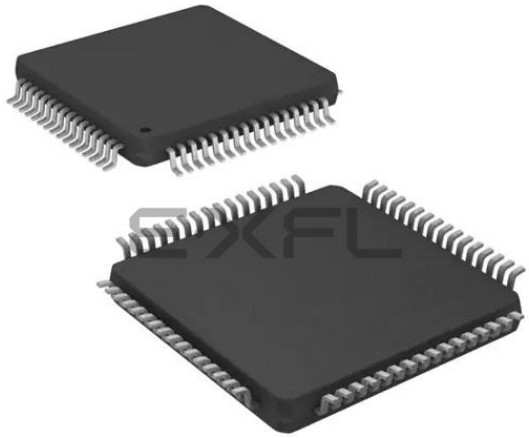


Welcome to E-XFL.COM

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, WDT
Number of I/O	24
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4.25K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 2x16b
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/c8051f065

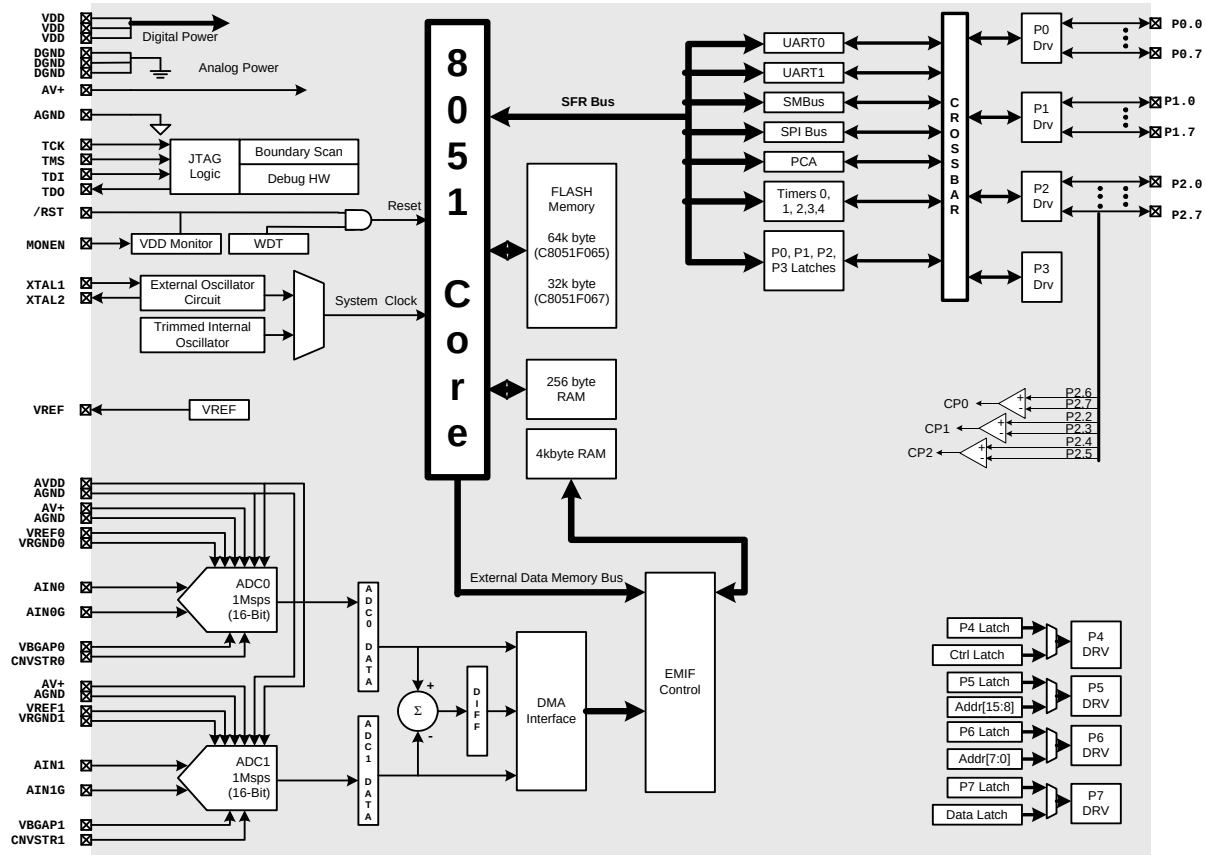


Figure 1.4. C8051F065 / C8051F067 Block Diagram

1.11. Analog Comparators

The C8051F060/1/2/3/4/5/6/7 MCUs include three analog comparators on-chip. The comparators have software programmable hysteresis and response time. Each comparator can generate an interrupt on its rising edge, falling edge, or both. The interrupts are capable of waking up the MCU from sleep mode, and Comparator 0 can be used as a reset source. The output state of the comparators can be polled in software or routed to Port I/O pins via the Crossbar. Outputs from the comparator can be routed through the crossbar. The comparators can be programmed to a low power shutdown mode when not in use.

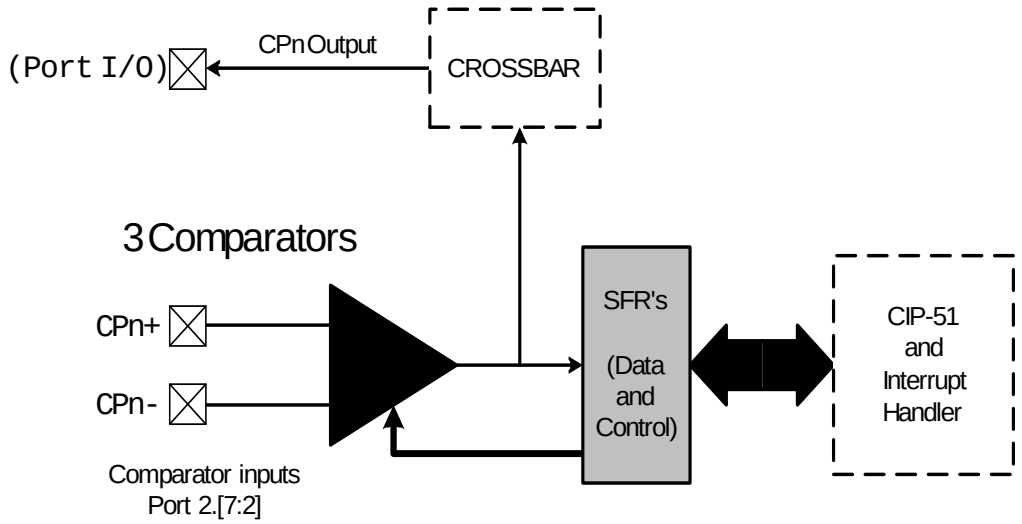


Figure 1.15. Comparator Block Diagram

3. Global DC Electrical Characteristics

Table 3.1. Global DC Electrical Characteristics

-40 to +85 °C, 25 MHz System Clock unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
Analog Supply Voltage (AV+, AVDD)	(Note 1)	2.7	3.0	3.6	V
Digital Supply Voltage (VDD)		2.7	3.0	3.6	V
Analog-to-Digital Supply Delta (|VDD - AV+| or |VDD - AVDD|)				0.5	V
Supply Current from Analog Peripherals (active)	Internal REF, ADC, DAC, Comparators all enabled. (Note 2)		14		mA
Supply Current from Analog Peripherals (inactive)	Internal REF, ADC, DAC, Comparators all disabled, oscillator disabled.		0.2		µA
Supply Current from CPU and Digital Peripherals (CPU active) (Note 3)	VDD=2.7 V, Clock=25 MHz VDD=2.7 V, Clock=1 MHz VDD=2.7 V, Clock=32 kHz VDD=3.0 V, Clock=25 MHz VDD=3.0 V, Clock=1 MHz VDD=3.0 V, Clock=32 kHz		18 0.7 30 20 1.0 35		mA mA µA mA mA µA
Supply Current from CPU and Digital Peripherals (CPU inactive, not accessing Flash) (Note 3)	VDD=2.7 V, Clock=25 MHz VDD=2.7 V, Clock=1 MHz VDD=2.7 V, Clock=32 kHz VDD=3.0 V, Clock=25 MHz VDD=3.0 V, Clock=1 MHz VDD=3.0 V, Clock=32 kHz		13 0.5 20 16 0.8 23		mA mA µA mA mA µA
Supply Current with all systems shut down	Oscillator not running		0.2		µA
VDD Supply RAM Data Retention Voltage			1.5		V
SYSCLK (System Clock)	(Note 4)	0		25	MHz
Specified Operating Temperature Range		-40		+85	°C

Note 1: Analog Supply AV+ must be greater than 1 V for VDD monitor to operate.
Note 2: Internal Oscillator and VDD Monitor current not included. Individual supply current contributions for each peripheral are listed in the chapter.
Note 3: Current increases linearly with supply Voltage.
Note 4: SYSCLK must be at least 32 kHz to enable debugging.

5.3. ADC Modes of Operation

ADC0 and ADC1 have a maximum conversion speed of 1 Msps. The conversion clocks for the ADCs are derived from the system clock. The ADCnSC bits in the ADCnCF register determine how many system clocks (from 1 to 16) are used for each conversion clock.

5.3.1. Starting a Conversion

For ADC0, conversions can be initiated in one of four ways, depending on the programmed states of the ADC0 Start of Conversion Mode bits (AD0CM1, AD0CM0) in ADC0CN. For ADC0, conversions may be initiated by:

1. Writing a '1' to the AD0BUSY bit of ADC0CN;
2. A Timer 3 overflow (i.e. timed continuous conversions);
3. A rising edge detected on the external ADC convert start signal, CNVSTR0;
4. A Timer 2 overflow (i.e. timed continuous conversions).

ADC1 conversions can be initiated in five different ways, according to the ADC1 Start of Conversion Mode bits (AD1CM2-AD1CM0) in ADC1CN. For ADC1, conversions may be initiated by:

1. Writing a '1' to the AD1BUSY bit of ADC1CN;
2. A Timer 3 overflow (i.e. timed continuous conversions);
3. A rising edge detected on the external ADC convert start signal, CNVSTR1;
4. A Timer 2 overflow (i.e. timed continuous conversions);
5. Writing a '1' to the AD0BUSY bit of ADC0CN.

The ADnBUSY bit is set to logic 1 during conversion and restored to logic 0 when conversion is complete. The falling edge of ADnBUSY triggers an interrupt (when enabled) and sets the ADnINT interrupt flag (ADCnCN.5). In single-ended mode, the converted data for ADCn is available in the ADCn data word MSB and LSB registers, ADCnH, ADCnL. In differential mode, the converted data (combined from ADC0 and ADC1) is available in the ADC0 data word MSB and LSB registers, ADC0H, ADC0L.

When initiating conversions by writing a '1' to ADnBUSY, the ADnINT bit should be polled to determine when a conversion has completed (ADCn interrupts may also be used). The recommended polling procedure is shown below.

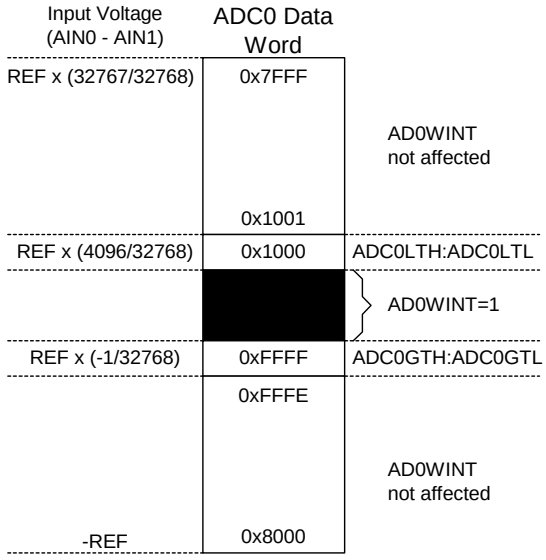
Step 1. Write a '0' to ADnINT;
Step 2. Write a '1' to ADnBUSY;
Step 3. Poll ADnINT for '1';
Step 4. Process ADCn data.

When an external start-of-conversion source is required in differential mode the two pins (CNVSTR0 and CNVSTR1) should be tied together.

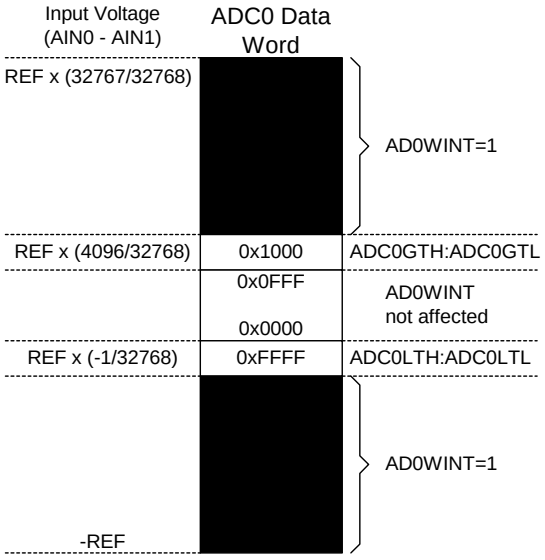
5.3.2. Tracking Modes

The ADnTM bit in register ADCnCN controls the ADCn track-and-hold mode. When the ADC is enabled, the ADC input is continuously tracked when a conversion is not in progress. When the ADnTM bit is logic 1, each conversion is preceded by a tracking period (after the start-of-conversion signal). When the CNVSTRn signal is used to initiate conversions, the ADC will track until a rising edge occurs on the CNVSTRn pin (see Figure 5.4 and Table 5.1 for conversion timing parameters). Setting ADnTM to 1 can be useful to ensure that settling time requirements are met when an external multiplexer is used on the analog input (see Section "5.3.3. Settling Time Requirements" on page 56).

Figure 5.29. 16-Bit ADC0 Window Interrupt Example: Differential Data



Given:
AMX0SL = 0x40,
ADC0LTH:ADC0LTL = 0x1000,
ADC0GTH:ADC0GTL = 0xFFFF.
An ADC0 End of Conversion will cause an ADC0 Window Compare Interrupt (AD0WINT = '1') if the resulting ADC0 Data Word is < 0x1000 and > 0xFFFF. (In two's-complement math, 0xFFFF = -1.)



Given:
AMX0SL = 0x40,
ADC0LTH:ADC0LTL = 0xFFFF,
ADC0GTH:ADC0GTL = 0x1000.
An ADC0 End of Conversion will cause an ADC0 Window Compare Interrupt (AD0WINT = '1') if the resulting ADC0 Data Word is < 0xFFFF or > 0x1000. (In two's-complement math, 0xFFFF = -1.)

7. 10-Bit ADC (ADC2, C8051F060/1/2/3)

The ADC2 subsystem for the C8051F060/1/2/3 consists of an analog multiplexer (referred to as AMUX2), and a 200 ksps, 10-bit successive-approximation-register ADC with integrated track-and-hold and programmable window detector (see block diagram in Figure 7.1). The AMUX2, data conversion modes, and window detector can all be configured from within software via the Special Function Registers shown in Figure 7.1. ADC2 operates in both Single-ended and Differential modes, and may be configured to measure any of the pins on Port 1, or the Temperature Sensor output. The ADC2 subsystem is enabled only when the AD2EN bit in the ADC2 Control register (ADC2CN) is set to logic 1. The ADC2 subsystem is in low power shutdown when this bit is logic 0.

Figure 7.1. ADC2 Functional Block Diagram

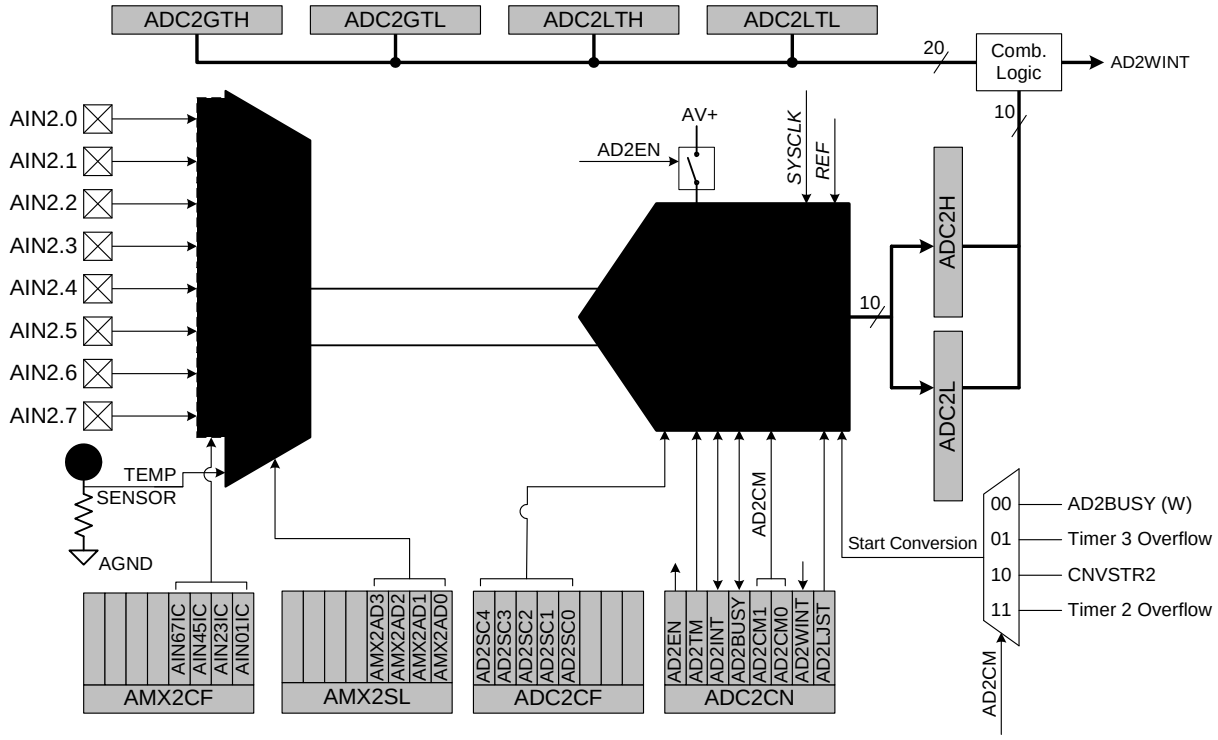
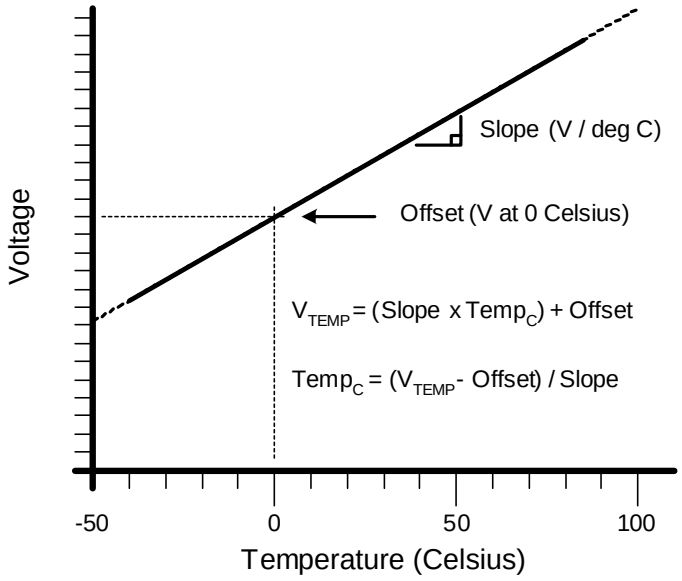


Figure 7.2. Temperature Sensor Transfer Function



7.2. Modes of Operation

ADC2 has a maximum conversion speed of 200 ksps. The ADC2 conversion clock is a divided version of the system clock, determined by the AD2SC bits in the ADC2CF register (system clock divided by (AD2SC + 1) for 0 ≤ AD2SC ≤ 31). The ADC2 conversion clock should be no more than 3 MHz.

7.2.1. Starting a Conversion

A conversion can be initiated in one of four ways, depending on the programmed states of the ADC2 Start of Conversion Mode bits (AD2CM1-0) in register ADC2CN. Conversions may be initiated by one of the following:

1. Writing a ‘1’ to the AD2BUSY bit of register ADC2CN
2. A Timer 3 overflow (i.e. timed continuous conversions)
3. A rising edge on the CNVSTR2 input signal (Assigned by the crossbar)
4. A Timer 2 overflow

When CNVSTR2 is used as a conversion start source, it must be enabled in the crossbar, and the corresponding pin must be set to open-drain, high-impedance mode (see Section “18. Port Input/Output” on page 203 for more details on Port I/O configuration).

Writing a ‘1’ to AD2BUSY provides software control of ADC2 whereby conversions are performed "on-demand". During conversion, the AD2BUSY bit is set to logic 1 and reset to logic 0 when the conversion is complete. The falling edge of AD2BUSY triggers an interrupt (when enabled) and sets the ADC2 interrupt flag (AD2INT). Note: When polling for ADC conversion completions, the ADC2 interrupt flag (AD2INT) should be used. Converted data is available in the ADC2 data registers, ADC2H and ADC2L, when bit AD2INT is logic 1. Note that when Timer 2 or Timer 3 overflows are used as the conversion source, low byte overflows are used if the timer is in 8-bit mode; and high byte overflows are used if the timer is in 16-bit mode. See Section “24. Timers” on page 287 for timer configuration.

Figure 7.8. ADC2H: ADC2 Data Word MSB Register

SFR Page: 2
SFR Address: 0xBF

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	

Bits7-0: ADC2 Data Word High-Order Bits.
For AD2LJST = 0: Bits 7-2 are the sign extension of Bit 1. Bits 1-0 are the upper 2 bits of the 10-bit ADC2 Data Word.
For AD2LJST = 1: Bits 7-0 are the most-significant bits of the 10-bit ADC2 Data Word.

Figure 7.9. ADC2L: ADC2 Data Word LSB Register

SFR Page: 2
SFR Address: 0xBE

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	

Bits7-0: ADC2 Data Word Low-Order Bits.
For AD2LJST = 0: Bits 7-0 are the lower 8 bits of the 10-bit Data Word.
For AD2LJST = 1: Bits 7-6 are the lower 2 bits of the 10-bit Data Word. Bits 5-0 will always read ‘0’.

.

Table 8.1. DAC Electrical Characteristics

VDD = 3.0 V, AV+ = 3.0 V, VREF = 2.40 V (REFBE = 0), No Output Load unless otherwise specified

Parameter	Conditions	Min	Typ	Max	Units
Static Performance					
Resolution			12		bits
Integral Nonlinearity			±1.5		LSB
Differential Nonlinearity				±1	LSB
Output Noise	No Output Filter 100 kHz Output Filter 10 kHz Output Filter		250 128 41		µVrms
Offset Error	Data Word = 0x014		±3	±30	mV
Offset Tempco			6		ppm/°C
Full-Scale Error			±20	±60	mV
Full-Scale Error Tempco			10		ppm/°C
VDD Power Supply Rejection Ratio			-60		dB
Output Impedance in Shutdown Mode	DACnEN = 0		100		kΩ
Output Sink Current			300		µA
Output Short-Circuit Current	Data Word = 0xFFF		15		mA
Dynamic Performance					
Voltage Output Slew Rate	Load = 40pF		0.44		V/µs
Output Settling Time to 1/2 LSB	Load = 40pF, Output swing from code 0xFFF to 0x014		10		µs
Output Voltage Swing		0		VREF-1LSB	V
Startup Time			10		µs
Analog Outputs					
Load Regulation	I_L = 0.01mA to 0.3mA at code 0xFFF		60		ppm
Power Consumption (each DAC)					
Power Supply Current (AV+ supplied to DAC)	Data Word = 0x7FF		300	500	µA

Figure 13.9. SFRPGCN: SFR Page Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	-	-	-	-	-	-	SFRPGEN	00000001
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0x96
SFR Page: F

Bits7-1: Reserved.

Bit0: SFRPGEN: SFR Automatic Page Control Enable.
Upon interrupt, the C8051 Core will vector to the specified interrupt service routine and automatically switch the SFR page to the corresponding peripheral or function's SFR page. This bit is used to control this autopaging function.
0: SFR Automatic Paging disabled. C8051 core will not automatically change to the appropriate SFR page (i.e., the SFR page that contains the SFRs for the peripheral/function that was the source of the interrupt).
1: SFR Automatic Paging enabled. Upon interrupt, the C8051 will switch the SFR page to the page that contains the SFRs for the peripheral or function that is the source of the interrupt.

Figure 13.10. SFRPAGE: SFR Page 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: 0x84
SFR Page: All Pages

Bits7-0: SFR Page Bits: Byte Represents the SFR Page the C8051 MCU uses when reading or modifying SFRs.
Write: Sets the SFR Page.
Read: Byte is the SFR page the C8051 MCU is using.

When enabled in the SFR Page Control Register (SFRPGCN), the C8051 will automatically switch to the SFR Page that contains the SFRs of the corresponding peripheral/function that caused the interrupt, and return to the previous SFR page upon return from interrupt (unless SFR Stack was altered before a returning from the interrupt).
SFRPAGE is the top byte of the SFR Page Stack, and push/pop events of this stack are caused by interrupts (and **not** by reading/writing to the SFRPAGE register)

Figure 13.22. EIE2: Extended Interrupt Enable 2

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
EDMA0	ES1	ECAN0	EADC2	EWADC2	ET4	EADC1	ET3	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xE7
SFR Page: All Pages

Bit7: EDMA0: Enable DMA0 Interrupt.
This bit sets the masking of the DMA0 Interrupt.
0: Disable DMA0 interrupt.
1: Enable DMA0 interrupt.

Bit6: ES1: Enable UART1 Interrupt.
This bit sets the masking of the UART1 Interrupt.
0: Disable UART1 interrupt.
1: Enable UART1 interrupt.

Bit5: ECAN0: Enable CAN Controller Interrupt.
This bit sets the masking of the CAN Controller Interrupt.
0: Disable CAN Controller Interrupt.
1: Enable interrupt requests generated by the CAN Controller.

Bit4: EADC2: Enable ADC2 End Of Conversion Interrupt.
This bit sets the masking of the ADC2 End of Conversion interrupt.
0: Disable ADC2 End of Conversion interrupt.
1: Enable interrupt requests generated by the ADC2 End of Conversion Interrupt.

Bit3: EWADC2: Enable Window Comparison ADC1 Interrupt.
This bit sets the masking of ADC2 Window Comparison interrupt.
0: Disable ADC2 Window Comparison Interrupt.
1: Enable Interrupt requests generated by ADC2 Window Comparisons.

Bit2: ET4: Enable Timer 4 Interrupt
This bit sets the masking of the Timer 4 interrupt.
0: Disable Timer 4 interrupt.
1: Enable interrupt requests generated by the TF4 flag.

Bit1: EADC1: Enable ADC1 End of Conversion Interrupt.
This bit sets the masking of the ADC1 End of Conversion Interrupt.
0: Disable ADC1 Conversion Interrupt.
1: Enable interrupt requests generated by the ADC1 Conversion Interrupt.

Bit0: ET3: Enable Timer 3 Interrupt.
This bit sets the masking of the Timer 3 interrupt.
0: Disable all Timer 3 interrupts.
1: Enable interrupt requests generated by the TF3 flag.

Figure 13.23. EIP1: Extended Interrupt Priority 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
PADC0	PCP2	PCP1	PCP0	PPCA0	PWADC0	PSMB0	PSPI0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xF6
SFR Page: All Pages

Bit7: PADC0: ADC End of Conversion Interrupt Priority Control.
This bit sets the priority of the ADC0 End of Conversion Interrupt.
0: ADC0 End of Conversion interrupt set to low priority level.
1: ADC0 End of Conversion interrupt set to high priority level.

Bit6: PCP2: Comparator2 (CP2) Interrupt Priority Control.
This bit sets the priority of the CP2 interrupt.
0: CP2 interrupt set to low priority level.
1: CP2 interrupt set to high priority level.

Bit5: PCP1: Comparator1 (CP1) Interrupt Priority Control.
This bit sets the priority of the CP1 interrupt.
0: CP1 interrupt set to low priority level.
1: CP1 interrupt set to high priority level.

Bit4: PCP0: Comparator0 (CP0) Interrupt Priority Control.
This bit sets the priority of the CP0 interrupt.
0: CP0 interrupt set to low priority level.
1: CP0 interrupt set to high priority level.

Bit3: 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.

Bit2: 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.

Bit1: PSMB0: System Management Bus (SMBus0) Interrupt Priority Control.
This bit sets the priority of the SMBus0 interrupt.
0: SMBus interrupt set to low priority level.
1: SMBus interrupt set to high priority level.

Bit0: PSPI0: Serial Peripheral Interface (SPI0) Interrupt Priority Control.
This bit sets the priority of the SPI0 interrupt.
0: SPI0 interrupt set to low priority level.
1: SPI0 interrupt set to high priority level.

Figure 13.24. EIP2: Extended Interrupt Priority 2

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
PDMA0	PS1	PCAN0	PADC2	PWADC2	PT4	PADC1	PT3	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xF7
SFR Page: All Pages

Bit7: PDMA0: DMA0 Interrupt Priority Control.
This bit sets the priority of the DMA0 interrupt.
0: DMA0 interrupt set to low priority.
1: DMA0 interrupt set to high priority.

Bit6: PS1: UART1 Interrupt Priority Control.
This bit sets the priority of the UART1 interrupt.
0: UART1 interrupt set to low priority.
1: UART1 interrupt set to high priority.

Bit5: PCAN0: CAN Interrupt Priority Control.
This bit sets the priority of the CAN Interrupt.
0: CAN Interrupt set to low priority level.
1: CAN Interrupt set to high priority level.

Bit4: PADC2: ADC2 End Of Conversion Interrupt Priority Control.
This bit sets the priority of the ADC2 End of Conversion interrupt.
0: ADC2 End of Conversion interrupt set to low priority.
1: ADC2 End of Conversion interrupt set to high priority.

Bit3: PWADC2: ADC2 Window Comparator Interrupt Priority Control.
0: ADC2 Window interrupt set to low priority.
1: ADC2 Window interrupt set to high priority.

Bit2: PT4: Timer 4 Interrupt Priority Control.
This bit sets the priority of the Timer 4 interrupt.
0: Timer 4 interrupt set to low priority.
1: Timer 4 interrupt set to high priority.

Bit1: PADC1: ADC End of Conversion Interrupt Priority Control.
This bit sets the priority of the ADC1 End of Conversion Interrupt.
0: ADC1 End of Conversion interrupt set to low priority level.
1: ADC1 End of Conversion interrupt set to high priority level.

Bit0: PT3: Timer 3 Interrupt Priority Control.
This bit sets the priority of the Timer 3 interrupts.
0: Timer 3 interrupt set to low priority level.
1: Timer 3 interrupt set to high priority level.

Figure 14.4. RSTSRC: Reset Source Register

R	R/W	R/W	R/W	R	R/W	R/W	R/W	Reset Value
-	CNVRSEF	C0RSEF	SWRSEF	WDTRSF	MCDRSF	PORSF	PINRSF	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xEF
SFR Page: 0

Bit7: Reserved.

Bit6: CNVRSEF: Convert Start Reset Source Enable and Flag
Write: 0: CNVSTR2 is not a reset source.
1: CNVSTR2 is a reset source (active low).
Read: 0: Source of prior reset was not CNVSTR2.
1: Source of prior reset was CNVSTR2.

Bit5: C0RSEF: Comparator0 Reset Enable and Flag.
Write: 0: Comparator0 is not a reset source.
1: Comparator0 is a reset source (active low).
Read: 0: Source of last reset was not Comparator0.
1: Source of last reset was Comparator0.

Bit4: SWRSF: Software Reset Force and Flag.
Write: 0: No effect.
1: Forces an internal reset. /RST pin is not effected.
Read: 0: Source of last reset was not a write to the SWRSF bit.
1: Source of last reset was a write to the SWRSF bit.

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

Bit2: MCDRSF: Missing Clock Detector Flag.
Write: 0: Missing Clock Detector disabled.
1: Missing Clock Detector enabled; triggers a reset if a missing clock condition is detected.
Read: 0: Source of last reset was not a Missing Clock Detector timeout.
1: Source of last reset was a Missing Clock Detector timeout.

Bit1: PORSF: Power-On Reset Flag.
Write: If the VDD monitor circuitry is enabled (by tying the MONEN pin to a logic high state), this bit can be written to select or de-select the VDD monitor as a reset source.
0: De-select the VDD monitor as a reset source.
1: Select the VDD monitor as a reset source.
Important: At power-on, the VDD monitor is enabled/disabled using the external VDD monitor enable pin (MONEN). The PORSF bit does not disable or enable the VDD monitor circuit. It simply selects the VDD monitor as a reset source.
Read: This bit is set whenever a power-on reset occurs. This may be due to a true power-on reset or a VDD monitor reset. In either case, data memory should be considered indeterminate following the reset.
0: Source of last reset was not a power-on or VDD monitor reset.
1: Source of last reset was a power-on or VDD monitor reset.
Note: When this flag is read as '1', all other reset flags are indeterminate.

Bit0: PINRSF: HW Pin Reset Flag.
Write: 0: No effect.
1: Forces a Power-On Reset. /RST is driven low.
Read: 0: Source of prior reset was not /RST pin.
1: Source of prior reset was /RST pin.

18.2. Ports 4 through 7 (C8051F060/2/4/6 only)

All Port pins on Ports 4 through 7 can be accessed as General-Purpose I/O (GPIO) pins by reading and writing the associated Port Data registers (See Figure 18.19, Figure 18.21, Figure 18.23, and Figure 18.25), a set of SFRs which are byte-addressable. Note that Port 4 has only three pins: P4.5, P4.6, and P4.7. Note also that the Port 4, 5, 6, and 7 registers are located on SFR Page F. The SFRPAGE register must be set to 0x0F to access these Port registers.

A Read of a Port Data register (or Port bit) will always return the logic state present at the pin itself, regardless of whether the Crossbar has allocated the pin for peripheral use or not. An exception to this occurs during the execution of a *read-modify-write* instruction (ANL, ORL, XRL, CPL, INC, DEC, DJNZ, JBC, CLR, SETB, and the bitwise MOV write operation). During the *read* cycle of the *read-modify-write* instruction, it is the contents of the Port Data register, not the state of the Port pins themselves, which is read.

18.2.1. Configuring Ports which are not Pinned Out

Although P3, P4, P5, P6, and P7 are not brought out to pins on the C8051F061/3/5/7 devices, the Port Data registers are still present and can be used by software. Because the digital input paths also remain active, it is recommended that these pins not be left in a ‘floating’ state in order to avoid unnecessary power dissipation arising from the inputs floating to non-valid logic levels. This condition can be prevented by any of the following:

1. Leave the weak pull-up devices enabled by setting WEAKPUD (XBR2.7) to a logic 0.
2. Configure the output modes of P3, P4, P5, P6, and P7 to “Push-Pull” by writing 0xFF to the associated output mode register (PnMDOUT).
3. Force the output states of P3, P4, P5, P6, and P7 to logic 0 by writing zeros to the Port Data registers: P3 = 0x00, P4 = 0x00, P5 = 0x00, P6= 0x00, and P7 = 0x00.

18.2.2. Configuring the Output Modes of the Port Pins

The output mode of each port pin can be configured to be either Open-Drain or Push-Pull. In the Push-Pull configuration, a logic 0 in the associated bit in the Port Data register will cause the Port pin to be driven to GND, and a logic 1 will cause the Port pin to be driven to VDD. In the Open-Drain configuration, a logic 0 in the associated bit in the Port Data register will cause the Port pin to be driven to GND, and a logic 1 will cause the Port pin to assume a high-impedance state. The Open-Drain configuration is useful to prevent contention between devices in systems where the Port pin participates in a shared interconnection in which multiple outputs are connected to the same physical wire.

The output modes of the Port pins on Ports 4 through 7 are determined by the bits in their respective PnMDOUT Output Mode Registers. Each bit in PnMDOUT controls the output mode of its corresponding port pin (see Figure 18.20, Figure 18.22, Figure 18.24, and Figure 18.26). For example, to place Port pin 5.3 in push-pull mode (digital output), set P5MDOUT.3 to logic 1. All port pins default to open-drain mode upon device reset.

18.2.3. Configuring Port Pins as Digital Inputs

A Port pin is configured as a digital input by setting its output mode to “Open-Drain” and writing a logic 1 to the associated bit in the Port Data register. For example, P7.7 is configured as a digital input by setting P7MDOUT.7 to a logic 0 and P7.7 to a logic 1.

18.2.4. Weak Pull-ups

By default, each Port pin has an internal weak pull-up device enabled which provides a resistive connection (about 100 kΩ) between the pin and VDD. The weak pull-up devices can be globally disabled by writ-

Figure 18.21. P5: Port5 Data Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
P5.7	P5.6	P5.5	P5.4	P5.3	P5.2	P5.1	P5.0	11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Bit Addressable

SFR Address: 0xD8
SFR Page: F

Bits7-0: P5.[7:0]: Port5 Output Latch Bits.
Write - Output appears on I/O pins.
0: Logic Low Output.
1: Logic High Output (open, if corresponding P5MDOUT bit = 0). See Figure 18.22.
Read - Returns states of I/O pins.
0: P5.n pin is logic low.
1: P5.n pin is logic high.

Note: P5.[7:0] can be driven by the External Data Memory Interface (as Address[15:8] in Non-multiplexed mode). See Section “17. External Data Memory Interface and On-Chip XRAM” on page 187 for more information about the External Memory Interface.

Figure 18.22. P5MDOUT: Port5 Output Mode 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: 0x9D
SFR Page: F

Bits7-0: P5MDOUT.[7:0]: Port5 Output Mode Bits.
0: Port Pin output mode is configured as Open-Drain.
1: Port Pin output mode is configured as Push-Pull.

Please refer to the Bosch CAN User's Guide for information on the function and use of the Message Handler Registers.

19.2.4. CIP-51 MCU Special Function Registers

C8051F060/1/2/3 peripherals are modified, monitored, and controlled using Special Function Registers (SFRs). Most of the CAN Controller registers cannot be accessed *directly* using the SFRs. Three of the CAN Controller's registers may be accessed directly with SFRs. All other CAN Controller registers are accessed indirectly using three CIP-51 MCU SFRs: the CAN Data Registers (CAN0DATH and CAN0DATL) and CAN Address Register (CAN0ADR). In this way, there are a total of five CAN registers used to configure and run the CAN Controller.

19.2.5. Using CAN0ADR, CAN0DATH, and CANDATL To Access CAN Registers

Each CAN Controller Register has an index number (see Table below). The CAN register address space is 128 words (256 bytes). A CAN register is accessed via the CAN Data Registers (CAN0DATH and CAN0DATL) when a CAN register's index number is placed into the CAN Address Register (CAN0ADR). For example, if the Bit Timing Register is to be configured with a new value, CAN0ADR is loaded with 0x03. The low byte of the desired value is accessed using CAN0DATL and the high byte of the bit timing register is accessed using CAN0DATH. CAN0DATL is bit addressable for convenience. To load the value 0x2304 into the Bit Timing Register:

```
CAN0ADR = 0x03;     // Load Bit Timing Register's index (Table 18.1)
CAN0DATH = 0x23;    // Move the upper byte into data reg high byte
CAN0DATL = 0x04;    // Move the lower byte into data reg low byte
```

Note: CAN0CN, CAN0STA, and CAN0TST may be accessed either by using the index method, or by direct access with the CIP-51 MCU SFRs. CAN0CN is located at SFR location 0xF8/SFR page 1 (Figure 19.6), CAN0TST at 0xDB/SFR page 1 (Figure 19.7), and CAN0STA at 0xC0/SFR page 1 (Figure 19.8).

19.2.6. CAN0ADR Autoincrement Feature

For ease of programming message objects, CAN0ADR features autoincrementing for the index ranges 0x08 to 0x12 (Interface Registers 1) and 0x20 to 0x2A (Interface Registers 2). When the CAN0ADR register has an index in these ranges, **the CAN0ADR will autoincrement by 1 to point to the next CAN register 16-bit word upon a read/write of CAN0DATL**. This speeds programming of the frequently programmed interface registers when configuring message objects.

NOTE: Table below supersedes Figure 5 in section 3, "Programmer's Model" of the Bosch CAN User's Guide.

Table 19.1. CAN Register Index and Reset Values

CAN Register Index	Register name	Reset Value	Notes
0x00	CAN Control Register	0x0001	Accessible in CIP-51 SFR Map
0x01	Status Register	0x0000	Accessible in CIP-51 SFR Map
0x02	Error Register	0x0000	Read Only
0x03	Bit Timing Register	0x2301	Write Enabled by CCE Bit in CAN0CN

20. System Management BUS / I2C BUS (SMBUS0)

The SMBus0 I/O interface is a two-wire, bi-directional serial bus. SMBus0 is compliant with the System Management Bus Specification, version 1.1, and compatible with the I2C serial bus. Reads and writes to the interface by the system controller are byte oriented with the SMBus0 interface autonomously controlling the serial transfer of the data. A method of extending the clock-low duration is available to accommodate devices with different speed capabilities on the same bus.

SMBus0 may operate as a master and/or slave, and may function on a bus with multiple masters. SMBus0 provides control of SDA (serial data), SCL (serial clock) generation and synchronization, arbitration logic, and START/STOP control and generation.

Figure 20.1. SMBus0 Block Diagram

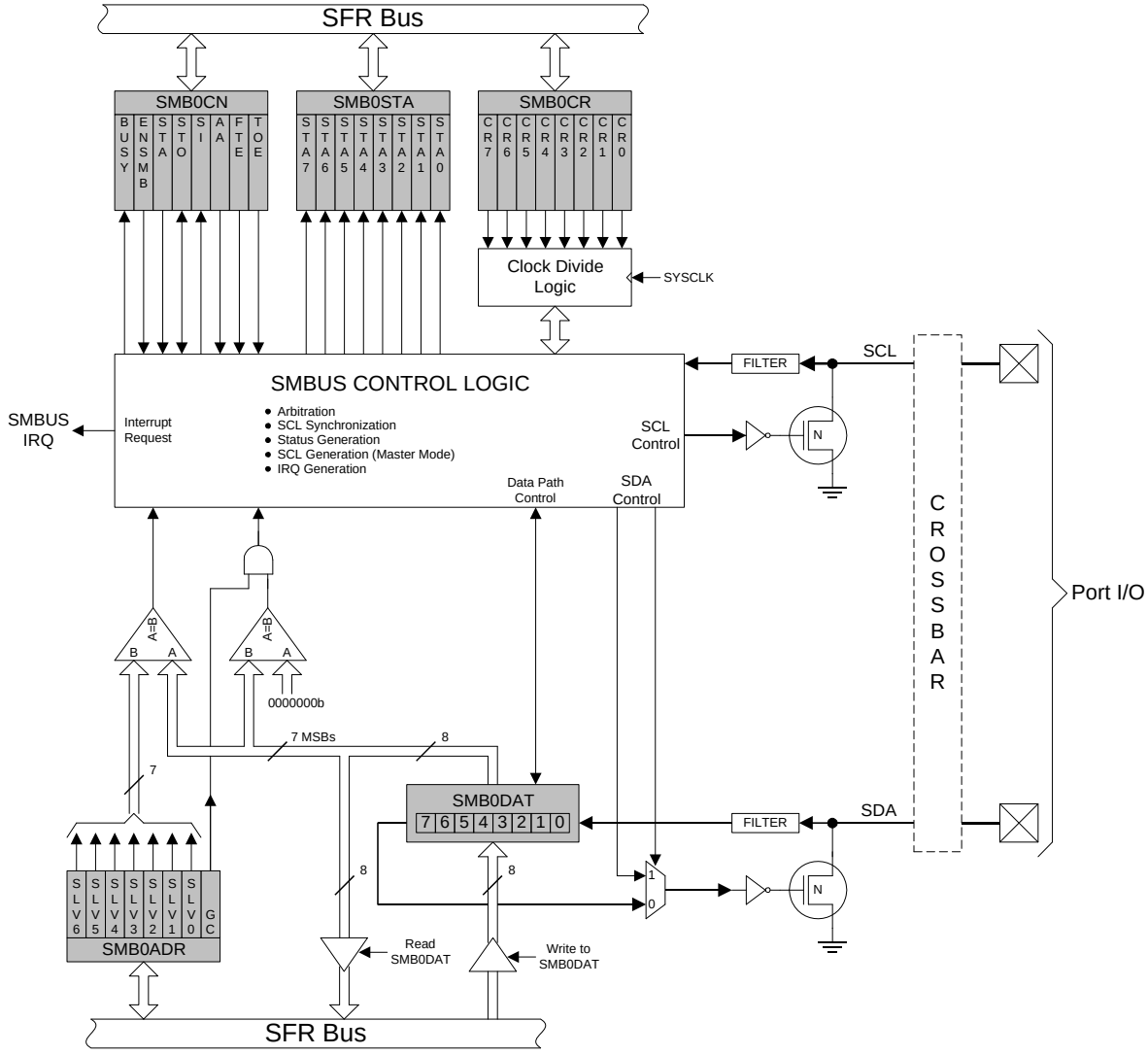


Table 21.1. SPI Slave Timing Parameters

Parameter	Description	Min	Max	Units
Master Mode Timing† (See Figure 21.12 and Figure 21.13)				
T_{MCKH}	SCK High Time	$1*T_{SYSCLK}$		ns
T_{MCKL}	SCK Low Time	$1*T_{SYSCLK}$		ns
T_{MIS}	MISO Valid to SCK Shift Edge	$1*T_{SYSCLK}$ + 20		ns
T_{MIH}	SCK Shift Edge to MISO Change	0		ns
Slave Mode Timing† (See Figure 21.14 and Figure 21.15)				
T_{SE}	NSS Falling to First SCK Edge	$2*T_{SYSCLK}$		ns
T_{SD}	Last SCK Edge to NSS Rising	$2*T_{SYSCLK}$		ns
T_{SEZ}	NSS Falling to MISO Valid		$4*T_{SYSCLK}$	ns
T_{SDZ}	NSS Rising to MISO High-Z		$4*T_{SYSCLK}$	ns
T_{CKH}	SCK High Time	$5*T_{SYSCLK}$		ns
T_{CKL}	SCK Low Time	$5*T_{SYSCLK}$		ns
T_{SIS}	MOSI Valid to SCK Sample Edge	$2*T_{SYSCLK}$		ns
T_{SIH}	SCK Sample Edge to MOSI Change	$2*T_{SYSCLK}$		ns
T_{SOH}	SCK Shift Edge to MISO Change		$4*T_{SYSCLK}$	ns
T_{SLH}	Last SCK Edge to MISO Change (CKPHA = 1 ONLY)	$6*T_{SYSCLK}$	$8*T_{SYSCLK}$	ns
†T_{SYSCLK} is equal to one period of the device system clock (SYSCLK).				

25.2.6. 16-Bit Pulse Width Modulator Mode

Each PCA0 module may also be operated in 16-Bit PWM mode. In this mode, the 16-bit capture/compare module defines the number of PCA0 clocks for the low time of the PWM signal. When the PCA0 counter matches the module contents, the output on CEXn is asserted high; when the counter overflows, CEXn is asserted low. To output a varying duty cycle, new value writes should be synchronized with PCA0 CCFn match interrupts. 16-Bit PWM Mode is enabled by setting the ECOMn, PWMn, and PWM16n bits in the PCA0CPMn register. For a varying duty cycle, CCFn should also be set to logic 1 to enable match interrupts. The duty cycle for 16-Bit PWM Mode is given by Equation 25.3.

Important Note About Capture/Compare Registers: When writing a 16-bit value to the PCA0 Capture/Compare registers, the low byte should always be written first. Writing to PCA0CPLn clears the ECOMn bit to ‘0’; writing to PCA0CPHn sets ECOMn to ‘1’.

Equation 25.3. 16-Bit PWM Duty Cycle

$$DutyCycle = \frac{(65536 - PCA0CPn)}{65536}$$

Figure 25.9. PCA 16-Bit PWM Mode