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	48 MIPS
Connectivity	EBI/EMI, I²C, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	40
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 ~ 5.25V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-TQFP
Supplier Device Package	48-TQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f384-gqr

2. C8051F34x Compatibility

The C8051F38x family is designed to be a pin and code compatible replacement for the C8051F34x device family, with an enhanced feature set. The C8051F38x device should function as a drop-in replacement for the C8051F34x devices in most applications. Table 2.1 lists recommended replacement part numbers for C8051F34x devices. See “2.1. Hardware Incompatibilities” to determine if any changes are necessary when upgrading an existing C8051F34x design to the C8051F38x.

Table 2.1. C8051F38x Replacement Part Numbers

C8051F34x Part Number	C8051F38x Part Number
C8051F340-GQ	C8051F380-GQ
C8051F341-GQ	C8051F382-GQ
C8051F342-GQ	C8051F381-GQ
C8051F342-GM	C8051F381-GM
C8051F343-GQ	C8051F383-GQ
C8051F343-GM	C8051F383-GM
C8051F344-GQ	C8051F380-GQ
C8051F345-GQ	C8051F382-GQ
C8051F346-GQ	C8051F381-GQ
C8051F346-GM	C8051F381-GM
C8051F347-GQ	C8051F383-GQ
C8051F347-GM	C8051F383-GM
C8051F348-GQ	C8051F386-GQ
C8051F349-GQ	C8051F387-GQ
C8051F349-GM	C8051F387-GM
C8051F34A-GQ	C8051F381-GQ
C8051F34A-GM	C8051F381-GM
C8051F34B-GQ	C8051F383-GQ
C8051F34B-GM	C8051F383-GM
C8051F34C-GQ	C8051F384-GQ
C8051F34D-GQ	C8051F385-GQ

3. Pinout and Package Definitions

Table 3.1. Pin Definitions for the C8051F380/1/2/3/4/5/6/7/C

Name	Pin Numbers		Type	Description
	48-pin	32-pin		
V_{DD}	10	6	Power In Power Out	2.7–3.6 V Power Supply Voltage Input. 3.3 V Voltage Regulator Output.
GND	7	3		Ground.
$\overline{RST}$/ C2CK	13	9	D I/O D I/O	Device Reset. Open-drain output of internal POR or V_{DD} monitor. An external source can initiate a system reset by driving this pin low for at least 15 µs. Clock signal for the C2 Debug Interface.
C2D	14	—	D I/O	Bi-directional data signal for the C2 Debug Interface.
P3.0 / C2D	—	10	D I/O D I/O	Port 3.0. See Section 20 for a complete description of Port 3. Bi-directional data signal for the C2 Debug Interface.
REGIN	11	7	Power In	5 V Regulator Input. This pin is the input to the on-chip voltage regulator.
VBUS	12	8	D In	VBUS Sense Input. This pin should be connected to the VBUS signal of a USB network. A 5 V signal on this pin indicates a USB network connection.
D+	8	4	D I/O	USB D+.
D–	9	5	D I/O	USB D–.
P0.0	6	2	D I/O or A In	Port 0.0. See Section 20 for a complete description of Port 0.
P0.1	5	1	D I/O or A In	Port 0.1.
P0.2	4	32	D I/O or A In	Port 0.2.
P0.3	3	31	D I/O or A In	Port 0.3.
P0.4	2	30	D I/O or A In	Port 0.4.
P0.5	1	29	D I/O or A In	Port 0.5.
P0.6	48	28	D I/O or A In	Port 0.6.

14.7.2.3. 8-bit MOVX with Bank Select: EMI0CF[4:2] = 010

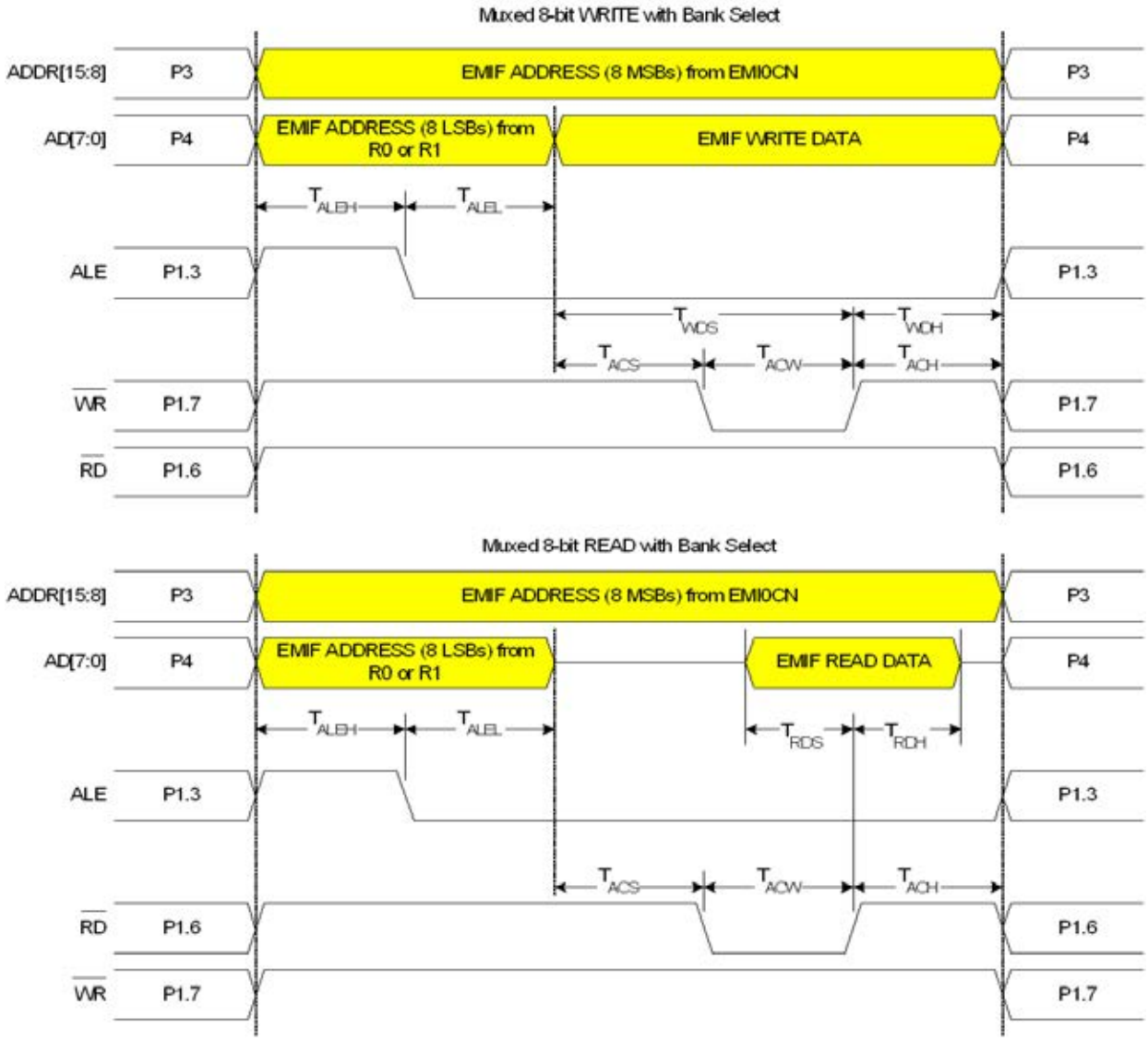


Figure 14.10. Multiplexed 8-bit MOVX with Bank Select Timing

SFR Definition 16.2. IP: Interrupt Priority

Bit	7	6	5	4	3	2	1	0
Name		PSPI0	PT2	PS0	PT1	PX1	PT0	PX0
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	0	0	0	0	0	0

SFR Address = 0xB8; SFR Page = All Pages; Bit-Addressable

Bit	Name	Function
7	Unused	Read = 1b, Write = Don't Care.
6	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.
5	PT2	**Timer 2 Interrupt Priority Control.** This bit sets the priority of the Timer 2 interrupt. 0: Timer 2 interrupt set to low priority level. 1: Timer 2 interrupt set to high priority level.
4	PS0	**UART0 Interrupt Priority Control.** This bit sets the priority of the UART0 interrupt. 0: UART0 interrupt set to low priority level. 1: UART0 interrupt set to high priority level.
3	PT1	**Timer 1 Interrupt Priority Control.** This bit sets the priority of the Timer 1 interrupt. 0: Timer 1 interrupt set to low priority level. 1: Timer 1 interrupt set to high priority level.
2	PX1	**External Interrupt 1 Priority Control.** This bit sets the priority of the External Interrupt 1 interrupt. 0: External Interrupt 1 set to low priority level. 1: External Interrupt 1 set to high priority level.
1	PT0	**Timer 0 Interrupt Priority Control.** This bit sets the priority of the Timer 0 interrupt. 0: Timer 0 interrupt set to low priority level. 1: Timer 0 interrupt set to high priority level.
0	PX0	**External Interrupt 0 Priority Control.** This bit sets the priority of the External Interrupt 0 interrupt. 0: External Interrupt 0 set to low priority level. 1: External Interrupt 0 set to high priority level.

SFR Definition 18.3. FLSCL: Flash Scale

Bit	7	6	5	4	3	2	1	0
Name	FOSE	Reserved		FLRT	Reserved			
Type	R/W	R/W		R/W	R/W			
Reset	1	0	0	0	0	0	0	0

SFR Address = 0xB6; SFR Page = All Pages

Bit	Name	Function
7	FOSE	**Flash One-shot Enable.** This bit enables the Flash read one-shot. When the Flash one-shot disabled, the Flash sense amps are enabled for a full clock cycle during Flash reads. At system clock frequencies below 10 MHz, disabling the Flash one-shot will increase system power consumption. 0: Flash one-shot disabled. 1: Flash one-shot enabled.
6:5	Reserved	Must write 00b.
4	FLRT	**FLASH Read Time.** This bit should be programmed to the smallest allowed value, according to the system clock speed. 0: SYSCLK <= 25 MHz. 1: SYSCLK <= 48 MHz.
3:0	Reserved	Must write 0000b.

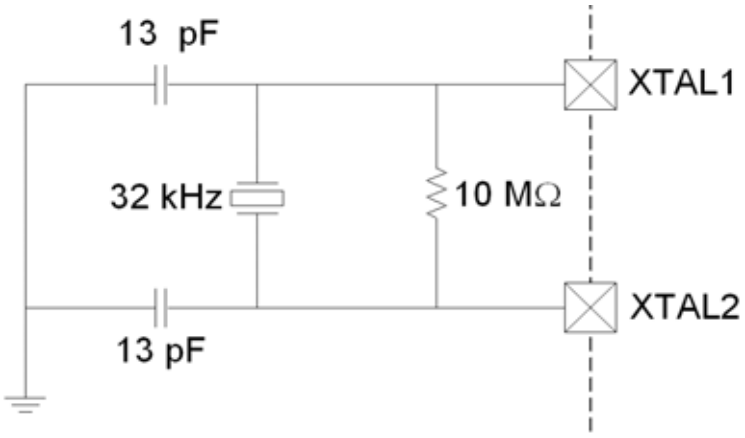


Figure 19.2. External Crystal Example

Important Note on External Crystals: Crystal oscillator circuits are quite sensitive to PCB layout. The crystal should be placed as close as possible to the XTAL pins on the device. The traces should be as short as possible and shielded with ground plane from any other traces which could introduce noise or interference.

When using an external crystal, the external oscillator drive circuit must be configured by software for *Crystal Oscillator Mode* or *Crystal Oscillator Mode with divide by 2 stage*. The divide by 2 stage ensures that the clock derived from the external oscillator has a duty cycle of 50%. The External Oscillator Frequency Control value (XFCN) must also be specified based on the crystal frequency (see SFR Definition 19.6).

When the crystal oscillator is first enabled, the external oscillator valid detector allows software to determine when the external system clock is valid and running. Switching to the external oscillator before the crystal oscillator has stabilized can result in unpredictable behavior. The recommended procedure for starting the crystal is:

1. Configure XTAL1 and XTAL2 for analog I/O.
2. Disable the XTAL1 and XTAL2 digital output drivers by writing 1s to the appropriate bits in the Port Latch register.
3. Configure and enable the external oscillator.
4. Wait at least 1 ms.
5. Poll for XTLVLD $\geq$ 1.
6. Switch the system clock to the external oscillator.

SFR Definition 20.8. P1: Port 1

Bit	7	6	5	4	3	2	1	0
Name	P1[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Address = 0x90; SFR Page = All Pages; Bit Addressable

Bit	Name	Description	Write	Read
7:0	P1[7:0]	**Port 1 Data.** Sets the Port latch logic value or reads the Port pin logic state in Port cells configured for digital I/O.	0: Set output latch to logic LOW. 1: Set output latch to logic HIGH.	0: P1.n Port pin is logic LOW. 1: P1.n Port pin is logic HIGH.

SFR Definition 20.9. P1MDIN: Port 1 Input Mode

Bit	7	6	5	4	3	2	1	0
Name	P1MDIN[7:0]							
Type	R/W							
Reset	1*	1	1	1	1	1	1	1

SFR Address = 0xF2; SFR Page = All Pages

Bit	Name	Function
7:0	P1MDIN[7:0]	**Analog Configuration Bits for P1.7–P1.0 (respectively).** Port pins configured for analog mode have their weak pullup, digital driver, and digital receiver disabled. 0: Corresponding P1.n pin is configured for analog mode. 1: Corresponding P1.n pin is not configured for analog mode.

SFR Definition 20.20. P4: Port 4

Bit	7	6	5	4	3	2	1	0
Name	P4[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Address = 0xC7; SFR Page = All Pages

Bit	Name	Description	Write	Read
7:0	P4[7:0]	**Port 4 Data.** Sets the Port latch logic value or reads the Port pin logic state in Port cells configured for digital I/O.	0: Set output latch to logic LOW. 1: Set output latch to logic HIGH.	0: P4.n Port pin is logic LOW. 1: P4.n Port pin is logic HIGH.

SFR Definition 20.21. P4MDIN: Port 4 Input Mode

Bit	7	6	5	4	3	2	1	0
Name	P4MDIN[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Address = 0xF5; SFR Page = All Pages

Bit	Name	Function
7:0	P4MDIN[7:0]	**Analog Configuration Bits for P4.7–P4.0 (respectively).** Port pins configured for analog mode have their weak pullup, digital driver, and digital receiver disabled. 0: Corresponding P4.n pin is configured for analog mode. 1: Corresponding P4.n pin is not configured for analog mode.

SFR Definition 21.1. USB0XCN: USB0 Transceiver Control

Bit	7	6	5	4	3	2	1	0
Name	PREN	PHYEN	SPEED	PHYTST[1:0]		DFREC	Dp	Dn
Type	R/W	R/W	R/W	R/W		R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD7; SFR Page = All Pages

Bit	Name	Function
7	PREN	**Internal Pull-up Resistor Enable.** The location of the pull-up resistor (D+ or D-) is determined by the SPEED bit. 0: Internal pull-up resistor disabled (device effectively detached from USB network). 1: Internal pull-up resistor enabled when VBUS is present (device attached to the USB network).
6	PHYEN	**Physical Layer Enable.** 0: USB0 physical layer Transceiver disabled (suspend). 1: USB0 physical layer Transceiver enabled (normal).
5	SPEED	**USB0 Speed Select.** This bit selects the USB0 speed. 0: USB0 operates as a Low Speed device. If enabled, the internal pull-up resistor appears on the D– line. 1: USB0 operates as a Full Speed device. If enabled, the internal pull-up resistor appears on the D+ line.
4:3	PHYTST[1:0]	**Physical Layer Test Bits.** 00: Mode 0: Normal (non-test mode) (D+ = X, D- = X) 01: Mode 1: Differential 1 Forced (D+ = 1, D- = 0) 10: Mode 2: Differential 0 Forced (D+ = 0, D- = 1) 11: Mode 3: Single-Ended 0 Forced (D+ = 0, D– = 0)
2	DFREC	**Differential Receiver Bit** The state of this bit indicates the current differential value present on the D+ and D- lines when PHYEN = 1. 0: Differential 0 signalling on the bus. 1: Differential 1 signalling on the bus.
1	Dp	**D+ Signal Status.** This bit indicates the current logic level of the D+ pin. 0: D+ signal currently at logic 0. 1: D+ signal currently at logic 1.
0	Dn	**D- Signal Status.** This bit indicates the current logic level of the D- pin. 0: D- signal currently at logic 0. 1: D- signal currently at logic 1.

SFR Definition 21.3. USB0DAT: USB0 Data

Bit	7	6	5	4	3	2	1	0
Name	USB0DAT[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x97; SFR Page = All Pages

Bit	Name	Description	Write	Read
7:0	USB0DAT[7:0]	**USB0 Data Bits.** This SFR is used to indirectly read and write USB0 registers.	Write Procedure: 1. Poll for BUSY (USB0ADR.7) => 0. 2. Load the target USB0 register address into the USBADDR bits in register USB0ADR. 3. Write data to USB0DAT. 4. Repeat (Step 2 may be skipped when writing to the same USB0 register).	Read Procedure: 1. Poll for BUSY (USB0ADR.7) => 0. 2. Load the target USB0 register address into the USBADDR bits in register USB0ADR. 3. Write 1 to the BUSY bit in register USB0ADR (steps 2 and 3 can be performed in the same write). 4. Poll for BUSY (USB0ADR.7) => 0. 5. Read data from USB0DAT. 6. Repeat from Step 2 (Step 2 may be skipped when reading the same USB0 register; Step 3 may be skipped when the AUTORD bit (USB0ADR.6) is logic 1).

A Bulk or Interrupt pipe can be shut down (or Halted) by writing 1 to the SDSTL bit (EINCSRL.4). While SDSTL = 1, hardware will respond to all IN requests with a STALL condition. Each time hardware generates a STALL condition, an interrupt will be generated and the STSTL bit (EINCSRL.5) set to 1. The STSTL bit must be reset to 0 by firmware.

Hardware will automatically reset INPRDY to 0 when a packet slot is open in the endpoint FIFO. Note that if double buffering is enabled for the target endpoint, it is possible for firmware to load two packets into the IN FIFO at a time. In this case, hardware will reset INPRDY to 0 immediately after firmware loads the first packet into the FIFO and sets INPRDY to 1. An interrupt will not be generated in this case; an interrupt will only be generated when a data packet is transmitted.

When firmware writes 1 to the FCDT bit (EINCSRH.3), the data toggle for each IN packet will be toggled continuously, regardless of the handshake received from the host. This feature is typically used by Interrupt endpoints functioning as rate feedback communication for Isochronous endpoints. When FCDT = 0, the data toggle bit will only be toggled when an ACK is sent from the host in response to an IN packet.

21.12.2. Endpoints1-3 IN Isochronous Mode

When the ISO bit (EINCSRH.6) is set to 1, the target endpoint operates in Isochronous (ISO) mode. Once an endpoint has been configured for ISO IN mode, the host will send one IN token (data request) per frame; the location of data within each frame may vary. Because of this, it is recommended that double buffering be enabled for ISO IN endpoints.

Hardware will automatically reset INPRDY (EINCSRL.0) to 0 when a packet slot is open in the endpoint FIFO. Note that if double buffering is enabled for the target endpoint, it is possible for firmware to load two packets into the IN FIFO at a time. In this case, hardware will reset INPRDY to 0 immediately after firmware loads the first packet into the FIFO and sets INPRDY to 1. An interrupt will not be generated in this case; an interrupt will only be generated when a data packet is transmitted.

If there is not a data packet ready in the endpoint FIFO when USB0 receives an IN token from the host, USB0 will transmit a zero-length data packet and set the UNDRUN bit (EINCSRL.2) to 1.

The ISO Update feature (see Section 21.7) can be useful in starting a double buffered ISO IN endpoint. If the host has already set up the ISO IN pipe (has begun transmitting IN tokens) when firmware writes the first data packet to the endpoint FIFO, the next IN token may arrive and the first data packet sent before firmware has written the second (double buffered) data packet to the FIFO. The ISO Update feature ensures that any data packet written to the endpoint FIFO will not be transmitted during the current frame; the packet will only be sent after a SOF signal has been received.

22.5.3. Write Sequence (Slave)

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

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

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

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

The interface exits Slave Receiver Mode after receiving a STOP. The interface will switch to Slave Transmitter Mode if SMB0DAT is written while an active Slave Receiver. Figure 22.7 shows a typical slave write sequence. Two received data bytes are shown, though any number of bytes may be received. Notice that the ‘data byte transferred’ interrupts occur at different places in the sequence, depending on whether hardware ACK generation is enabled. The interrupt occurs **before** the ACK with hardware ACK generation disabled, and **after** the ACK when hardware ACK generation is enabled.

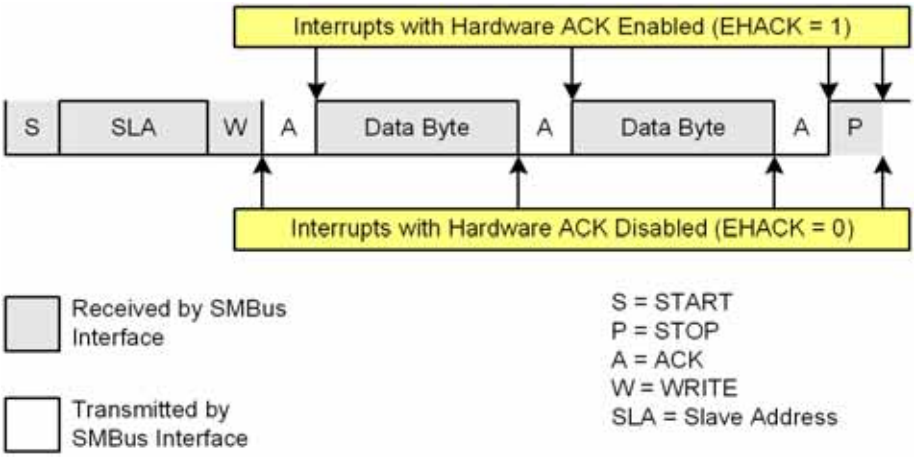


Figure 22.7. Typical Slave Write Sequence

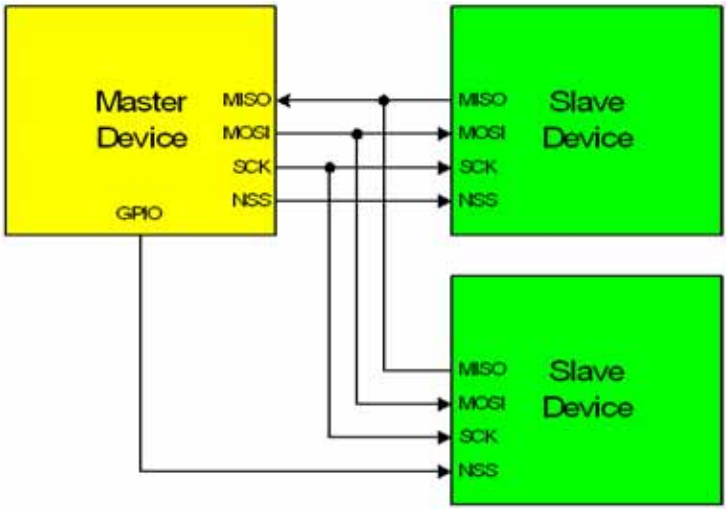


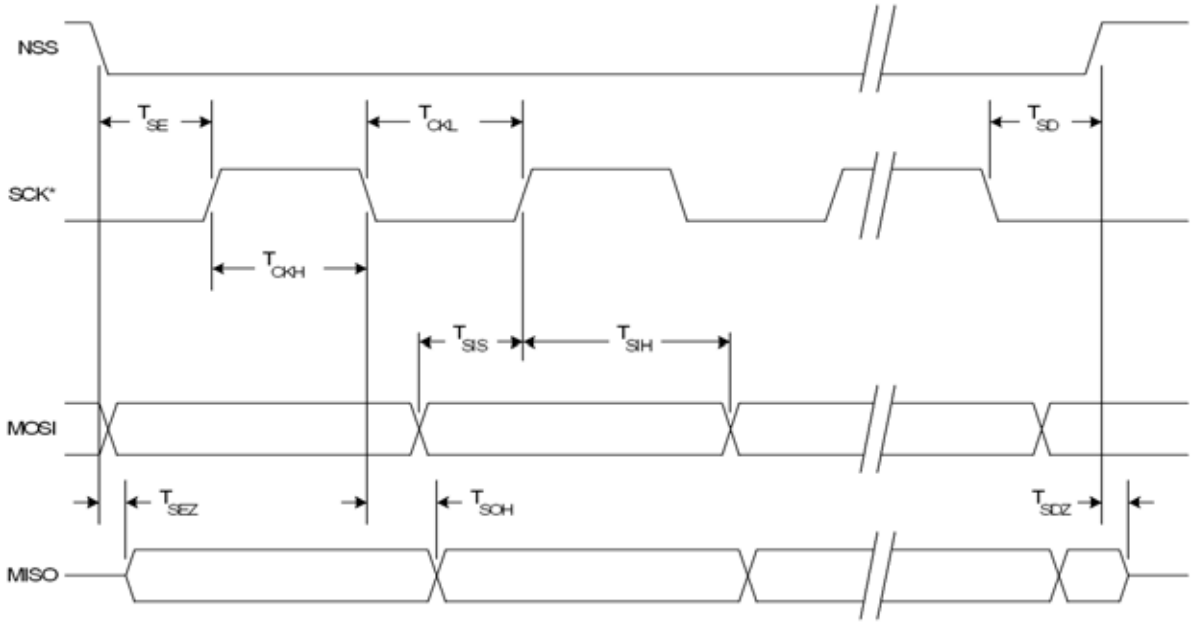
Figure 25.4. 4-Wire Single Master Mode and 4-Wire Slave Mode Connection Diagram

25.3. SPI0 Slave Mode Operation

When SPI0 is enabled and not configured as a master, it will operate as a SPI slave. As a slave, bytes are shifted in through the MOSI pin and out through the MISO pin by a master device controlling the SCK signal. A bit counter in the SPI0 logic counts SCK edges. When 8 bits have been shifted through the shift register, the SPIF flag is set to logic 1, and the byte is copied into the receive buffer. Data is read from the receive buffer by reading SPI0DAT. A slave device cannot initiate transfers. Data to be transferred to the master device is pre-loaded into the shift register by writing to SPI0DAT. Writes to SPI0DAT are double-buffered, and are placed in the transmit buffer first. If the shift register is empty, the contents of the transmit buffer will immediately be transferred into the shift register. When the shift register already contains data, the SPI will load the shift register with the transmit buffer's contents after the last SCK edge of the next (or current) SPI transfer.

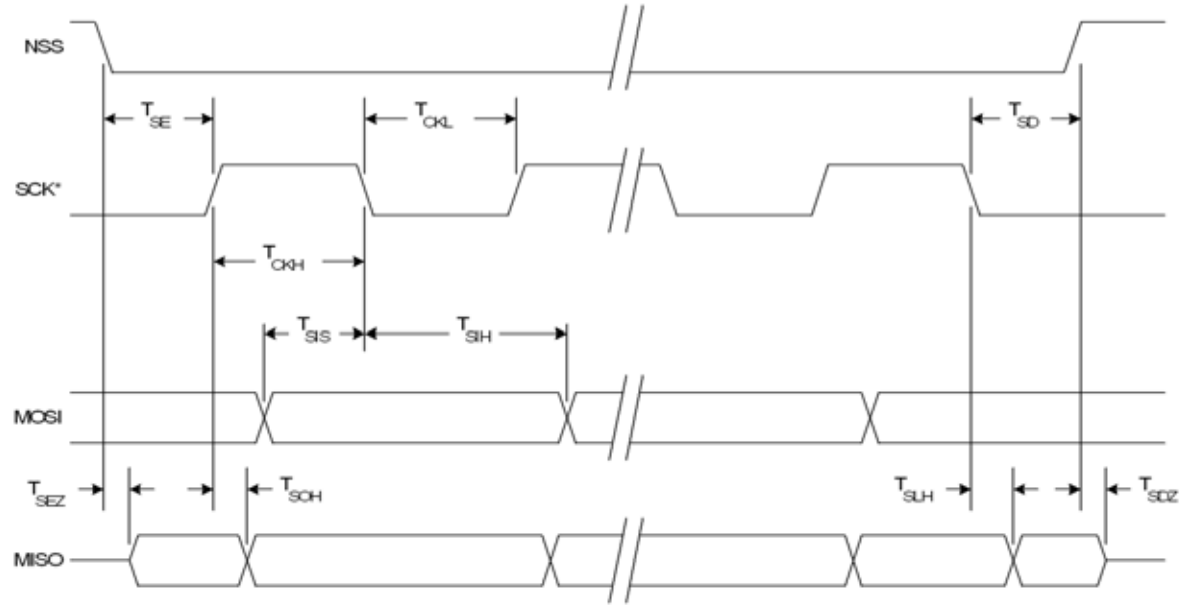
When configured as a slave, SPI0 can be configured for 4-wire or 3-wire operation. The default, 4-wire slave mode, is active when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 1. In 4-wire mode, the NSS signal is routed to a port pin and configured as a digital input. SPI0 is enabled when NSS is logic 0, and disabled when NSS is logic 1. The bit counter is reset on a falling edge of NSS. Note that the NSS signal must be driven low at least 2 system clocks before the first active edge of SCK for each byte transfer. Figure 25.4 shows a connection diagram between two slave devices in 4-wire slave mode and a master device.

3-wire slave mode is active when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 0. NSS is not used in this mode, and is not mapped to an external port pin through the crossbar. Since there is no way of uniquely addressing the device in 3-wire slave mode, SPI0 must be the only slave device present on the bus. It is important to note that in 3-wire slave mode there is no external means of resetting the bit counter that determines when a full byte has been received. The bit counter can only be reset by disabling and re-enabling SPI0 with the SPIEN bit. Figure 25.3 shows a connection diagram between a slave device in 3-wire slave mode and a master device.



* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.

Figure 25.10. SPI Slave Timing (CKPHA = 0)



* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.

Figure 25.11. SPI Slave Timing (CKPHA = 1)

SFR Definition 26.5. TL0: Timer 0 Low Byte

Bit	7	6	5	4	3	2	1	0
Name	TL0[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8A; SFR Page = All Pages

Bit	Name	Function
7:0	TL0[7:0]	**Timer 0 Low Byte.** The TL0 register is the low byte of the 16-bit Timer 0.

SFR Definition 26.6. TL1: Timer 1 Low Byte

Bit	7	6	5	4	3	2	1	0
Name	TL1[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8B; SFR Page = All Pages

Bit	Name	Function
7:0	TL1[7:0]	**Timer 1 Low Byte.** The TL1 register is the low byte of the 16-bit Timer 1.

27.3.2. Software Timer (Compare) Mode

In Software Timer mode, the PCA counter/timer value is compared to the module's 16-bit capture/compare register (PCA0CPHn and PCA0CPLn). When a match occurs, the Capture/Compare Flag (CCFn) in PCA0CN is set to logic 1. An interrupt request is generated if the CCFn interrupt for that module is enabled. The CCFn bit is not automatically cleared by hardware when the CPU vectors to the interrupt service routine, and must be cleared by software. Setting the ECOMn and MATn bits in the PCA0CPMn register enables Software Timer mode.

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.

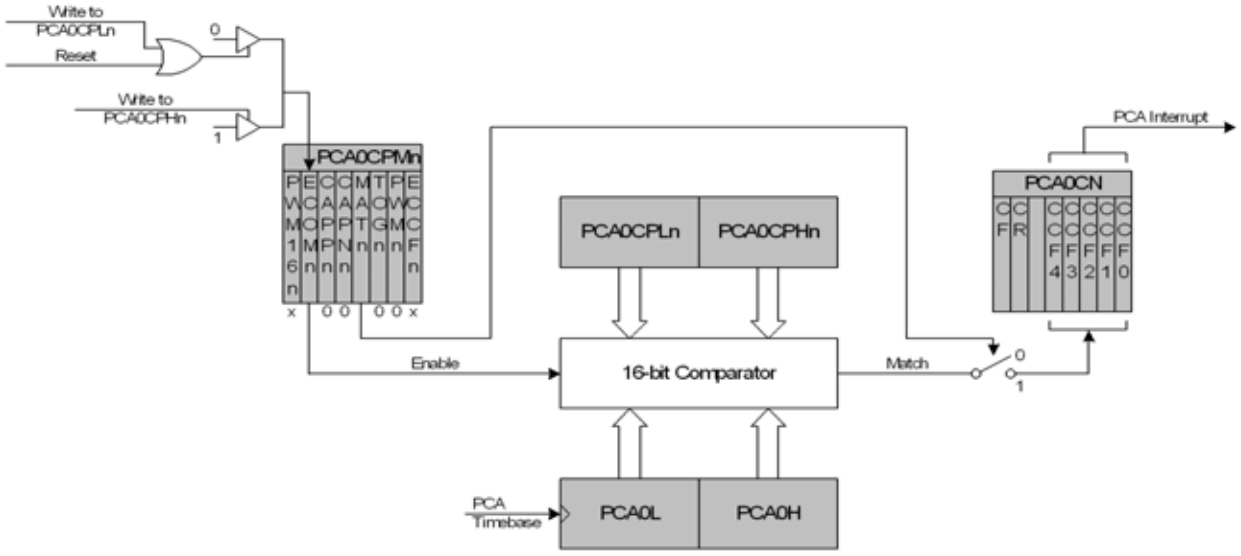


Figure 27.5. PCA Software Timer Mode Diagram