by either one or twelve as specified by the Timer 3 Clock Select bit T3M in the Timer 3 Control Register TMR3CN). Timer 3 is always configured as an auto-reload timer, with the reload value held in the TMR3RLL (low byte) and TMR3RLH (high byte) registers. Timer 3 can be used to start an ADC Data Conversion, for SMBus timing (see Section 14.5), or as a general-purpose timer. Timer 3 does not have a counter mode.

Figure 17.19. Timer 3 Block Diagram

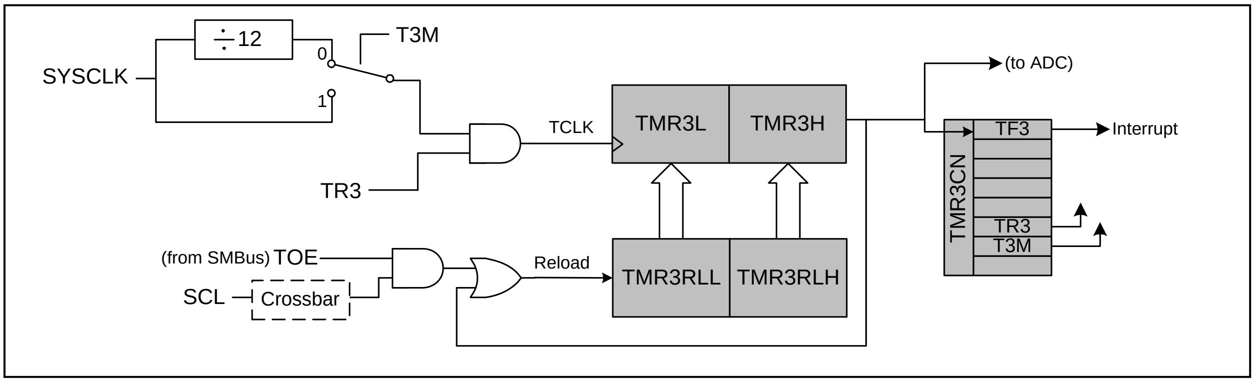


Figure 17.20. TMR3CN: Timer 3 Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
TF3	-	-	-	-	TR3	T3M	-	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x91

Bit7: TF3: Timer3 Overflow Flag.
Set by hardware 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 and must be cleared by software.

Bits6-3: UNUSED. Read = 0000b, Write = don’t care.

Bit2: TR3: Timer 3 Run Control.
This bit enables/disables Timer 3.
0: Timer 3 disabled.
1: Timer 3 enabled.

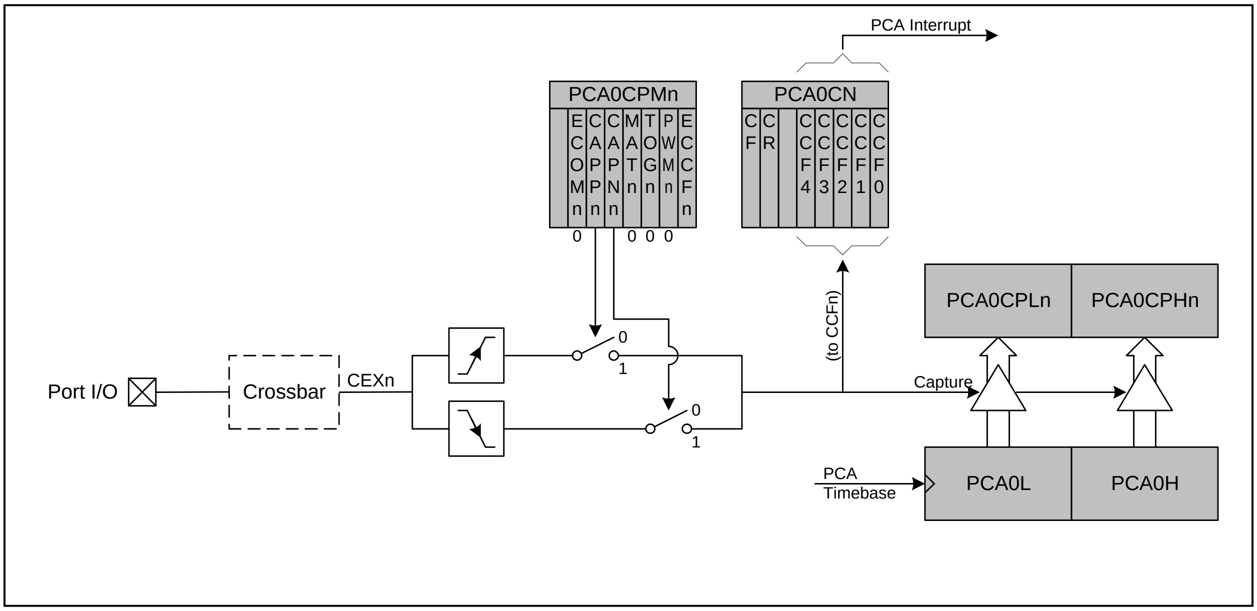
Bit1: T3M: Timer 3 Clock Select.
This bit controls the division of the system clock supplied to Counter/Timer 3.
0: Counter/Timer 3 uses the system clock divided by 12.
1: Counter/Timer 3 uses the system clock.

Bit0: UNUSED. Read = 0, Write = don’t care.

18.1.1. 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 load 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.

Figure 18.3. PCA Capture Mode Diagram



19.2. Flash Programming Commands

The Flash memory can be programmed directly over the JTAG interface using the Flash Control, Flash Data, Flash Address, and Flash Scale registers. These Indirect Data Registers are accessed via the JTAG Instruction Register. Read and write operations on indirect data registers are performed by first setting the appropriate DR address in the IR register. Each read or write is then initiated by writing the appropriate Indirect Operation Code (IndOpCode) to the selected data register. Incoming commands to this register have the following format:

19:18	17:0
IndOpCode	WriteData

IndOpCode: These bit set the operation to perform according to the following table:

IndOpCode	**Operation**
0x	Poll
10	Read
11	Write

The Poll operation is used to check the Busy bit as described below. Although a Capture-DR is performed, no Update-DR is allowed for the Poll operation. Since updates are disabled, polling can be accomplished by shifting in/out a single bit.

The Read operation initiates a read from the register addressed by the IR. Reads can be initiated by shifting only 2 bits into the indirect register. After the read operation is initiated, polling of the Busy bit must be performed to determine when the operation is complete.

The write operation initiates a write of WriteData to the register addressed by the IR. Registers of any width up to 18 bits can be written. If the register to be written contains fewer than 18 bits, the data in WriteData should be left-justified, i.e. its MSB should occupy bit 17 above. This allows shorter registers to be written in fewer JTAG clock cycles. For example, an 8-bit register could be written by shifting only 10 bits. After a Write is initiated, the Busy bit should be polled to determine when the next operation can be initiated. The contents of the Instruction Register should not be altered while either a read or write operation is in progress.

Outgoing data from the indirect Data Register has the following format:

19	18:1	0
0	ReadData	Busy

The Busy bit indicates that the current operation is not complete. It goes high when an operation is initiated and returns low when complete. Read and Write commands are ignored while Busy is high. In fact, if polling for Busy to be low will be followed by another read or write operation, JTAG writes of the next operation can be made while checking for Busy to be low. They will be ignored until Busy is read low, at which time the new operation will initiate. This bit is placed at bit 0 to allow polling by single-bit shifts. When waiting for a Read to complete and Busy is 0, the following 18 bits can be shifted out to obtain the resulting data. ReadData is always right-justified. This allows registers shorter than 18 bits to be read using a reduced number of shifts. For example, the result from a byte-read requires 9 bit shifts (Busy + 8 bits).

19.3. Debug Support

Each MCU has on-chip JTAG and debug circuitry that provide *non-intrusive, full speed, in-circuit debug using the production part installed in the end application* using the four pin JTAG I/F. Silicon Laboratories' debug system supports inspection and modification of memory and registers, setting breakpoints, watchpoints, single stepping, and run and halt commands. No additional target RAM, program memory, or communications channels are required. All the digital and analog peripherals are functional and work correctly (remain in sync) while debugging. The WDT is disabled when the MCU is halted during single stepping or at a breakpoint.

The C8051F015DK is a development kit with all the hardware and software necessary to develop application code and perform in-circuit debugging with the C8051F018/9. Each kit includes an Integrated Development Environment (IDE) which has a debugger and integrated 8051 assembler. It has an RS-232 to JTAG protocol translator module referred to as the EC. There is also a target application board with a C8051F015 installed and with a large prototyping area. The kit also includes RS-232 and JTAG cables, and wall-mount power supply.