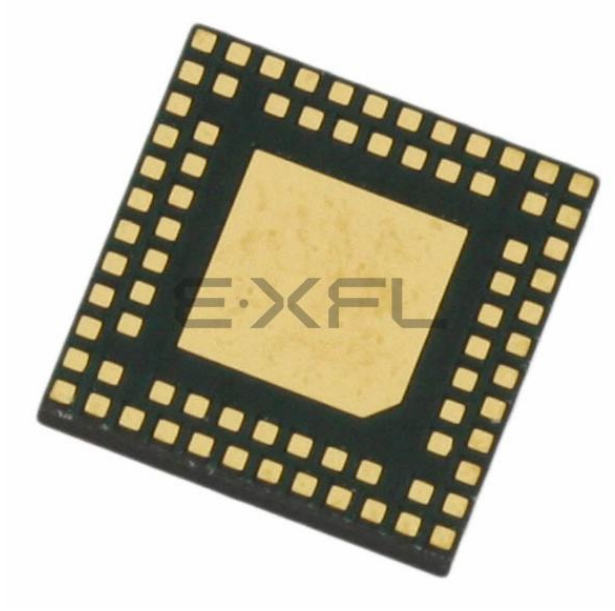

Welcome to E-XFL.COM

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	I²C, SPI, UART/USART
Peripherals	DMA, LCD, POR, PWM, WDT
Number of I/O	57
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8.25K 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/c8051f966-a-gmr

1.5. SAR ADC with 16-bit Auto-Averaging Accumulator and Autonomous Low Power Burst Mode

The ADC0 on C8051F96x devices is a 300 ksps, 10-bit or 75 ksps, 12-bit successive-approximation-register (SAR) ADC with integrated track-and-hold and programmable window detector. ADC0 also has an autonomous low power Burst Mode which can automatically enable ADC0, capture and accumulate samples, then place ADC0 in a low power shutdown mode without CPU intervention. It also has a 16-bit accumulator that can automatically oversample and average the ADC results. See Section “5.4. 12-Bit Mode” on page 84 for more details on using the ADC in 12-bit mode.

The ADC is fully configurable under software control via Special Function Registers. The ADC0 operates in single-ended mode and may be configured to measure various different signals using the analog multiplexer described in Section “5.7. ADC0 Analog Multiplexer” on page 95. The voltage reference for the ADC is selected as described in Section “5.9. Voltage and Ground Reference Options” on page 100.

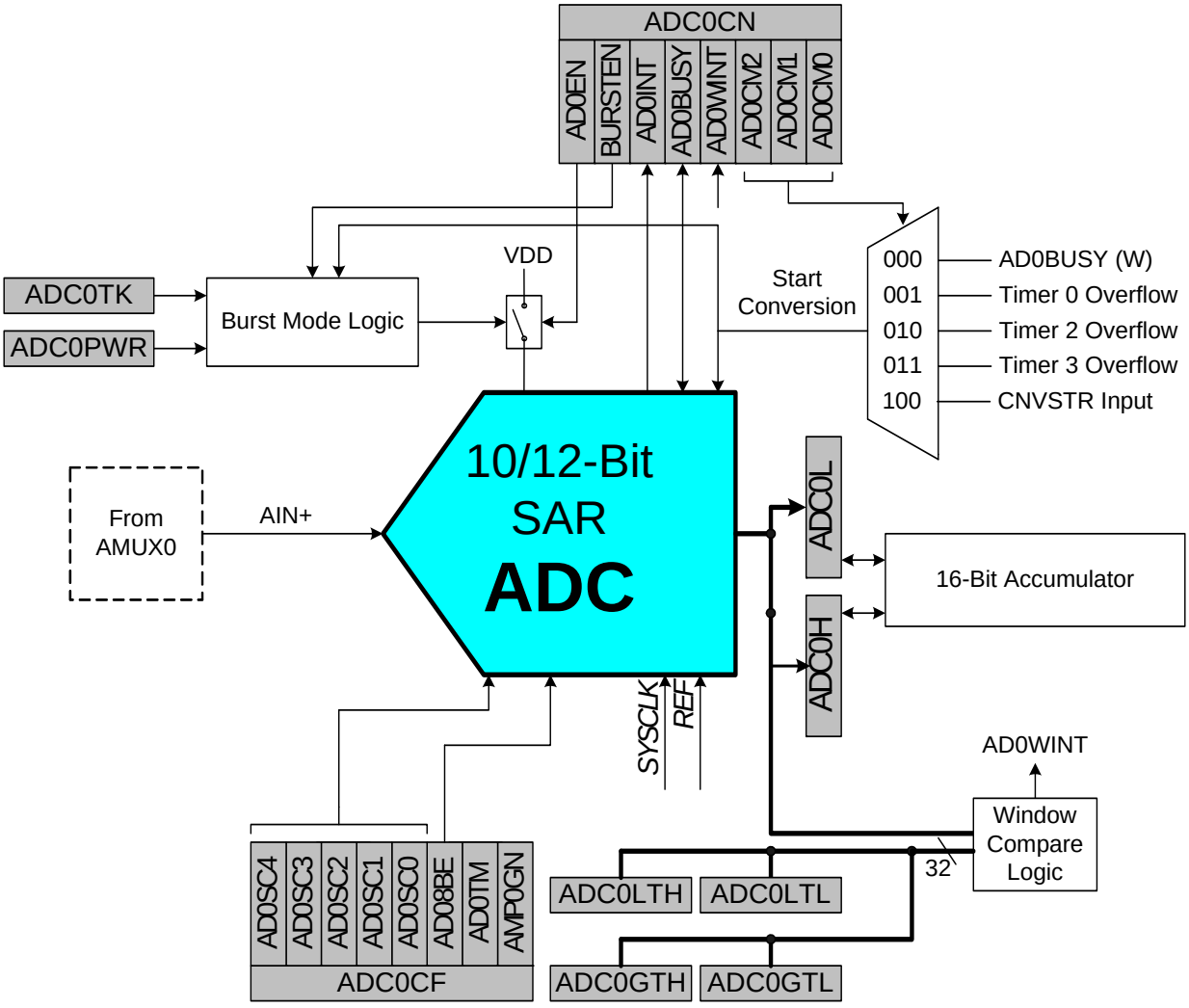

Figure 1.13. ADC0 Functional Block Diagram

2. Ordering Information

Table 2.1. Product Selection Guide

Ordering Part Number	MIPS (Peak)	Flash Memory (kB)	RAM (bytes)	Digital Port I/Os	AES 128, 192, 256 Encryption	LCD Segments (4-MUX)	SmaRTClock Real Time Clock	SMBus/I^2C	UART	Enhanced SPI	Timers (16-bit)	PCA Channels	10/12-bit 300/75 ksps ADC channels with internal VREF and temp sensor	Analog Comparators	Package
C8051F960-B-GM	25	128	8448	57	✓	128	✓	1	1	2	4	6	16	2	DQFN-76 (6x6)
C8051F960-B-GQ	25	128	8448	57	✓	128	✓	1	1	2	4	6	16	2	TQFP80 (12x12)
C8051F961-B-GM	25	128	8448	34	✓	36	✓	1	1	2	4	6	16	2	QFN-40 (6x6)
C8051F962-B-GM	25	128	8448	57	✓	—	✓	1	1	2	4	6	16	2	DQFN-76 (6x6)
C8051F962-B-GQ	25	128	8448	57	✓	—	✓	1	1	2	4	6	16	2	TQFP80 (12x12)
C8051F963-B-GM	25	128	8448	34	✓	—	✓	1	1	2	4	6	16	2	QFN-40 (6x6)
C8051F964-B-GM	25	64	8448	57	✓	128	✓	1	1	2	4	6	16	2	DQFN-76 (6x6)
C8051F964-B-GQ	25	64	8448	57	✓	128	✓	1	1	2	4	6	16	2	TQFP80 (12x12)
C8051F965-B-GM	25	64	8448	34	✓	36	✓	1	1	2	4	6	16	2	QFN-40 (6x6)
C8051F966-B-GM	25	32	8448	57	✓	128	✓	1	1	2	4	6	16	2	DQFN-76 (6x6)
C8051F966-B-GQ	25	32	8448	57	✓	128	✓	1	1	2	4	6	16	2	TQFP80 (12x12)
C8051F967-B-GM	25	32	8448	34	✓	36	✓	1	1	2	4	6	16	2	QFN-40 (6x6)
C8051F968-B-GM	25	16	4352	57	✓	128	✓	1	1	2	4	6	16	2	DQFN-76 (6x6)
C8051F968-B-GQ	25	16	4352	57	✓	128	✓	1	1	2	4	6	16	2	TQFP80 (12x12)
C8051F969-B-GM	25	16	4352	34	✓	36	✓	1	1	2	4	6	16	2	QFN-40 (6x6)

All packages are Lead-free (RoHS Compliant).

Rev A not recommended for new designs.

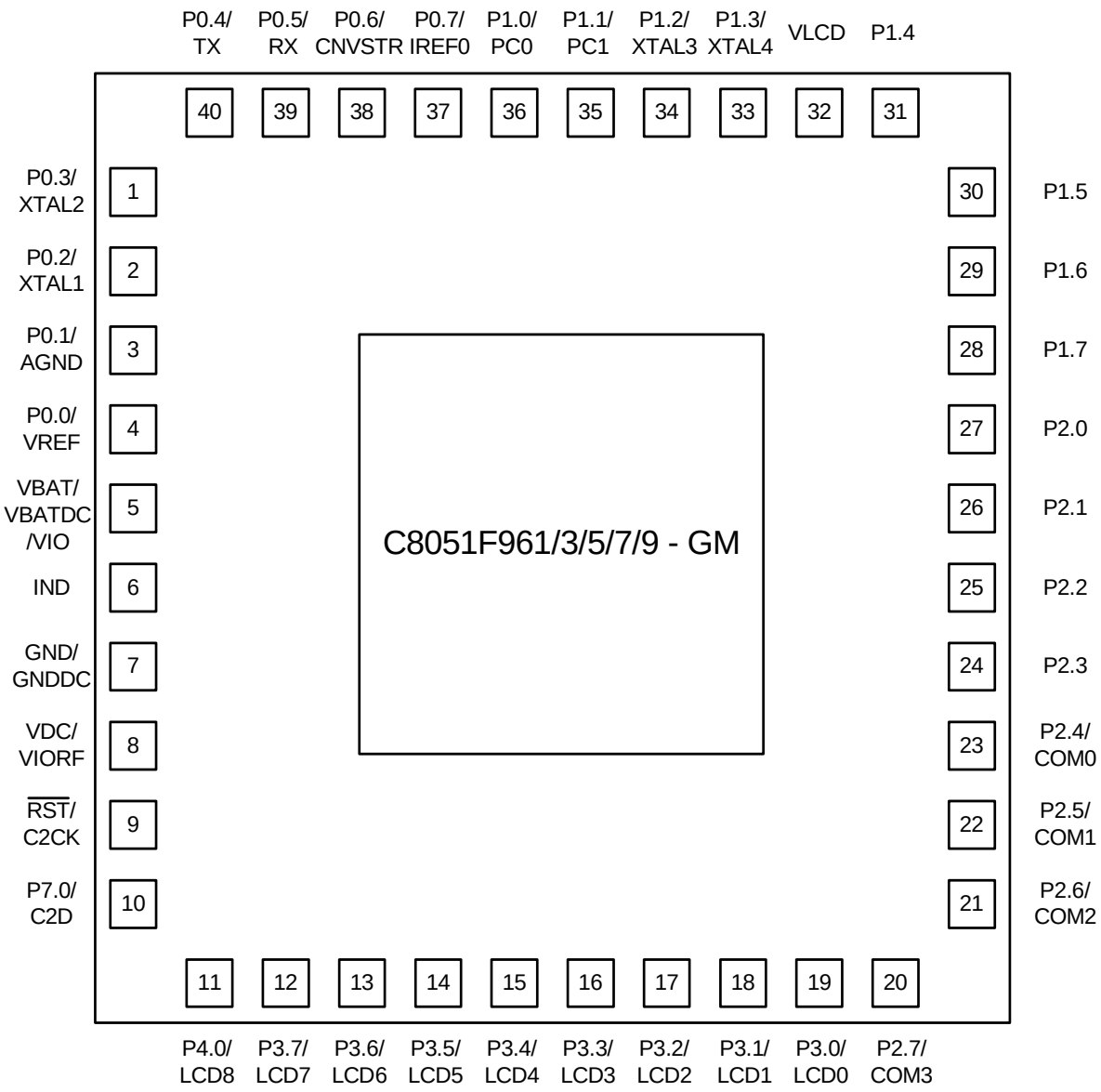

Figure 3.2. QFN-40 Pinout Diagram (Top View)

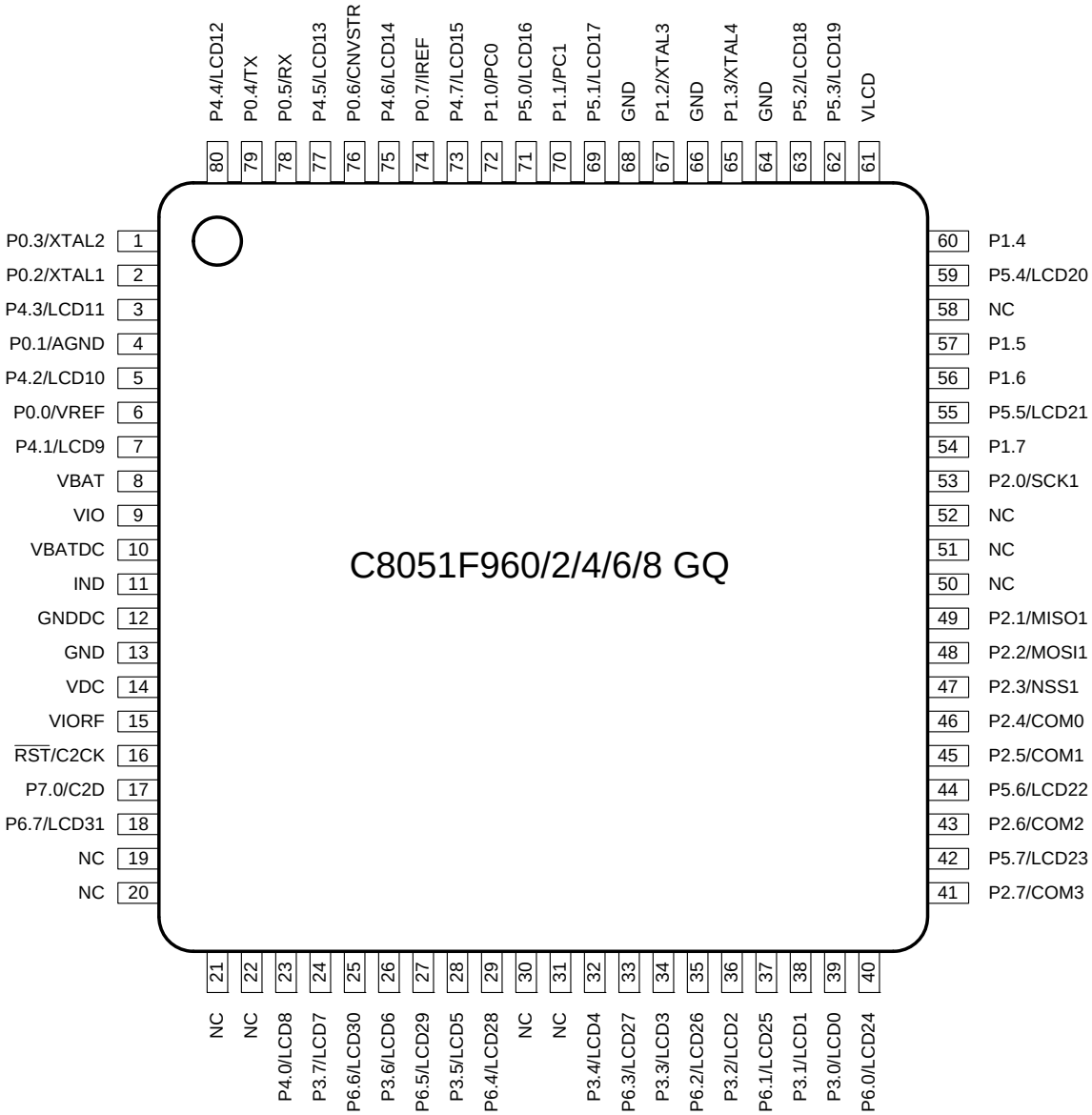

Figure 3.3. TQFP-80 Pinout Diagram (Top View)

Table 4.5. Port I/O DC Electrical Characteristics

V_{IO} = 1.8 to 3.8 V, –40 to +85 °C unless otherwise specified.

Parameters	Conditions	Min	Typ	Max	Units
Output High Voltage	High Drive Strength, PnDRV.n = 1				
	IOH = –3 mA, Port I/O push-pull	V_{IO}– 0.7	—	—	
	IOH = –10 µA, Port I/O push-pull	V_{IO} – 0.1	—	—	
	IOH = –10 mA, Port I/O push-pull		See Chart		
					V
	Low Drive Strength, PnDRV.n = 0				
	IOH = –1 mA, Port I/O push-pull	V_{IO} – 0.7	—	—	
	IOH = –10 µA, Port I/O push-pull	V_{IO} – 0.1	—	—	
	IOH = –3 mA, Port I/O push-pull	—	See Chart	—	
Output Low Voltage	High Drive Strength, PnDRV.n = 1				
	I_{OL} = 8.5 mA	—	—	0.6	
	I_{OL} = 10 µA	—	—	0.1	
	I_{OL} = 25 mA	—	See Chart	—	
					V
	Low Drive Strength, PnDRV.n = 0				
	I_{OL} = 1.4 mA	—	—	0.6	
	I_{OL} = 10 µA	—	—	0.1	
	I_{OL} = 4 mA	—	See Chart	—	
Input High Voltage	V_{BAT} = 2.0 to 3.8 V	V_{IO} – 0.6	—	—	V
	V_{BAT} = 1.8 to 2.0 V	0.7 x V_{IO}	—	—	V
Input Low Voltage	V_{BAT} = 2.0 to 3.8 V	—	—	0.6	V
	V_{BAT} = 1.8 to 2.0 V	—	—	0.3 x V_{IO}	V
Input Leakage Current	Weak Pullup Off				
	Weak Pullup On, V_{IN} = 0 V,	—	—	±1	
	V_{BAT} = 1.8 V	—	4	—	µA
	Weak Pullup On, Vin = 0 V, V_{BAT} = 3.8 V	—	20	35	

Table 4.14. Voltage Reference Electrical Characteristics

V_{BAT} = 1.8 to 3.8 V, –40 to +85 °C unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
Internal High-Speed Reference (REFSL[1:0] = 11)					
Output Voltage	–40 to +85 °C, V_{BAT} = 1.8–3.8 V	1.62	1.65	1.68	V
VREF Turn-on Time		—	—	1.5	µs
Supply Current	Normal Power Mode Low Power Mode	— —	260 140	— —	µA
External Reference (REFSL[1:0] = 00, REFOE = 0)					
Input Voltage Range		0	—	V_{BAT}	V
Input Current	Sample Rate = 300 ksps; VREF = 3.0 V	—	5.25	—	µA

SFR Definition 8.6. PSW: Program Status Word

Bit	7	6	5	4	3	2	1	0
Name	CY	AC	F0	RS[1:0]		OV	F1	PARITY
Type	R/W	R/W	R/W	R/W		R/W	R/W	R
Reset	0	0	0	0	0	0	0	0

SFR Page = All Pages; SFR Address = 0xD0; Bit-Addressable

Bit	Name	Function
7	CY	**Carry Flag.** This bit is set when the last arithmetic operation resulted in a carry (addition) or a borrow (subtraction). It is cleared to logic 0 by all other arithmetic operations.
6	AC	**Auxiliary Carry Flag.** This bit is set when the last arithmetic operation resulted in a carry into (addition) or a borrow from (subtraction) the high order nibble. It is cleared to logic 0 by all other arithmetic operations.
5	F0	**User Flag 0.** This is a bit-addressable, general purpose flag for use under software control.
4:3	RS[1:0]	**Register Bank Select.** These bits select which register bank is used during register accesses. 00: Bank 0, Addresses 0x00-0x07 01: Bank 1, Addresses 0x08-0x0F 10: Bank 2, Addresses 0x10-0x17 11: Bank 3, Addresses 0x18-0x1F
2	OV	**Overflow Flag.** This bit is set to 1 under the following circumstances: ■ An ADD, ADDC, or SUBB instruction causes a sign-change overflow. ■ A MUL instruction results in an overflow (result is greater than 255). ■ A DIV instruction causes a divide-by-zero condition. The OV bit is cleared to 0 by the ADD, ADDC, SUBB, MUL, and DIV instructions in all other cases.
1	F1	**User Flag 1.** This is a bit-addressable, general purpose flag for use under software control.
0	PARITY	**Parity Flag.** This bit is set to logic 1 if the sum of the eight bits in the accumulator is odd and cleared if the sum is even.

10.6. Timing

The timing parameters of the External Memory Interface can be configured to enable connection to devices having different setup and hold time requirements. The Address Setup time, Address Hold time, RD and WR strobe widths, and in multiplexed mode, the width of the ALE pulse are all programmable in units of SYSCLK periods through EMI0TC, shown in SFR Definition 10.3, and EMI0CF[1:0].

The timing for an off-chip MOVX instruction can be calculated by adding 4 SYSCLK cycles to the timing parameters defined by the EMI0TC register. Assuming non-multiplexed operation, the minimum execution time for an off-chip XRAM operation is 5 SYSCLK cycles (1 SYSCLK for RD or WR pulse + 4 SYSCLKs). For multiplexed operations, the Address Latch Enable signal will require a minimum of 2 additional SYSCLK cycles. Therefore, the minimum execution time for an off-chip XRAM operation in multiplexed mode is 7 SYSCLK cycles (2 for /ALE + 1 for RD or WR + 4). The programmable setup and hold times default to the maximum delay settings after a reset. Table 10.2 lists the ac parameters for the External Memory Interface, and Figure 10.4 through Figure 10.9 show the timing diagrams for the different External Memory Interface modes and MOVX operations.

10.6.1.2. 8-bit MOVX without Bank Select: EMI0CF[4:2] = 101 or 111

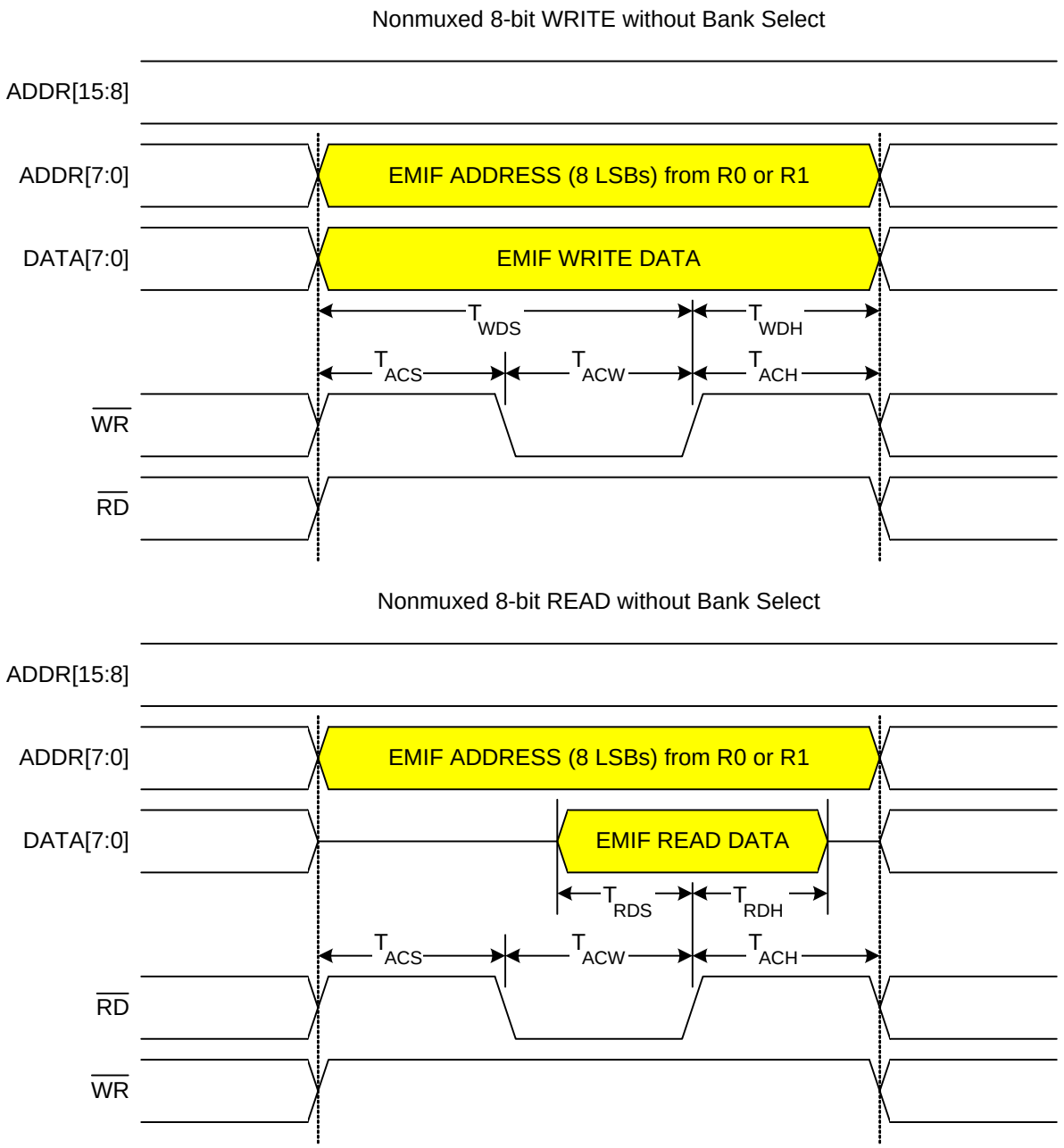

Figure 10.5. Non-multiplexed 8-bit MOVX without Bank Select Timing

14.6.1.1. CBC Encryption Data Flow

The AES0 module data flow for CBC encryption is shown in Figure 14.5. The plaintext is written to the AES0BIN sfr. For the first block, the initialization vector is written to the AES0XIN sfr. For subsequent blocks, the previous block ciphertext is written to the AES0XIN sfr. The AES0DCF sfr is configured to XOR AES0XIN with AES0BIN for the AES core data input. The XOR on the output is not used. The AES core is configured for an encryption operation. The encryption key is written to AES0KIN. The key size is set to the desired key size.

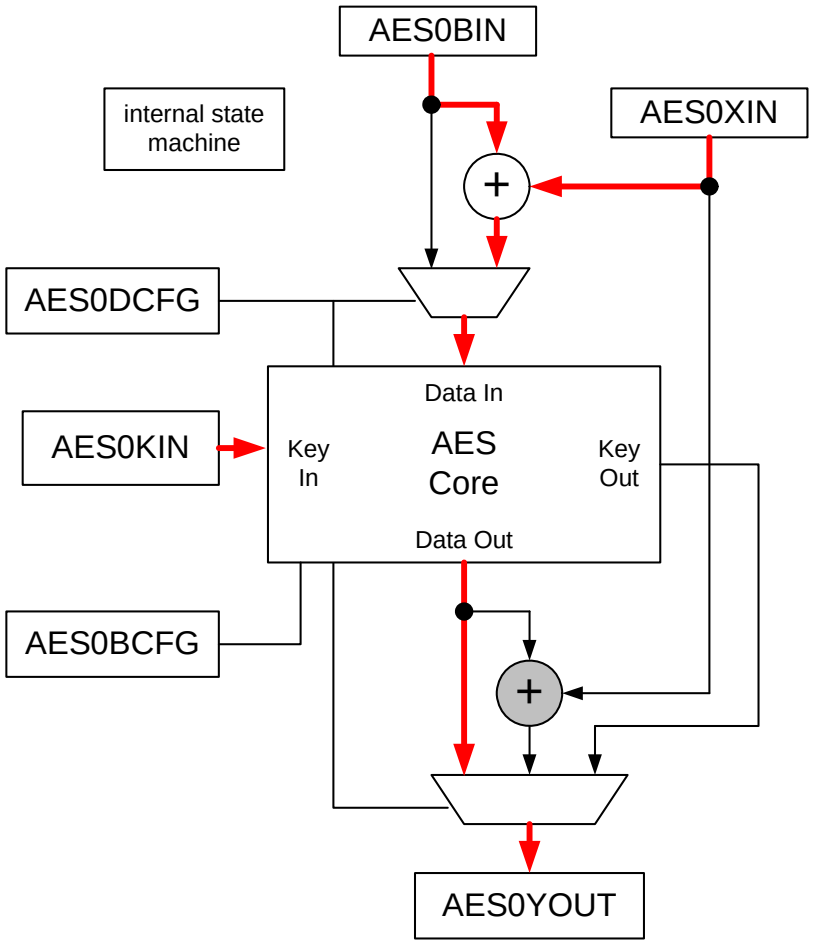

Figure 14.5. CBC Encryption Data Flow

14.6.3.1. CBC Encryption using SFRs

- First Configure AES Module for CBC Block Cipher Mode Encryption
  - Reset AES module by writing 0x00 to AES0BCFG.
  - Configure the AES Module data flow for XOR on input data by writing 0x01 to the AES0DCFG sfr.
  - Write key size to bits 1 and 0 of the AES0BCFG.
  - Configure the AES core for encryption by setting bit 2 of AES0BCFG.
  - Enable the AES core by setting bit 3 of AES0BCFG.
- Repeat alternating write sequence 16 times
  - Write plaintext byte to AES0BIN.
  - Write initialization vector to AES0XIN
  - Write encryption key byte to AES0KIN.
- Write remaining encryption key bytes to AES0KIN for 192-bit and 256-bit decryption only.
- Wait on AES done interrupt or poll bit 5 of AES0BCFG.
- Read 16 encrypted bytes from the AES0YOUT sfr.

If encrypting multiple blocks, this process may be repeated. It is not necessary reconfigure the AES module for each block. When using Cipher Block Chaining the initialization vector is written to the AES0XIN sfr for the first block only, as described. Additional blocks will chain the encrypted data from the previous block.

- Disable the DMA by writing 0x00 to DMA0EN
- Increment counter and repeat all steps for additional blocks

14.6.6.1. CTR Encryption using SFRs

- First Configure AES Module for CTR Block Cipher Mode Encryption
  - Reset AES module by writing 0x00 to AES0BCFG.
  - Configure the AES Module data flow for XOR on output data by writing 0x02 to the AES0DCFG sfr.
  - Write key size to bits 1 and 0 of the AES0BCFG.
  - Configure the AES core for encryption by setting bit 2 of AES0BCFG.
  - Enable the AES core by setting bit 3 of AES0BCFG.
- Repeat alternating write sequence 16 times
  - Write plaintext byte to AES0BIN.
  - Write counter byte to AES0XIN
  - Write encryption key byte to AES0KIN.
- Write remaining encryption key bytes to AES0KIN for 192-bit and 256-bit decryption only.
- Wait on AES done interrupt or poll bit 5 of AES0BCFG.
- Read 16 encrypted bytes from the AES0YOUT sfr.

If encrypting multiple blocks, increment the counter and repeat this process. It is not necessary reconfigure the AES module for each block.

SFR Definition 14.4. AES0XIN: AES XOR Input

Bit	7	6	5	4	3	2	1	0
Name	AES0XIN[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 Address = 0xEC; SFR page = 0x2; Not bit-Addressable

Bit	Name	Function
7:0	AES0XIN[7:0]	**AES XOR Input.** The AES0XIN may be used in conjunction with the AES0BIN sfr for some cipher block modes. When used with the DMA, the DMA will write directly to this sfr. When used without the DMA - AES0BIN, AES0XIN, and AES0KIN must be written in sequence. Reading this register will yield the last value written. This can be used for debug purposes.

SFR Definition 14.5. AES0KIN: AES Key Input

Bit	7	6	5	4	3	2	1	0
Name	AES0KIN[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 Address = 0xED; SFR page = 0x2; Not bit-Addressable

Bit	Name	Function
7:0	AES0KIN[7:0]	**AES Key Input.** During an encryption operation, the plaintext is written to the AES0BIN sfr. During an decryption operation, the ciphertext is written to the AES0BIN sfr. During a key inversion the encryption key is written to AES0BIN. When used with the DMA, the DMA will write directly to this sfr. The AES0BIN may be used in conjunction with the AES0XIN sfr for some cipher block modes. When used without the DMA - AES0BIN, AES0XIN, and AES0KIN must be written in sequence.

18.5. Flash Write and Erase Guidelines

Any system which contains routines which write or erase flash memory from software involves some risk that the write or erase routines will execute unintentionally if the CPU is operating outside its specified operating range of VDD, system clock frequency, or temperature. This accidental execution of flash modifying code can result in alteration of flash memory contents causing a system failure that is only recoverable by re-Flashing the code in the device.

To help prevent the accidental modification of flash by firmware, the VDD Monitor must be enabled and enabled as a reset source on C8051F96x devices for the flash to be successfully modified. **If either the VDD Monitor or the VDD Monitor reset source is not enabled, a Flash Error Device Reset will be generated when the firmware attempts to modify the flash.**

The following guidelines are recommended for any system that contains routines which write or erase flash from code.

18.5.1. VDD Maintenance and the VDD Monitor

1. If the system power supply is subject to voltage or current "spikes," add sufficient transient protection devices to the power supply to ensure that the supply voltages listed in the Absolute Maximum Ratings table are not exceeded.
2. Make certain that the minimum VDD rise time specification of 1 ms is met. If the system cannot meet this rise time specification, then add an external VDD brownout circuit to the $\overline{\text{RST}}$ pin of the device that holds the device in reset until VDD reaches the minimum device operating voltage and re-asserts $\overline{\text{RST}}$ if VDD drops below the minimum device operating voltage.
3. Keep the on-chip VDD Monitor enabled and enable the VDD Monitor as a reset source as early in code as possible. This should be the first set of instructions executed after the Reset Vector. For C-based systems, this will involve modifying the startup code added by the 'C' compiler. See your compiler documentation for more details. Make certain that there are no delays in software between enabling the VDD Monitor and enabling the VDD Monitor as a reset source. Code examples showing this can be found in “AN201: Writing to Flash from Firmware," available from the Silicon Laboratories web site.

Notes:
On C8051F96x devices, both the VDD Monitor and the VDD Monitor reset source must be enabled to write or erase flash without generating a Flash Error Device Reset.

On C8051F96x devices, both the VDD Monitor and the VDD Monitor reset source are enabled by hardware after a power-on reset.

4. As an added precaution, explicitly enable the VDD Monitor and enable the VDD Monitor as a reset source inside the functions that write and erase flash memory. The VDD Monitor enable instructions should be placed just after the instruction to set PSWE to a 1, but before the flash write or erase operation instruction.
5. Make certain that all writes to the RSTSRC (Reset Sources) register use direct assignment operators and explicitly DO NOT use the bit-wise operators (such as AND or OR). For example, "RSTSRC = 0x02" is correct, but "RSTSRC |= 0x02" is incorrect.
6. Make certain that all writes to the RSTSRC register explicitly set the PORSF bit to a '1'. Areas to check are initialization code which enables other reset sources, such as the Missing Clock Detector or Comparator, for example, and instructions which force a Software Reset. A global search on "RSTSRC" can quickly verify this.

SFR Definition 20.2. DC0CF: DC-DC Converter Configuration

Bit	7	6	5	4	3	2	1	0
Name	BYPASS	VSEL[3:0]				OSCDIS	SWSEL[1:0]	
Type	R	R/W				R/W		
Reset	0	0	0	0	1	0	1	1

SFR Page = 0x0; SFR Address = 0x96

Bit	Name	Function
7	BYPASS	**DC-DC Converter Bypass Switch Active Indicator.** 0: The bypass switch is open. 1: The bypass switch is closed (VDC is connected to VBATDC).
6:3	VSEL[3:0]	**DC-DC Converter Output Voltage Select.** Specifies the target output voltage. 0000: Target output voltage is 1.8 V. 1000: Target output voltage is 2.6 V. 0001: Target output voltage is 1.9 V. 1001: Target output voltage is 2.7 V. 0010: Target output voltage is 2.0 V. 1010: Target output voltage is 2.8 V. 0011: Target output voltage is 2.1 V. 1011: Target output voltage is 2.9 V. 0100: Target output voltage is 2.2 V. 1100: Target output voltage is 3.0 V. 0101: Target output voltage is 2.3 V. 1101: Target output voltage is 3.1 V. 0110: Target output voltage is 2.4 V. 1110: Target output voltage is 3.3 V. 0111: Target output voltage is 2.5 V. 1111: Target output voltage is 3.5 V.
2	VSEL[2:0]	**DC-DC Converter Local Oscillator Disabled.** 0: The local oscillator inside the dc-dc converter is enabled. 1: The local oscillator inside the dc-dc converter is disabled.
1:0	SWSEL[1:0]	**DC-DC Converter Power Switch Select.** Selects the size of the power switches (M1, M2). Using smaller switches will resut in higher efficiency at low supply currents. 00: Minimum switch size, optimized for load currents smaller than 5 mA. 01: Reserved. 10: Reserved. 11: Maximum switch size, optimized for load currents greater than 5 mA.

To enable the Pulse Counter as a wake up source, enable the source in the PC0INT0/1 SFRs and enable the Pulse Counter as a wake-up source by setting bit 0 (PC0WK) to 1 in the PMU0FL SFR. Upon waking, firmware should read the PMCU0CF and PMU0FL SFRs to determine the wake-up source. If the PC0WK bit is set indicating that the Pulse Counter has woke the MCU, firmware should read the flag bits PC0INT0/1 SFRs to determine the Pulse Counter wake-up source and clear the flag bits before going back to sleep.

PC0INT0 includes the more common interrupt and wake-up sources. These include comparator match, counter overflow, and quadrature direction change. PC0INT1 includes interrupt and wake-up sources for the advanced features, including flutter detection and quadrature error.

25.9. Real-Time Register Access

Several of the Pulse Counter registers values change in real-time synchronous to the RTC clock. Hardware synchronization between the RTC clock domain and the system clock domain hardware would result in long delays when reading real-time registers. Instead, real-time register values are available instantaneously, but the read must be qualified using the read valid bit (PC0TH bit 0). If the register value does not change during the read access, the read valid bit will be set indicating the last was valid. If the value of the real-time register changes during the read access, the read valid bit is 0, indicating the read was invalid. After an invalid read, firmware must read the register and check the read valid bit again.

These 8-bit counter registers need to be qualified using the read valid bit:

- PC0STAT
- PC0HIST
- PC0INT0
- PC0INT1
- PC0CTR0L
- PC0CTC1L

The 24-bit counters are three-byte real-time read-only registers that require a special access method for reading. Firmware must read the low-byte (PC0CTR0L and PC0CTR1L) first and qualify using the read valid bit. Reading the low-byte latches the middle and high bytes. If the read valid bit is 0, the read is invalid and firmware must read the low-byte and check the read valid bit again. If the read valid bit is set, the read is valid and the middle and high bytes are also safe to read. Firmware should read the middle and high bytes only after reading the low byte and qualifying with the read valid bit.

The 24-bit compators are three-byte real-time read-write registers that require a special access method for writing. Firmware must write the low-byte last. After writing the low-byte, it might take up to two RTC clock cycles for the new comparator value to take effect. System designers should consider the synchronization delay when setting the comparator value. The counter may be incremented before new comparator value takes effect. Setting the comparator to at least 2 counts above the current count will eliminate the chance of missing the comparator match during synchronization.

Example code is provided with accessor functions for all the real-time Pulse Counter registers.

25.10. Advanced Features

25.10.1. Quadrature Error

The quadrature encoder must only send valid quadrature codes. A valid quadrature sequence consists of four valid states. The quadrature codes are only permitted to transition to one of the adjacent states, and an invalid transition will result in a quadrature error. Note that a quadrature error is likely to occur when first enabling the quadrature counter mode, since the Pulse Counter state machine starts at the LL state and the initial state of the quadrature is arbitrary. It is safe to ignore the first quadrature error immediately after initialization.

SFR Definition 25.4. PC0STAT: PC0 Status

Bit	7	6	5	4	3	2	1	0
Name	FLUTTER	DIRECTION	STATE[1:0]		PC1PREV	PC0PREV	PC1	PC0
Type	RO	RO	RO		RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xC1; SFR Page = 0x2

Bit	Name	Function
7	FLUTTER	**Flutter** During quadrature mode, a disparity may occur between the number of negative edges of PC1 and PC0 or the number of positive edges of PC1 and PC0. This could indicate flutter on one reed switch or one reed switch may be faulty. 0: No flutter detected. 1: Flutter detected.
6	DIRECTION	**Direction** Only applicable for quadrature mode. (First letter is PC1; second letter is PC0) 0: Counter clock-wise - (LL-LH-HH-HL) 1: Clock-wise - (LL-HL-HH-LH)
5:4	STATE[1:0]	**PC0 State** Current State of Internal State Machine.
3	PC1PREV	**PC1 Previous** Previous Output of PC1 Integrator.
2	PC0PREV	**PC0 Previous** Previous Output of PC0 Integrator.
1	PC1	**PC1** Current Output of PC1 Integrator.
0	PC0	**PC0** Current Output of PC0 Integrator.

To initiate a Master mode Bidirectional data transfer:

1. Configure the SPI1 SFRs normally for Master mode.
   a. Enable Master mode by setting bit 6 in SPI1CFG.
   b. Configure the clock polarity CKPOL and clock phase CKPHA as desired in SPI1CFG.
   c. Configure SPI1CKR for the desired SPI clock rate.
   d. Configure the desired 4-wire master or 3-wire master mode in SPI1CN.
   e. Enable the SPI by setting bit 0 of SPI1CN.
2. Configure the first DMA channel for the XRAM-to-SPI1DATA transfer:
   a. Disable the first DMA channel by clearing the corresponding bit in DMA0EN.
   b. Select the first DMA channel by writing to DMA0SEL.
   c. Configure the selected DMA channel to use the XRAM-to-SPI1DAT peripheral request by writing 0x03 to DMA0NCF.
   d. Write 0 to DMA0NMD to disable wrapping.
   e. Write the address of the first byte of master output (MOSI) data to DMA0NBAH:L.
   f. Write the size of the SPI transfer in bytes to DMA0NSZH:L.
   g. Clear the address offset SFRs CMA0A0H:L.
3. Configure the second DMA channel for the SPI1DAT-to-XRAM transfer:
   a. Disable the second DMA channel by clearing the corresponding bit in DMA0EN.
   b. Select the second DMA channel by writing to DMA0SEL.
   c. Configure the selected DMA channel to use the SPI1DAT-to-XRAM peripheral request by writing 0x04 to DMA0NCF.
   d. Enable DMA interrupts for the second channel by setting bit 7 of DMA0NCF.
   e. Write 0 to DMA0NMD to disable wrapping.
   f. Write the address for the first byte of master input (MISO) data to DMA0NBAH:L.
   g. Write the size of the SPI transfer in bytes to DMA0NSZH:L.
   h. Clear the address offset SFRs CMA0A0H:L.
   i. Enable the interrupt on the second channel by setting the corresponding bit in DMA0INT.
   j. Enable DMA interrupts by setting bit 5 of EIE2.
4. Clear the interrupt bits in DMA0INT for both channels.
5. Enable both channels by setting the corresponding bits in the DMA0EN SFR to initiate the SPI transfer operation.
6. Wait on the DMA interrupt.
7. Clear the DMA enables in the DMA0EN SFR.
8. Clear the DMA interrupts in the DMA0INT SFR.

31.9. Master Mode Unidirectional Data Transfer

A unidirectional SPI master mode DMA transfer will transfer a specified number of bytes out on the MOSI pin. The MOSI data must be stored in XRAM before initiating the DMA transfers. The SPI1DAT-to-XRAM peripheral request is not used. Since the DMA does not read the SPI1DAT SFR, the SPI will discard the MISO data.

A unidirectional transfer only requires one DMA channel to transfer XRAM data to the SPI1DAT SFR. The DMA interrupt will indicate the completion of the data transfer to the SPI1DAT SFR. When the interrupt occurs, the DMA has written all of the data to the SPI1DAT SFR, but the SPI has not transmitted the last byte. Firmware may poll on the SPIBSY bit to determine when the SPI has transmitted the last byte. Firmware should not deassert the NSS pin until after the SPI has transmitted the last byte.

To initiate a master mode unidirectional data transfer:

1. Configure the SPI1 SFRs normally for Master mode.
   a. Enable Master mode by setting bit 6 in SPI1CFG.
   b. Configure the clock polarity CKPOL and clock phase CKPHA as desired in SPI1CFG.
   c. Configure SPI1CKR for the desired SPI clock rate.
   d. Configure the desired 4-wire master or 3-wire master mode in SPI1CN.
   e. Enable the SPI by setting bit 0 of SPI1CN.
2. Configure the desired DMA channel for the XRAM-to-SPI1DAT transfer.
   a. Disable the desired DMA channel by clearing the corresponding bit in DMA0EN.
   b. Select the desired DMA channel by writing to DMA0SEL.
   c. Configure the selected DMA channel to use the XRAM-to-SPI1DAT XRAM peripheral request by writing 0x03 to DMA0NCF.
   d. Enable DMA interrupts for the desired channel by setting bit 7 of DMA0NCF.
   e. Write 0 to DMA0NMD to disable wrapping.
   f. Write the address for the first byte of master output (MOSI) data to DMA0NBAH:L.
   g. Write the size of the SPI transfer in bytes to DMA0NSZH:L.
   h. Clear the address offset SFRs CMA0A0H:L.
   i. Enable the interrupt on the desired channel by setting the corresponding bit in DMA0INT.
   j. Enable DMA interrupts by setting bit 5 of EIE2.
3. Clear the interrupt bit in DMA0INT for the desired channel.
4. Enable the desired channel by setting the corresponding bit in the DMA0EN SFR to initiate the SPI transfer operation.
5. Wait on the DMA interrupt.
6. Clear the DMA enables in the DMA0EN SFR.
7. Clear the DMA interrupts in the DMA0INT SFR.
8. If desired, wait on the SPIBSY bit in SPI1CFG for the last byte transfer to complete.

SFR Definition 33.7. PCA0CPLn: PCA Capture Module Low Byte

Bit	7	6	5	4	3	2	1	0
Name	PCA0CPn[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 Addresses: PCA0CPL0 = 0xFB, PCA0CPL1 = 0xE9, PCA0CPL2 = 0xEB,
PCA0CPL3 = 0xED, PCA0CPL4 = 0xFD, PCA0CPL5 = 0xD2

SFR Pages: PCA0CPL0 = 0x0, PCA0CPL1 = 0x0, PCA0CPL2 = 0x0,
PCA0CPL3 = 0x0, PCA0CPL4 = 0x0, PCA0CPL5 = 0x0

Bit	Name	Function
7:0	PCA0CPn[7:0]	**PCA Capture Module Low Byte.** The PCA0CPLn register holds the low byte (LSB) of the 16-bit capture module n. This register address also allows access to the low byte of the corresponding PCA channel's auto-reload value for 9, 10, or 11-bit PWM mode. The ARSEL bit in register PCA0PWM controls which register is accessed.
Note: A write to this register will clear the module's ECOMn bit to a 0.		

SFR Definition 33.8. PCA0CPHn: PCA Capture Module High Byte

Bit	7	6	5	4	3	2	1	0
Name	PCA0CPn[15:8]							
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 Addresses: PCA0CPH0 = 0xFC, PCA0CPH1 = 0xEA, PCA0CPH2 = 0xEC,
PCA0CPH3 = 0xEE, PCA0CPH4 = 0xFE, PCA0CPH5 = 0xD3

SFR Pages: PCA0CPH0 = 0x0, PCA0CPH1 = 0x0, PCA0CPH2 = 0x0,
PCA0CPH3 = 0x0, PCA0CPH4 = 0x0, PCA0CPH5 = 0x0

Bit	Name	Function
7:0	PCA0CPn[15:8]	**PCA Capture Module High Byte.** The PCA0CPHn register holds the high byte (MSB) of the 16-bit capture module n. This register address also allows access to the high byte of the corresponding PCA channel's auto-reload value for 9, 10, or 11-bit PWM mode. The ARSEL bit in register PCA0PWM controls which register is accessed.
Note: A write to this register will set the module's ECOMn bit to a 1.		