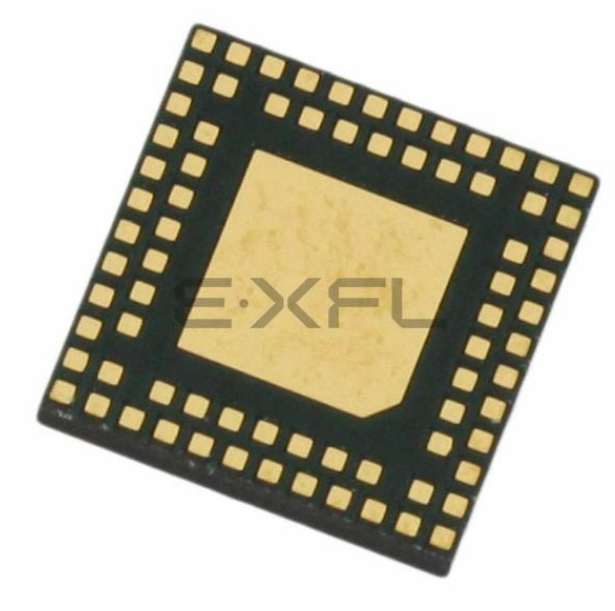
Silicon Labs - C8051F968-B-GM Datasheet



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	Active
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	DMA, LCD, POR, PWM, WDT
Number of I/O	57
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4.25K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.8V
Data Converters	A/D 16x10b/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	76-VFQFN Dual Rows, Exposed Pad
Supplier Device Package	76-DQFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f968-b-gm

3.1. DQFN-76 Package Specifications

3.1.1. Package Drawing

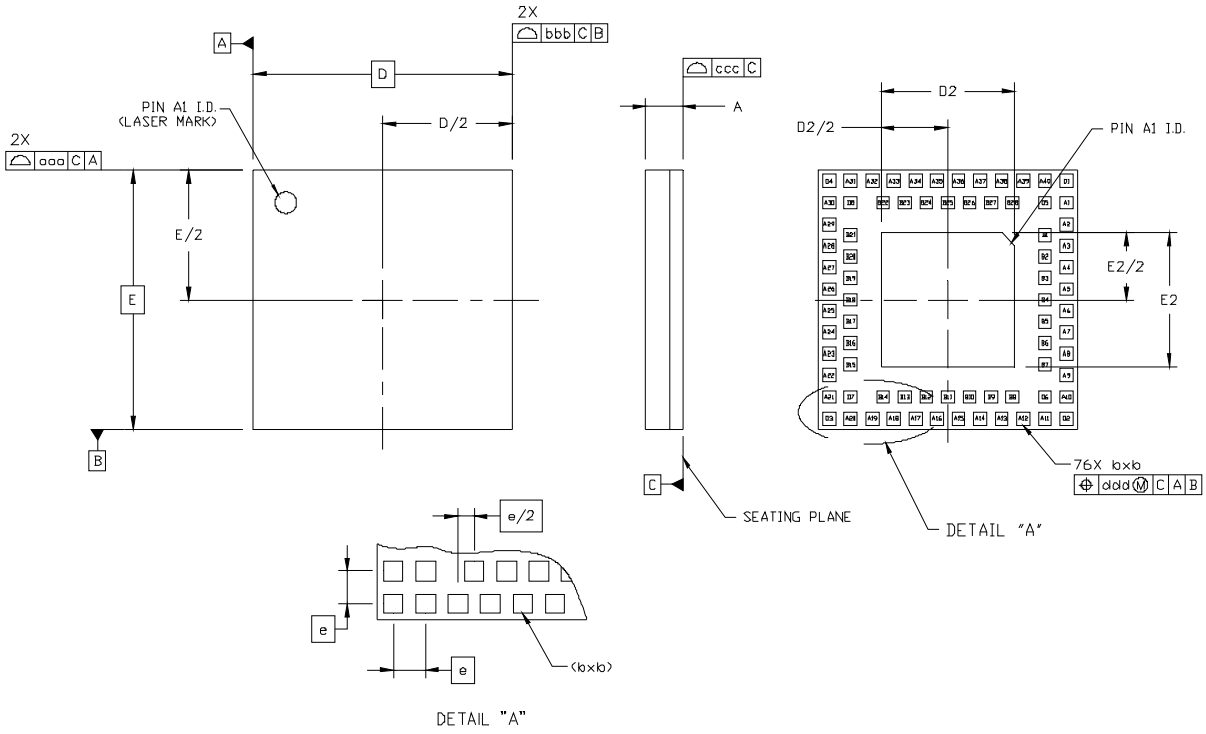


Figure 3.4. DQFN-76 Package Drawing

Table 3.2. DQFN-76 Package Dimensions

Dimension	Min	Typ	Max
A	0.74	0.84	0.94
b	0.25	0.30	0.35
D		6.00 BSC	
D2	3.00	3.10	3.20
e		0.50 BSC	
E		6.00 BSC	

Dimension	Min	Typ	Max
E2	3.00	3.10	3.20
aaa	—	—	0.10
bbb	—	—	0.10
ddd	—	—	0.08
eee	—	—	0.10

Notes:
1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. Dimensioning and Tolerancing per ANSI Y14.5M-1994.
3. Recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

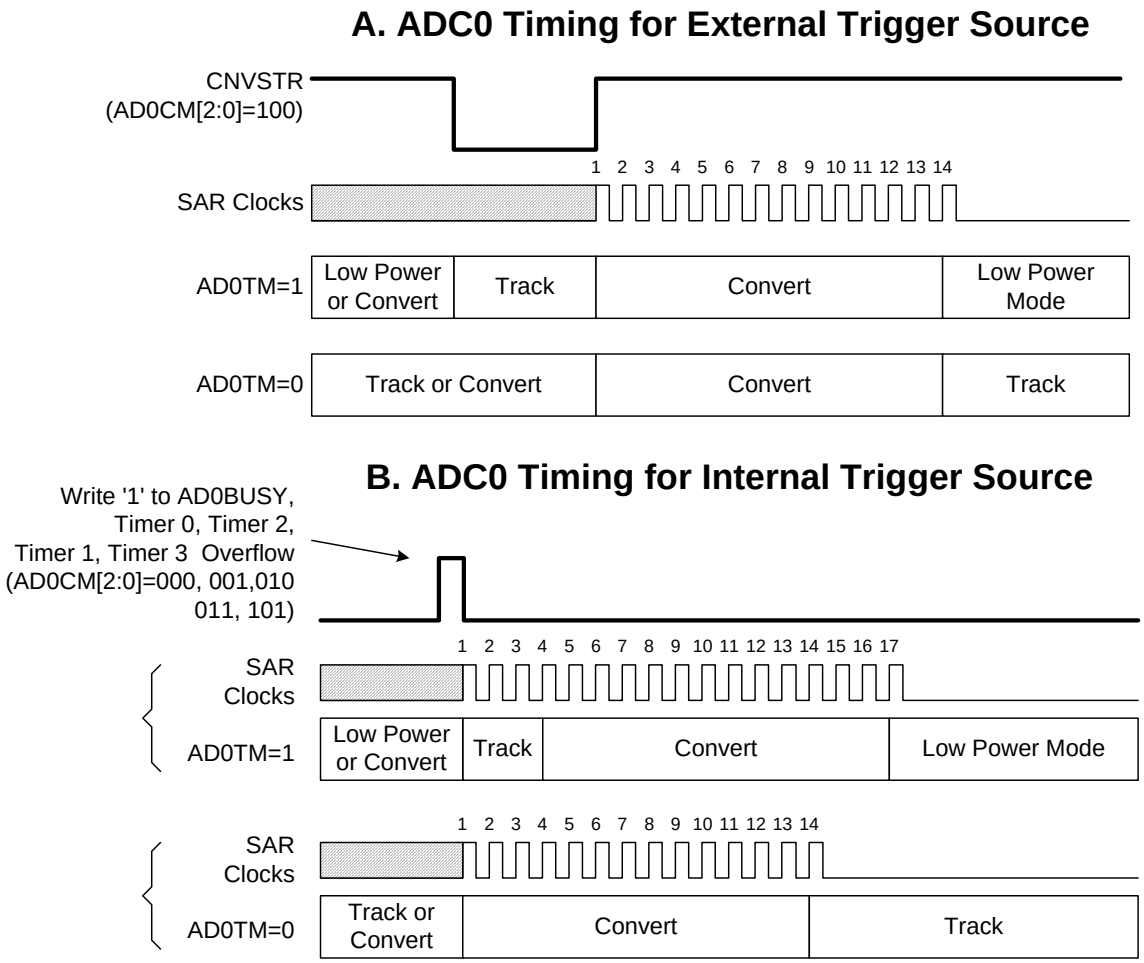


Figure 5.2. 10-Bit ADC Track and Conversion Example Timing (BURSTEN = 0)

SFR Definition 8.3. SP: Stack Pointer

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

SFR Page = All Pages; SFR Address = 0x81

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

SFR Definition 8.4. ACC: Accumulator

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

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

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

SFR Definition 8.5. B: B Register

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

SFR Page = All Pages; SFR Address = 0xF0; Bit-Addressable

Bit	Name	Function
7:0	B[7:0]	**B Register.** This register serves as a second accumulator for certain arithmetic operations.

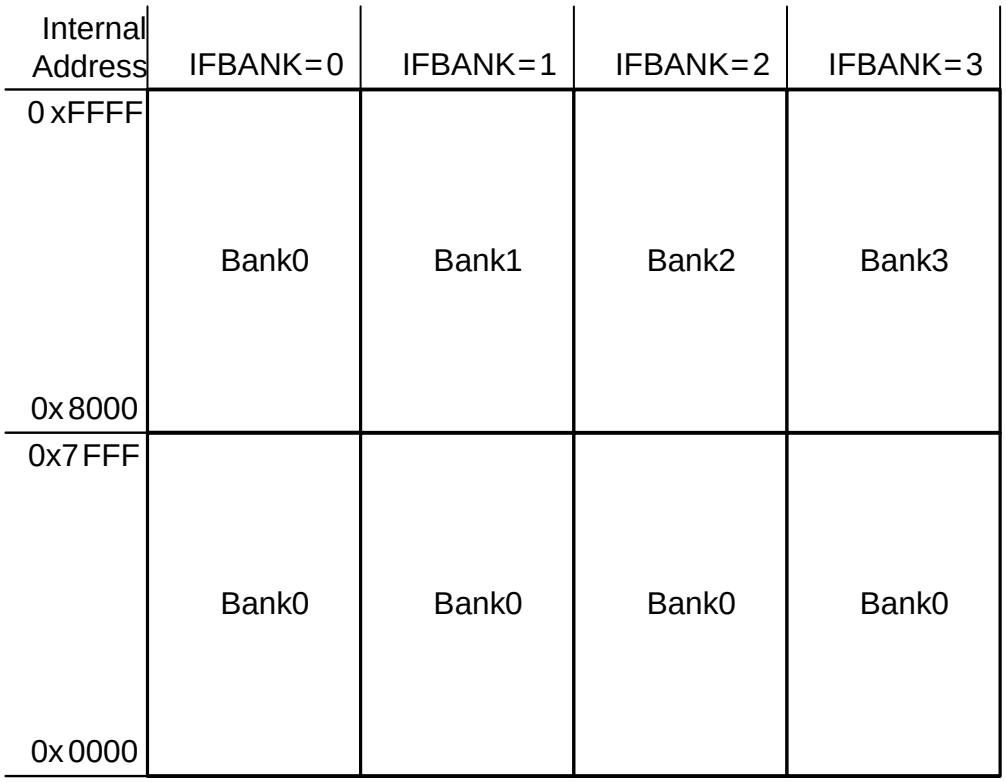


Figure 9.3. Address Memory Map for Instruction Fetches

SFR Definition 11.3. DMA0MINT: DMA0 Mid-Point Interrupt

Bit	7	6	5	4	3	2	1	0
Name		CH6_MINT	CH5_MINT	CH4_MINT	CH3_MINT	CH2_MINT	CH1_MINT	CH0_MINT
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x2; SFR Address = 0xD4

Bit	Name	Function
7	Unused	Read = 0b, Write = Don't Care
6	CH6_MINT	**Channel 6 Mid-Point Interrupt Flag.** 0: Mid-Point interrupt has not occured on channel 6. 1: Mid-Point interrupt has not occured on channel 6.
5	CH5_MINT	**Channel 5 Mid-Point Interrupt Flag.** 0: Mid-Point interrupt has not occured on channel 5. 1: Mid-Point interrupt has not occured on channel 5.
4	CH4_MINT	**Channel 4 Mid-Point Interrupt Flag.** 0: Mid-Point interrupt has not occured on channel 4. 1: Mid-Point interrupt has not occured on channel 4**.**
3	CH3_MINT	**Channel 3 Mid-Point Interrupt Flag.** 0: Mid-Point interrupt has not occured on channel 3. 1: Mid-Point interrupt has not occured on channel 3.
2	CH2_MINT	**Channel 2 Mid-Point Interrupt Flag.** 0: Mid-Point interrupt has not occured on channel 2. 1: Mid-Point interrupt has not occured on channel 2.
1	CH1_MINT	**Channel 1 Mid-Point Interrupt Flag.** 0: Mid-Point interrupt has not occured on channel 1. 1: Mid-Point interrupt has not occured on channel 1.
0	CH0_MINT	**Channel 0 Mid-Point Interrupt Flag.** 0: Mid-Point interrupt has not occured on channel 0. 1: Mid-Point interrupt has not occured on channel 0.

Note: Mid-point Interrupt flag is set when the offset address DMA0NAOH/L equals to half of data transfer size DMA0NSZH/L if the transfer size is an even number or half of data transfer size DMA0NSZH/L plus one if the transfer size is an odd number. This flag must be cleared by software or system reset.The mid-point interrupt is enabled by setting bit 6 of DMA0NCF with DMA0SEL configured for the corresponding channel.

SFR Definition 13.2. CRC1IN: CRC1 Data IN

Bit	7	6	5	4	3	2	1	0
Name	CRC1IN[7:0]							
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 Page = 0x2; SFR Address = 0xB9; Not Bit-Addressable

Bit	Name	Function
7:0	CRC1IN[7:0]	**CRC1Data IN.** CRC Data should be sequentially written, one byte at a time, to the CRC1IN Data input SFR. When the CRC1 module is used with the DMA, the DMA will write directly to this SFR.

SFR Definition 13.3. CRC1POLL: CRC1 Polynomial LSB

Bit	7	6	5	4	3	2	1	0
Name	CRC1POLL[7:0]							
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 Page = 0x2; SFR Address = 0xBC; Not Bit-Addressable

Bit	Name	Function
7:0	CRC1POLL[7:0]	**CRC1 Polynomial LSB.**

SFR Definition 13.4. CRC1POLH: CRC1 Polynomial MSB

Bit	7	6	5	4	3	2	1	0
Name	CRC1POLH[7:0]							
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 Page = 0x2; SFR Address = 0xBD; Not Bit-Addressable

Bit	Name	Function
7:0	CRC1POLH[7:0]	**CRC1 Polynomial MSB.**

SFR Definition 16.3. SFRNEXT: SFR Next

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

;SFR Page = All Pages; SFR Address = 0x85

Bit	Name	Function
7:0	SFRNEXT[7:0]	**SFR Page Bits.** This is the value that will go to the SFR Page register upon a return from interrupt. Write: Sets the SFR Page contained in the second byte of the SFR Stack. This will cause the SFRPAGE SFR to have this SFR page value upon a return from interrupt. Read: Returns the value of the SFR page contained in the second byte of the SFR stack. SFR page context is retained upon interrupts/return from interrupts in a 3 byte SFR Page Stack: SFRPAGE is the first entry, SFRNEXT is the second, and SFRLAST is the third entry. The SFR stack bytes may be used alter the context in the SFR Page Stack, and will not cause the stack to “push” or “pop”. Only interrupts and return from interrupts cause pushes and pops of the SFR Page Stack.

SFR Definition 17.3. EIE1: Extended Interrupt Enable 1

Bit	7	6	5	4	3	2	1	0
Name	ET3	ECP1	ECP0	EPCA0	EADC0	EWADC0	ERTC0A	ESMB0
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 Page = All Pages; SFR Address = 0xE6

Bit	Name	Function
7	ET3	**Enable Timer 3 Interrupt.** This bit sets the masking of the Timer 3 interrupt. 0: Disable Timer 3 interrupts. 1: Enable interrupt requests generated by the TF3L or TF3H flags.
6	ECP1	**Enable Comparator1 (CP1) Interrupt.** This bit sets the masking of the CP1 interrupt. 0: Disable CP1 interrupts. 1: Enable interrupt requests generated by the CP1RIF or CP1FIF flags.
5	ECP0	**Enable Comparator0 (CP0) Interrupt.** This bit sets the masking of the CP0 interrupt. 0: Disable CP0 interrupts. 1: Enable interrupt requests generated by the CP0RIF or CP0FIF flags.
4	EPCA0	**Enable Programmable Counter Array (PCA0) Interrupt.** This bit sets the masking of the PCA0 interrupts. 0: Disable all PCA0 interrupts. 1: Enable interrupt requests generated by PCA0.
3	EADC0	**Enable ADC0 Conversion Complete Interrupt.** This bit sets the masking of the ADC0 Conversion Complete interrupt. 0: Disable ADC0 Conversion Complete interrupt. 1: Enable interrupt requests generated by the AD0INT flag.
2	EWADC0	**Enable Window Comparison ADC0 Interrupt.** This bit sets the masking of ADC0 Window Comparison interrupt. 0: Disable ADC0 Window Comparison interrupt. 1: Enable interrupt requests generated by ADC0 Window Compare flag (AD0WINT).
1	ERTC0A	**Enable SmaRTClock Alarm Interrupts.** This bit sets the masking of the SmaRTClock Alarm interrupt. 0: Disable SmaRTClock Alarm interrupts. 1: Enable interrupt requests generated by a SmaRTClock Alarm.
0	ESMB0	**Enable SMBus (SMB0) Interrupt.** This bit sets the masking of the SMB0 interrupt. 0: Disable all SMB0 interrupts. 1: Enable interrupt requests generated by SMB0.

SFR Definition 17.5. EIE2: Extended Interrupt Enable 2

Bit	7	6	5	4	3	2	1	0
Name	EAES0	EENC0	EDMA0	EPC0	ESPI1	ERTC0F	EMAT	EWARN
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 Page = All Pages;SFR Address = 0xE7

Bit	Name	Function
7	EAES0	**Enable AES0 Interrupt.** This bit sets the masking of AES0 interrupts. 0: Disable all AES0 interrupts. 1: Enable interrupt requests generated by AES0.
6	EENC0	**Enable Encoder (ENC0) Interrupt.** This bit sets the masking of ENC0 interrupts. 0: Disable all ENC0 interrupts. 1: Enable interrupt requests generated by ENC0.
5	EDMA0	**Enable DMA0 Interrupt.** This bit sets the masking of DMA0 interrupts. 0: Disable all DMA0 interrupts. 1: Enable interrupt requests generated by DMA0.
4	EPC0	**Enable Pulse Counter (PC0) Interrupt.** This bit sets the masking of PC0 interrupts. 0: Disable all PC0 interrupts. 1: Enable interrupt requests generated by PC0.
3	ESPI1	**Enable Serial Peripheral Interface (SPI1) Interrupt.** This bit sets the masking of the SPI1 interrupts. 0: Disable all SPI1 interrupts. 1: Enable interrupt requests generated by SPI1.
2	ERTC0F	**Enable SmaRTClock Oscillator Fail Interrupt.** This bit sets the masking of the SmaRTClock Alarm interrupt. 0: Disable SmaRTClock Alarm interrupts. 1: Enable interrupt requests generated by SmaRTClock Alarm.
1	EMAT	**Enable Port Match Interrupts.** This bit sets the masking of the Port Match Event interrupt. 0: Disable all Port Match interrupts. 1: Enable interrupt requests generated by a Port Match.
0	EWARN	**Enable VDD/DC+ Supply Monitor Early Warning Interrupt.** This bit sets the masking of the VDD/DC+ Supply Monitor Early Warning interrupt. 0: Disable the VDD/DC+ Supply Monitor Early Warning interrupt. 1: Enable interrupt requests generated by VDD/DC+ Supply Monitor.

19. Power Management

C8051F96x devices support 6 power modes: Normal, Idle, Stop, Low Power Idle, Suspend, and Sleep. The power management unit (PMU0) allows the device to enter and wake-up from the available power modes. A brief description of each power mode is provided in Table 19.1. Detailed descriptions of each mode can be found in the following sections.

Table 19.1. Power Modes

Power Mode	Description	Wake-Up Sources	Power Savings
Normal	Device fully functional	N/A	Excellent MIPS/mW
Idle	All peripherals fully functional. Very easy to wake up.	Any Interrupt.	Good No Code Execution
Stop	Legacy 8051 low power mode. A reset is required to wake up.	Any Reset.	Good No Code Execution Precision Oscillator Disabled
Low Power Idle	Improved Idle mode that uses clock gating to save power.	Any Interrupt	Very Good No Code Execution Selective Clock Gating
Suspend	Similar to Stop Mode, but very fast wake-up time and code resumes execution at the next instruction.	SmaRTClock, Port Match, Comparator0, $\overline{\text{RST}}$ pin, Pulse Counter VBAT Monitor.	Very Good No Code Execution All Internal Oscillators Disabled System Clock Gated
Sleep	Ultra Low Power and flexible wake-up sources. Code resumes execution at the next instruction.	SmaRTClock, Port Match, Comparator0, $\overline{\text{RST}}$ pin, Pulse Counter VBAT Monitor.	Excellent Power Supply Gated All Oscillators except SmaRT-Clock Disabled

In battery powered systems, the system should spend as much time as possible in sleep mode in order to preserve battery life. When a task with a fixed number of clock cycles needs to be performed, the device should switch to normal mode, finish the task as quickly as possible, and return to sleep mode. Idle mode, low power idle mode, and suspend mode provide a very fast wake-up time; however, the power savings in these modes will not be as much as in sleep Mode. Stop Mode is included for legacy reasons; the system will be more power efficient and easier to wake up when idle, low power idle, suspend, or sleep mode is used.

Although switching power modes is an integral part of power management, enabling/disabling individual peripherals as needed will help lower power consumption in all power modes. Each analog peripheral can be disabled when not in use or placed in a low power mode. Digital peripherals such as timers or serial busses draw little power whenever they are not in use. Digital peripherals draw no power in Sleep Mode.

20.7. Bypass Mode

The dc-dc converter has a bypass switch (MBYP), see Figure 20.1, which allows the output voltage (VDC) to be directly tied to the input supply (VBATDC), bypassing the dc-dc converter. The bypass switch may be used independently from the dc-dc converter. For example, applications that need to power the VDC supply in the lowest power Sleep mode can turn on the bypass switch prior to turning off the dc-dc converter in order to avoid powering down the external circuitry connected to VDC.

There are two ways to close the bypass switch. Using the first method, Forced Bypass Mode, the FORBYP bit is set to a logic 1 forcing the bypass switch to close. Clearing the FORBYP bit to logic 0 will allow the switch to open if it is not being held closed using Automatic Bypass Mode.

The Automatic Bypass Mode, enabled by setting the AUTOBYP to logic 1, closes the bypass switch when the difference between VBATDC and the programmed output voltage is less than approximately 0.4 V. Once the difference exceeds approximately 0.5 V, the bypass switch is opened unless being held closed by Forced Bypass Mode. In most systems, Automatic Bypass Mode will be left enabled, and the Forced Bypass Mode will be used to close the switch as needed by the system.

20.8. DC-DC Converter Register Descriptions

The SFRs used to configure the dc-dc converter are described in the following register descriptions.

enable signal. The enable signal enables the pull-up resistor when high and disables when low. PC0 is the line to the reed switch. On the right side of PC0 waveform, the line voltage is decreasing towards ground when the pull-up resistors are disabled. Beneath the charging waveform, the arrows represent the sample points. The pulse counter samples the PC0 voltage once the charging completes. The sensed ones and zeros are the sampled data. Finally the integrator waveform illustrates the output of the digital integrator. The integrator is set to 4 initially and counts to down to 0 before toggling the output low. Once the integrator reaches the low state, it needs to count up to 4 before toggling its output to the high state. The debounce logic filters out switch bounce or noise that appears for a short duration.

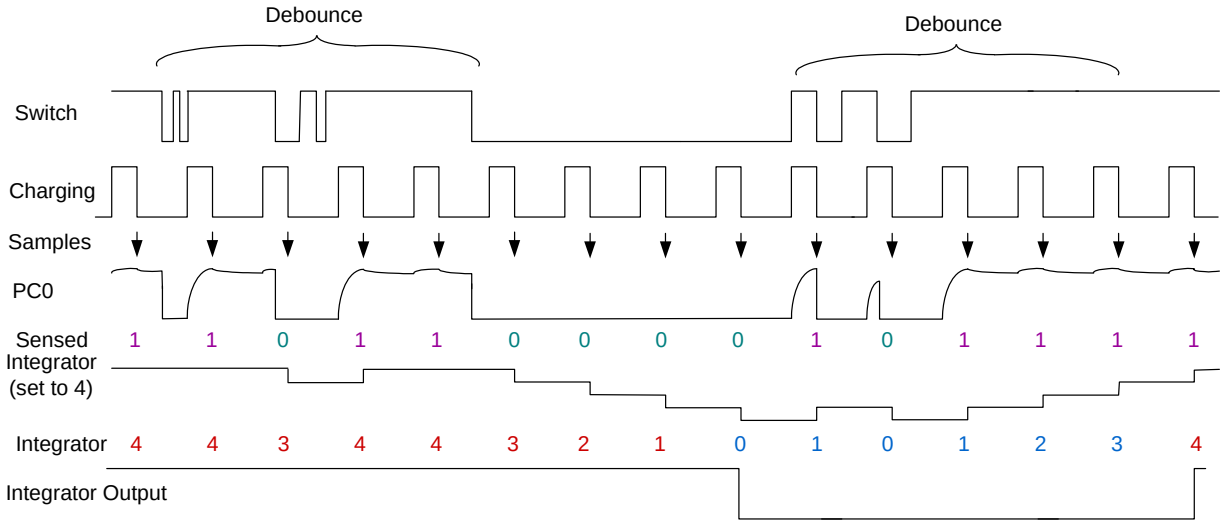


Figure 25.4. Debounce Timing

25.7. Reset Behavior

Unlike most MCU peripherals, an MCU reset does not completely reset the Pulse Counter. This includes a power on reset and all other reset sources. An MCU reset does not clear the counter values. The Pulse Counter SFRs do not reset to a default value upon reset. The 24-bit counter values are persistent unless cleared manually by writing to the PC0MD SFR. Note that if the VBAT voltage ever drops below the minimum operating voltage, this may compromise contents of the counters.

The PC0MD register should normally be written only once after reset. The PC0MD SFR is the master mode register. This register sets the counter mode and sample rate. Writing to the PC0MD SFR also resets the other PC0xxx SFRs.

Note that the RTC clock will reset on an MCU reset, so counting cannot resume until the RTC clock has been re-started.

Firmware should read the reset sources SFR RSTSRC to determine the source of the last reset and initialize the Pulse Counter accordingly.

When the pulse counter resets, it takes some time (typically two RTC clock cycles) to synchronize between internal clock domains. The counters do not increment during this synchronization time.

25.8. Wake up and Interrupt Sources

The Pulse Counter has multiple interrupt and wake-up source conditions. To enable an interrupt, enable the source in the PC0INT0/1 SFRs and enable the Pulse Counter interrupt using bit 4 of the EIE2 bit register. The Pulse Counter interrupt service routine should read the interrupt flags in PC0INT0/1 to determine the source of the interrupt and clear the interrupt flags.

26.3.2. Contrast Control Mode 2 (Minimum Contrast Mode)

In Contrast Control Mode 2, a minimum contrast voltage is maintained, as shown in Figure 26.4. The VLCD supply is powered directly from VBAT as long as VBAT is higher than the programmable VBAT monitor threshold voltage. As soon as the VBAT supply monitor detects that VBAT has dropped below the programmed value, the charge pump will be automatically enabled in order to acheive the desired minimum contrast voltage on VLCD. Minimum Contrast Mode is selected using the following procedure:

1. Clear Bit 2 of the LCD0MSCN register to 0b (LCD0MSCN &= ~0x04)
2. Set Bit 0 of the LCD0MSCF register to 1b (LCD0MSCF |= 0x01)
3. Set Bit 3 of the LCD0PWR register to 1b (LCD0PWR |= 0x08)
4. Set Bit 7 of the LCD0VBMCN register to 1b (LCD0VBMCN |= 0x80)

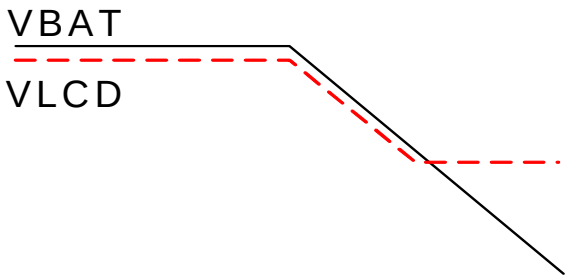


Figure 26.4. Contrast Control Mode 2

26.3.3. Contrast Control Mode 3 (Constant Contrast Mode)

In Contrast Control Mode 3, a constant contrast voltage is maintained. The VLCD supply is regulated to the programmed contrast voltage using a variable resistor between VBAT and VLCD as long as VBAT is higher than the programmable VBAT monitor threshold voltage. As soon as the VBAT supply monitor detects that VBAT has dropped below the programmed value, the charge pump will be automatically enabled in order to acheive the desired contrast voltage on VLCD. Constant Contrast Mode is selected using the following procedure:

1. Set Bit 2 of the LCD0MSCN register to 1b (LCD0MSCN |= 0x04)
2. Clear Bit 0 of the LCD0MSCF register to 0b (LCD0MSCF &= ~0x01)
3. Set Bit 3 of the LCD0PWR register to 1b (LCD0PWR |= 0x08)
4. Set Bit 7 of the LCD0VBMCN register to 1b (LCD0VBMCN |= 0x80)

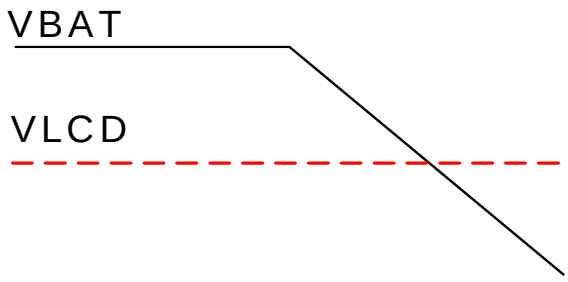


Figure 26.5. Contrast Control Mode 3

SFR Definition 26.5. LCD0MSCF: LCD0 Master Configuration

Bit	7	6	5	4	3	2	1	0
Name							DCENSLP	CHPBYP
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	0

SFR Page = 0x2; SFR Address = 0xAC

Bit	Name	Function
7:2	Reserved	Read = 111111b. Must write 111111b.
1	DCENSLP	**DCDC Converter Enable in Sleep Mode** 0: DCDC is disabled in Sleep Mode. 1: DCDC is enabled in Sleep Mode.
0	CHPBYP	**LCD0 Charge Pump Bypass** This bit should be set to 1b in Contrast Control Mode 1 and Mode 2. 0: LCD0 Charge Pump is not bypassed. 1: LCD0 Charge Pump is bypassed.

SFR Definition 26.6. LCD0PWR: LCD0 Power

Bit	7	6	5	4	3	2	1	0
Name					MODE			
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	1	0	0	1

SFR Page = 0x2; SFR Address = 0xA4

Bit	Name	Function
7:4	Unused	Read = 0000b. Write = don't care.
3	MODE	**LCD0 Contrast Control Mode Selection.** 0: LCD0 Contrast Control Mode 1 or Mode 4 is selected. 1: LCD0 Contrast Control Mode 2 or Mode 3 is selected.
2:0	Reserved	Read = 001b. Must write 001b.

SFR Definition 28.1. SMB0CF: SMBus Clock/Configuration

Bit	7	6	5	4	3	2	1	0
Name	ENSMB	INH	BUSY	EXTHOLD	SMBTOE	SMBFTE	SMBCS[1:0]	
Type	R/W	R/W	R	R/W	R/W	R/W	R/W	
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xC1

Bit	Name	Function
7	ENSMB	**SMBus Enable.** This bit enables the SMBus interface when set to 1. When enabled, the interface constantly monitors the SDA and SCL pins.
6	INH	**SMBus Slave Inhibit.** When this bit is set to logic 1, the SMBus does not generate an interrupt when slave events occur. This effectively removes the SMBus slave from the bus. Master Mode interrupts are not affected.
5	BUSY	**SMBus Busy Indicator.** This bit is set to logic 1 by hardware when a transfer is in progress. It is cleared to logic 0 when a STOP or free-timeout is sensed.
4	EXTHOLD	**SMBus Setup and Hold Time Extension Enable.** This bit controls the SDA setup and hold times according to Table 28.2. 0: SDA Extended Setup and Hold Times disabled. 1: SDA Extended Setup and Hold Times enabled.
3	SMBTOE	**SMBus SCL Timeout Detection Enable.** This bit enables SCL low timeout detection. If set to logic 1, the SMBus forces Timer 3 to reload while SCL is high and allows Timer 3 to count when SCL goes low. If Timer 3 is configured to Split Mode, only the High Byte of the timer is held in reload while SCL is high. Timer 3 should be programmed to generate interrupts at 25 ms, and the Timer 3 interrupt service routine should reset SMBus communication.
2	SMBFTE	**SMBus Free Timeout Detection Enable.** When this bit is set to logic 1, the bus will be considered free if SCL and SDA remain high for more than 10 SMBus clock source periods.
1:0	SMBCS[1:0]	**SMBus Clock Source Selection.** These two bits select the SMBus clock source, which is used to generate the SMBus bit rate. The selected device should be configured according to Equation 28.1. 00: Timer 0 Overflow 01: Timer 1 Overflow 10:Timer 2 High Byte Overflow 11: Timer 2 Low Byte Overflow

Table 28.3. Sources for Hardware Changes to SMB0CN

Bit	Set by Hardware When:	Cleared by Hardware When:
MASTER	• A START is generated.	• A STOP is generated. • Arbitration is lost.
TXMODE	• START is generated. • SMB0DAT is written before the start of an SMBus frame.	• A START is detected. • Arbitration is lost. • SMB0DAT is not written before the start of an SMBus frame.
STA	• A START followed by an address byte is received.	• Must be cleared by software.
STO	• A STOP is detected while addressed as a slave. • Arbitration is lost due to a detected STOP.	• A pending STOP is generated.
ACKRQ	• A byte has been received and an ACK response value is needed (only when hardware ACK is not enabled).	• After each ACK cycle.
ARBLOST	• A repeated START is detected as a MASTER when STA is low (unwanted repeated START). • SCL is sensed low while attempting to generate a STOP or repeated START condition. • SDA is sensed low while transmitting a 1 (excluding ACK bits).	• Each time SI is cleared.
ACK	• The incoming ACK value is low (ACKNOWLEDGE).	• The incoming ACK value is high (NOT ACKNOWLEDGE).
SI	• A START has been generated. • Lost arbitration. • A byte has been transmitted and an ACK/NACK received. • A byte has been received. • A START or repeated START followed by a slave address + R/W has been received. • A STOP has been received.	• Must be cleared by software.

28.4.3. Hardware Slave Address Recognition

The SMBus hardware has the capability to automatically recognize incoming slave addresses and send an ACK without software intervention. Automatic slave address recognition is enabled by setting the EHACK bit in register SMB0ADM to 1. This will enable both automatic slave address recognition and automatic hardware ACK generation for received bytes (as a master or slave). More detail on automatic hardware ACK generation can be found in Section 28.4.2.2.

The registers used to define which address(es) are recognized by the hardware are the SMBus Slave Address register (SFR Definition 28.3) and the SMBus Slave Address Mask register (SFR Definition 28.4). A single address or range of addresses (including the General Call Address 0x00) can be specified using these two registers. The most-significant seven bits of the two registers are used to define which addresses will be ACKed. A 1 in bit positions of the slave address mask SLVM[6:0] enable a comparison between the received slave address and the hardware's slave address SLV[6:0] for those bits. A 0 in a bit of the slave address mask means that bit will be treated as a "don't care" for comparison purposes. In this case, either a 1 or a 0 value are acceptable on the incoming slave address. Additionally, if the GC bit in register SMB0ADR is set to 1, hardware will recognize the General Call Address (0x00). Table 28.4 shows some example parameter settings and the slave addresses that will be recognized by hardware under those conditions.

31. Enhanced Serial Peripheral Interface with DMA Support (SPI1)

The Enhanced Serial Peripheral Interface (SPI1) provides access to a flexible, full-duplex synchronous serial bus. SPI1 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 SPI1 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.

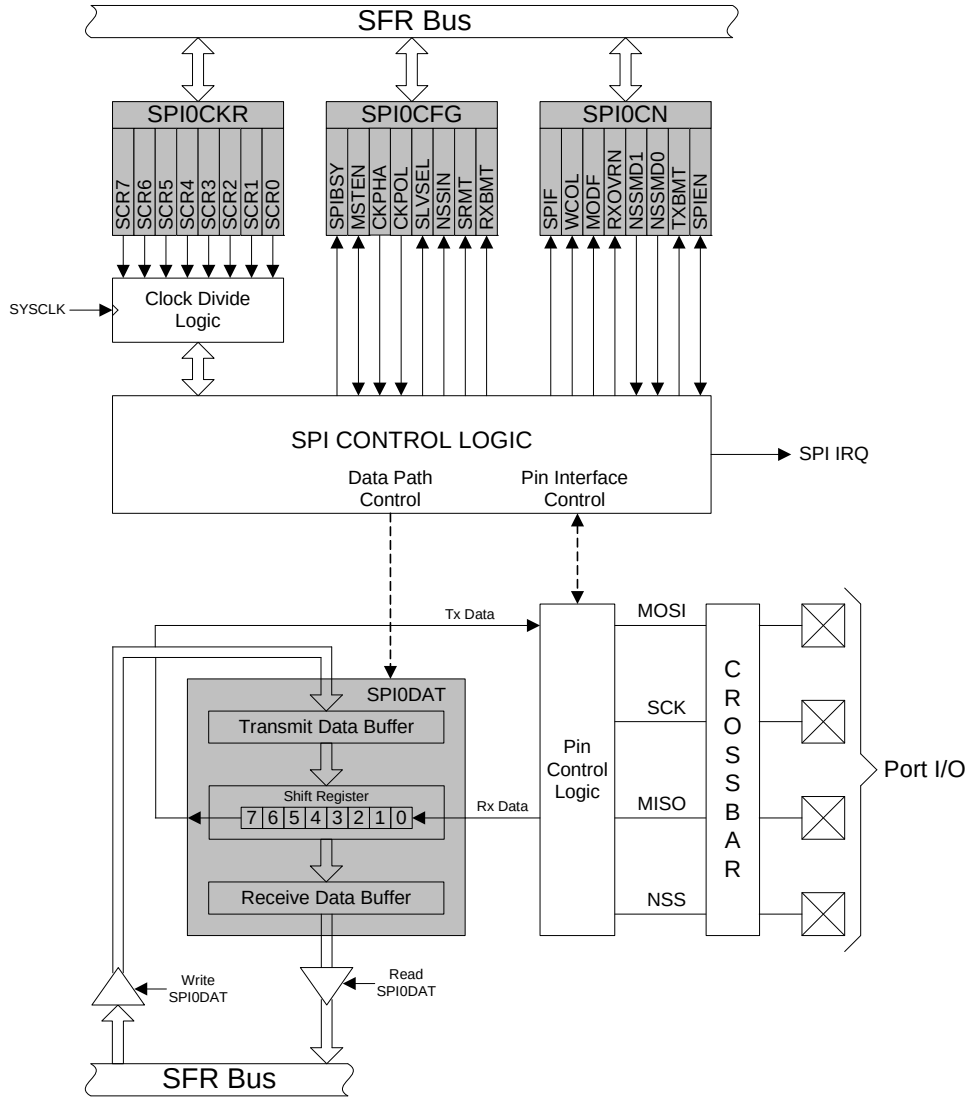


Figure 31.1. SPI Block Diagram

32.2.2. 8-bit Timers with Auto-Reload

When T2SPLIT is set, Timer 2 operates as two 8-bit timers (TMR2H and TMR2L). Both 8-bit timers operate in auto-reload mode as shown in Figure 32.5. TMR2RLL holds the reload value for TMR2L; TMR2RLH holds the reload value for TMR2H. The TR2 bit in TMR2CN handles the run control for TMR2H. TMR2L is always running when configured for 8-bit Mode.

Each 8-bit timer may be configured to use SYSCLK, SYSCLK divided by 12, SmaRTClock divided by 8 or Comparator 0 output. The Timer 2 Clock Select bits (T2MH and T2ML in CKCON) select either SYSCLK or the clock defined by the Timer 2 External Clock Select bits (T2XCLK[1:0] in TMR2CN), as follows:

T2MH	T2XCLK[1:0]	TMR2H Clock Source
0	00	SYSCLK / 12
0	01	SmaRTClock / 8
0	10	Reserved
0	11	Comparator 0
1	X	SYSCLK

T2ML	T2XCLK[1:0]	TMR2L Clock Source
0	00	SYSCLK / 12
0	01	SmaRTClock / 8
0	10	Reserved
0	11	Comparator 0
1	X	SYSCLK

The TF2H bit is set when TMR2H overflows from 0xFF to 0x00; the TF2L bit is set when TMR2L overflows from 0xFF to 0x00. When Timer 2 interrupts are enabled (IE.5), an interrupt is generated each time TMR2H overflows. If Timer 2 interrupts are enabled and TF2LEN (TMR2CN.5) is set, an interrupt is generated each time either TMR2L or TMR2H overflows. When TF2LEN is enabled, software must check the TF2H and TF2L flags to determine the source of the Timer 2 interrupt. The TF2H and TF2L interrupt flags are not cleared by hardware and must be manually cleared by software.

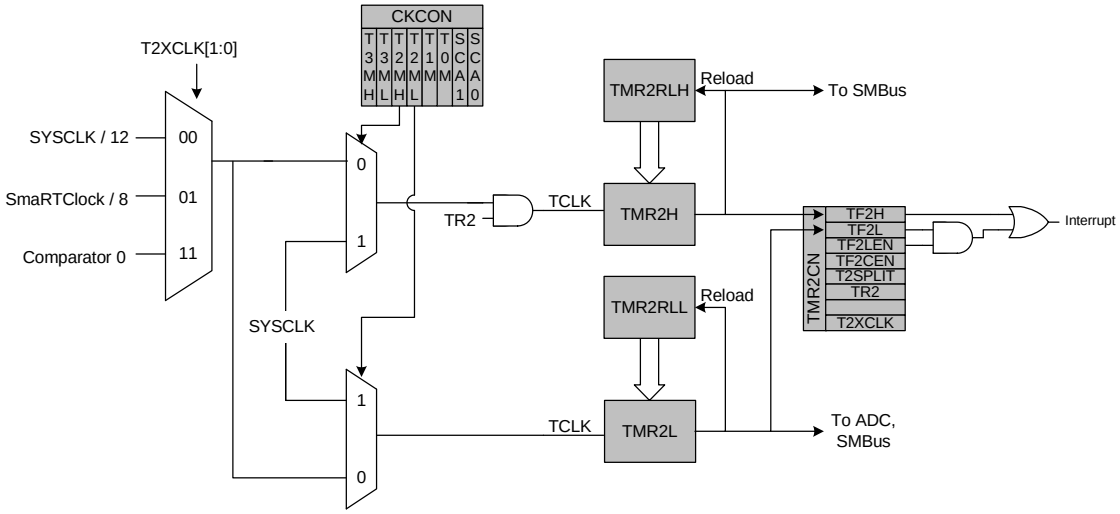


Figure 32.5. Timer 2 8-Bit Mode Block Diagram

32.2.3. Comparator 0/SmaRTClock Capture Mode

The Capture Mode in Timer 2 allows either Comparator 0 or the SmaRTClock period to be measured against the system clock or the system clock divided by 12. Comparator 0 and the SmaRTClock period can also be compared against each other. Timer 2 Capture Mode is enabled by setting TF2CEN to 1. Timer 2 should be in 16-bit auto-reload mode when using Capture Mode.

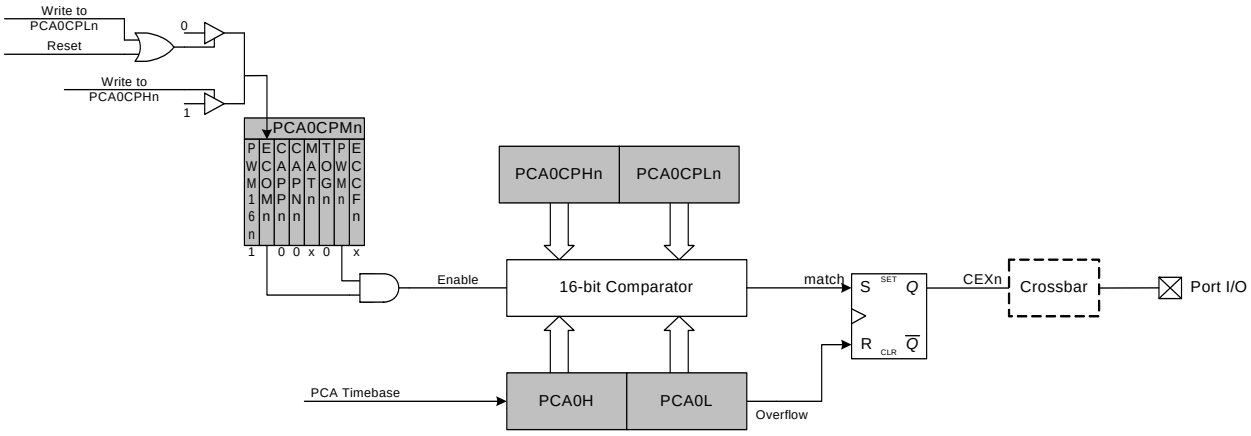


Figure 33.10. PCA 16-Bit PWM Mode

33.4. Watchdog Timer Mode

A programmable watchdog timer (WDT) function is available through the PCA Module 5. The WDT is used to generate a reset if the time between writes to the WDT update register (PCA0CPH2) exceed a specified limit. The WDT can be configured and enabled/disabled as needed by software.

With the WDTE bit set in the PCA0MD register, Module 5 operates as a watchdog timer (WDT). The Module 5 high byte is compared to the PCA counter high byte; the Module 5 low byte holds the offset to be used when WDT updates are performed. **The Watchdog Timer is enabled on reset. Writes to some PCA registers are restricted while the Watchdog Timer is enabled.** The WDT will generate a reset shortly after code begins execution. To avoid this reset, the WDT should be explicitly disabled (and optionally re-configured and re-enabled if it is used in the system).

33.4.1. Watchdog Timer Operation

While the WDT is enabled:

- PCA counter is forced on.
- Writes to PCA0L and PCA0H are not allowed.
- PCA clock source bits (CPS2–CPS0) are frozen.
- PCA Idle control bit (CIDL) is frozen.
- Module 5 is forced into software timer mode.
- Writes to the Module 5 mode register (PCA0CPM5) are disabled.

While the WDT is enabled, writes to the CR bit will not change the PCA counter state; the counter will run until the WDT is disabled. The PCA counter run control bit (CR) will read zero if the WDT is enabled but user software has not enabled the PCA counter. If a match occurs between PCA0CPH5 and PCA0H while the WDT is enabled, a reset will be generated. To prevent a WDT reset, the WDT may be updated with a write of any value to PCA0CPH5. Upon a PCA0CPH5 write, PCA0H plus the offset held in PCA0CPL5 is loaded into PCA0CPH5. (See Figure 33.11.)

33.5. Register Descriptions for PCA0

Following are detailed descriptions of the special function registers related to the operation of the PCA.

SFR Definition 33.1. PCA0CN: PCA Control

Bit	7	6	5	4	3	2	1	0
Name	CF	CR	CCF5	CCF4	CCF3	CCF2	CCF1	CCF0
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 Page = 0x0; SFR Address = 0xD8; Bit-Addressable

Bit	Name	Function
7	CF	**PCA Counter/Timer Overflow Flag.** Set by hardware when the PCA Counter/Timer overflows from 0xFFFF to 0x0000. When the Counter/Timer Overflow (CF) interrupt is enabled, setting this bit causes the CPU to vector to the PCA interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.
6	CR	**PCA Counter/Timer Run Control.** This bit enables/disables the PCA Counter/Timer. 0: PCA Counter/Timer disabled. 1: PCA Counter/Timer enabled.
5:0	CCF[5:0]	**PCA Module n Capture/Compare Flag.** These bits are set by hardware when a match or capture occurs in the associated PCA Module n. When the CCFn interrupt is enabled, setting this bit causes the CPU to vector to the PCA interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.