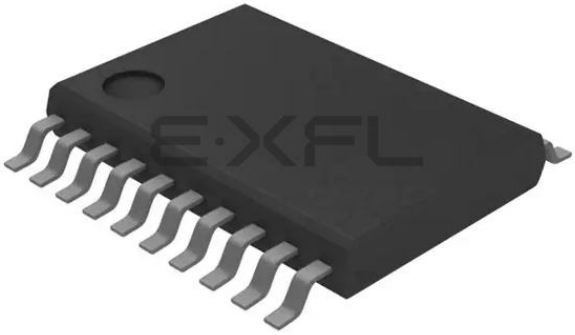


Welcome to E-XFL.COM

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	HC08
Core Size	8-Bit
Speed	8MHz
Connectivity	SCI, SPI
Peripherals	LVD, POR, PWM
Number of I/O	16
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 10x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	20-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	20-TSSOP
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc908qc16mdser

Chapter 14
System Integration Module (SIM)

Chapter 16
Timer Interface Module (TIM1)

Chapter 17
Timer Interface Module (TIM2)

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$000D	SPI Control Register (SPCR) See page 185.	Read:	SPRIE	R	SPMSTR	CPOL	CPHA	SPWOM	SPE	SPTIE
		Write:								
		Reset:	0	0	1	0	1	0	0	0
$000E	SPI Status and Control Register (SPSCR) See page 186.	Read:	SPRF	ERRIE	OVRF	MODF	SPTE	MODFEN	SPR1	SPR0
		Write:								
		Reset:	0	0	0	0	1	0	0	0
$000F	SPI Data Register (SPDR) See page 188.	Read:	R7	R6	R5	R4	R3	R2	R1	R0
		Write:	T7	T6	T5	T4	T3	T2	T1	T0
		Reset:	Unaffected by reset							
$0010	ESCI Control Register 1 (SCC1) See page 136.	Read:	LOOPS	ENSCI	TXINV	M	WAKE	ILTY	PEN	PTY
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0011	ESCI Control Register 2 (SCC2) See page 138.	Read:	SCTIE	TCIE	SCRIE	ILIE	TE	RE	RWU	SBK
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0012	ESCI Control Register 3 (SCC3) See page 141.	Read:	R8	T8	R	R	ORIE	NEIE	FEIE	PEIE
		Write:								
		Reset:	U	0	0	0	0	0	0	0
$0013	ESCI Status Register 1 (SCS1) See page 141.	Read:	SCTE	TC	SCRF	IDLE	OR	NF	FE	PE
		Write:								
		Reset:	1	1	0	0	0	0	0	0
$0014	ESCI Status Register 2 (SCS2) See page 143.	Read:	0	0	0	0	0	0	BKF	RPF
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0015	ESCI Data Register (SCDR) See page 144.	Read:	R7	R6	R5	R4	R3	R2	R1	R0
		Write:	T7	T6	T5	T4	T3	T2	T1	T0
		Reset:	Unaffected by reset							
$0016	ESCI Baud Rate Register (SCBR) See page 144.	Read:	LINT	LINR	SCP1	SCP0	R	SCR2	SCR1	SCR0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0017	ESCI Prescaler Register (SCPSC) See page 146.	Read:	PDS2	PDS1	PDS0	PSSB4	PSSB3	PSSB2	PSSB1	PSSB0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0018	ESCI Arbiter Control Register (SCIACTL) See page 149.	Read:	AM1	R	AM0	ACLK	AFIN	ARUN	AROVFL	ARD8
		Write:								
		Reset:	0	0	0	0	0	0	0	0

= Unimplemented R = Reserved U = Unaffected

Figure 2-2. Control, Status, and Data Registers (Sheet 2 of 8)

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0034	TIM1 Channel 3 Register High (T1CH3H) See page 204.	Read: Write:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Reset:	Indeterminate after reset							
$0035	TIM1 Channel 3 Register Low (T1CH3L) See page 204.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	Indeterminate after reset							
$0036	Oscillator Status and Control Register (OSCSC) See page 104.	Read: Write:	OSCOPT1	OSCOPT0	ICFS1	ICFS0	ECFS1	ECFS0	ECGON	ECGST
		Reset:	0	0	0	0	0	0	0	0
$0037	Reserved									
$0038	Oscillator Trim Register (OSCTRIM) See page 105.	Read: Write:	TRIM7	TRIM6	TRIM5	TRIM4	TRIM3	TRIM2	TRIM1	TRIM0
		Reset:	1	0	0	0	0	0	0	0
$0039 ↓ $003B	Reserved									
$003C	ADC10 Status and Control Register (ADSCR) See page 54.	Read: Write:	COCO	AIEN	ADCO	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0
		Reset:	0	0	0	1	1	1	1	1
$003D	ADC10 Data Register High (ADRH) See page 56.	Read:	0	0	0	0	0	0	AD9	AD8
		Write:	R	R	R	R	R	R	R	R
		Reset:	0	0	0	0	0	0	0	0
$003E	ADC10 Data Register Low (ADRL) See page 56.	Read:	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
		Write:	R	R	R	R	R	R	R	R
		Reset:	0	0	0	0	0	0	0	0
$003F	ADC10 Clock Register (ADCLK) See page 56.	Read: Write:	ADLPC	ADIV1	ADIV0	ADICLK	MODE1	MODE0	ADLSMP	ADACKEN
		Reset:	0	0	0	0	0	0	0	0
$0240	TIM2 Status and Control Register (T2SC) See page 213.	Read:	TOF	TOIE	TSTOP	0	0	PS2	PS1	PS0
		Write:	0			TRST				
		Reset:	0	0	1	0	0	0	0	0
$0241	TIM2 Counter Register High (T2CNTH) See page 214.	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	0	0	0	0	0	0	0	0

= Unimplemented R = Reserved U = Unaffected

Figure 2-2. Control, Status, and Data Registers (Sheet 5 of 8)

Table 2-1. Vector Addresses

Vector Priority	Vector	Address	Vector
Lowest	IF22-IF20	$FFD0-$FFD5	Unused vectors (available for user program)
	IF19	$FFD6,7	PWU vector
	IF18	$FFD8,9	TIM2 overflow vector
	IF17	$FFDA,B	TIM2 channel 1 vector
	IF16	$FFDC,D	TIM2 channel 0 vector
	IF15	$FFDE,F	ADC conversion complete vector
	IF14	$FFE0,1	Keyboard vector
	IF13	$FFE2,3	SPI transmit vector
	IF12	$FFE4,5	SPI receive vector
	IF11	$FFE6,7	ESCI transmit vector
	IF10	$FFE8,9	ESCI receive vector
	IF9	$FFEA,B	ESCI error vector
	IF8	—	Not used
	IF7	$FFEE,F	TIM1 Channel 3 vector
	IF6	$FFF0,1	TIM1 Channel 2 vector
	IF5	$FFF2,3	TIM1 overflow vector
	IF4	$FFF4,5	TIM1 Channel 1 vector
	IF3	$FFF6,7	TIM1 Channel 0 vector
	IF2	—	Not used
	IF1	$FFFA,B	$\overline{\text{IRQ}}$ vector
	—	$FFFC,D	SWI vector
Highest	—	$FFFE,F	Reset vector

2.5 Random-Access Memory (RAM)

This MCU includes static RAM. The locations in RAM below $0100 can be accessed using the more efficient direct addressing mode, and any single bit in this area can be accessed with the bit manipulation instructions (BCLR, BSET, BRCLR, and BRSET). Locating the most frequently accessed program variables in this area of RAM is preferred.

The RAM retains data when the MCU is in low-power wait or stop mode. At power-on, the contents of RAM are uninitialized. RAM data is unaffected by any reset provided that the supply voltage does not drop below the minimum value for RAM retention.

For compatibility with older M68HC05 MCUs, the HC08 resets the stack pointer to $00FF. In the devices that have RAM above $00FF, it is usually best to reinitialize the stack pointer to the top of the RAM so the direct page RAM can be used for frequently accessed RAM variables and bit-addressable program variables. Include the following 2-instruction sequence in your reset initialization routine (where RamLast is equated to the highest address of the RAM).

```
LDHX    #RamLast+1    ;point one past RAM
TXS                   ;SP<-(H:X-1)
```

2.6.3 FLASH Mass Erase Operation

Use the following procedure to erase the entire FLASH memory to read as a 1:

1. Set both the ERASE bit and the MASS bit in the FLASH control register.
2. Read the FLASH block protect register.
3. Write any data to any FLASH address[1] within the FLASH memory address range.
4. Wait for a time, t_{NVS}.
5. Set the HVEN bit.
6. Wait for a time, t_{MErase}.
7. Clear the ERASE and MASS bits.

NOTE

Mass erase is disabled whenever any block is protected (FLBPR does not equal $FF).

8. Wait for a time, t_{NVHL}.
9. Clear the HVEN bit.
10. After time, t_{RCV}, the memory can be accessed in read mode again.

NOTE

Programming and erasing of FLASH locations cannot be performed by code being executed from the FLASH memory. While these operations must be performed in the order as shown, other unrelated operations may occur between the steps.

CAUTION

A mass erase will erase the internal oscillator trim value at $FFC0.

1. When in monitor mode, with security sequence failed (see 18.3.2 Security), write to the FLASH block protect register instead of any FLASH address.

If the bus frequency is less than the ADCK frequency, precise sample time for continuous conversions cannot be guaranteed in short-sample mode (ADLSMP = 0). If the bus frequency is less than 1/11th of the ADCK frequency, precise sample time for continuous conversions cannot be guaranteed in long-sample mode (ADLSMP = 1).

When clear, the ADC10 will perform a single conversion (single conversion mode) each time ADSCR is written (assuming ADCH[4:0] do not decode all 1s).

1 = Continuous conversion following a write to ADSCR
0 = One conversion following a write to ADSCR

ADCH[4:0] — Channel Select Bits

The ADCH[4:0] bits form a 5-bit field that is used to select one of the input channels. The input channels are detailed in Table 3-2. The successive approximation converter subsystem is turned off when the channel select bits are all set to 1. This feature allows explicit disabling of the ADC10 and isolation of the input channel from the I/O pad. Terminating continuous conversion mode this way will prevent an additional, single conversion from being performed. It is not necessary to set the channel select bits to all 1s to place the ADC10 in a low-power state, however, because the module is automatically placed in a low-power state when a conversion completes.

Table 3-2. Input Channel Select

ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	Input Select[1]
0	0	0	0	0	AD0
0	0	0	0	1	AD1
0	0	0	1	0	AD2
0	0	0	1	1	AD3
0	0	1	0	0	AD4
0	0	1	0	1	AD5
0	0	1	1	0	AD6
0	0	1	1	1	AD7
0	1	0	0	0	Unused
Continuing through					Unused
1	0	1	1	1	Unused
1	1	0	0	0	AD8
1	1	0	0	1	AD9
1	1	0	1	0	BANDGAP REF[2]
1	1	0	1	1	Reserved
1	1	1	0	0	Reserved
1	1	1	0	1	V_{REFH}
1	1	1	1	0	V_{REFL}
1	1	1	1	1	Low-power state

1. If any unused or reserved channels are selected, the resulting conversion will be unknown.
2. Requires LVI to be powered (LVIPWRD =0, in CONFIG1)

6.3.3 Stack Pointer

The stack pointer is a 16-bit register that contains the address of the next location on the stack. During a reset, the stack pointer is preset to $00FF. The reset stack pointer (RSP) instruction sets the least significant byte to $FF and does not affect the most significant byte. The stack pointer decrements as data is pushed onto the stack and increments as data is pulled from the stack.

In the stack pointer 8-bit offset and 16-bit offset addressing modes, the stack pointer can function as an index register to access data on the stack. The CPU uses the contents of the stack pointer to determine the conditional address of the operand.

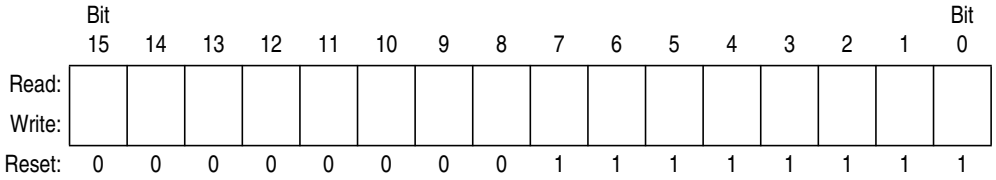


Figure 6-4. Stack Pointer (SP)

NOTE

The location of the stack is arbitrary and may be relocated anywhere in random-access memory (RAM). Moving the SP out of page 0 ($0000 to $00FF) frees direct address (page 0) space. For correct operation, the stack pointer must point only to RAM locations.

6.3.4 Program Counter

The program counter is a 16-bit register that contains the address of the next instruction or operand to be fetched.

Normally, the program counter automatically increments to the next sequential memory location every time an instruction or operand is fetched. Jump, branch, and interrupt operations load the program counter with an address other than that of the next sequential location.

During reset, the program counter is loaded with the reset vector address located at $FFFE and $FFFF. The vector address is the address of the first instruction to be executed after exiting the reset state.

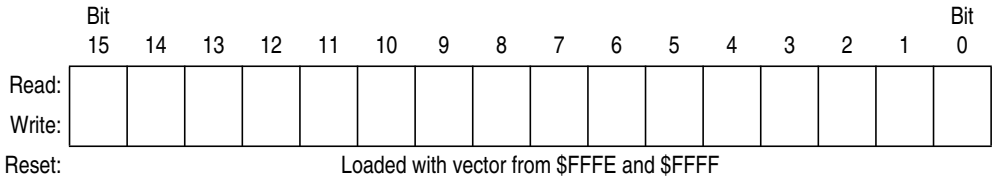
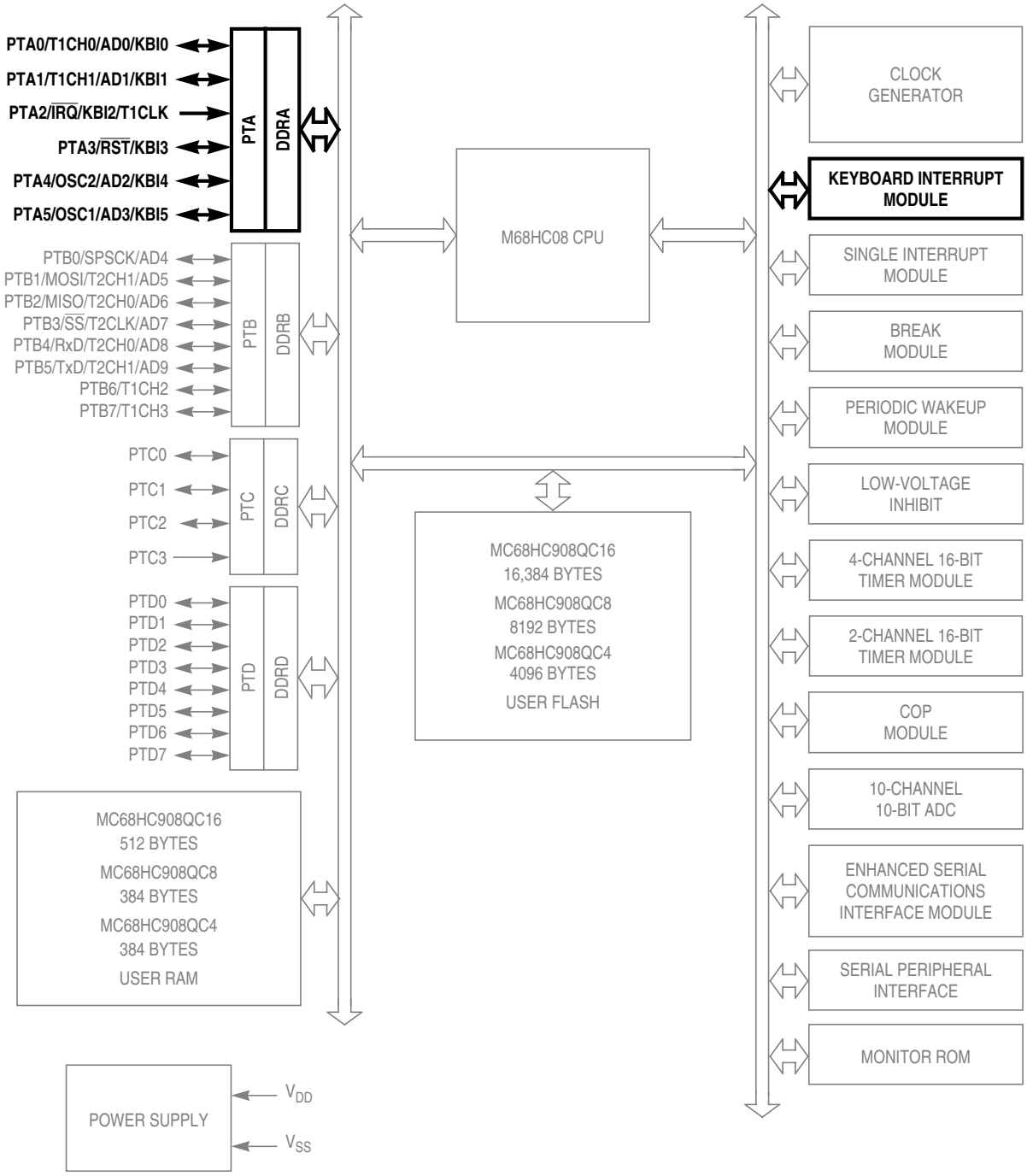


Figure 6-5. Program Counter (PC)



All port pins can be configured with internal pullup
PTC not available on 16-pin devices (see note in 11.1 Introduction)
PTD not available on 16-pin or 20-pin devices (see note in 11.1 Introduction)

Figure 8-2. Block Diagram Highlighting KBI Block and Pins

10.3.1.2 XTAL Oscillator Clock (XTALCLK)

XTALCLK is the XTAL oscillator output signal. It runs at the full speed of the crystal (f_{XCLK}) and comes directly from the crystal oscillator circuit. Figure 10-2 shows only the logical relation of XTALCLK to OSC1 and OSC2 and may not represent the actual circuitry. The duty cycle of XTALCLK is unknown and may depend on the crystal and other external factors. The frequency of XTALCLK can be unstable at start up.

10.3.1.3 RC Oscillator Clock (RCCLK)

RCCLK is the RC oscillator output signal. Its frequency is directly proportional to the time constant of the external R (R_{EXT}) and internal C. Figure 10-3 shows only the logical relation of RCCLK to OSC1 and may not represent the actual circuitry.

10.3.1.4 Internal Oscillator Clock (INTCLK)

INTCLK is the internal oscillator output signal. INTCLK is software selectable to be nominally 25.6 MHz, 12.8 MHz, 8.0 MHz, or 4.0 MHz. INTCLK can be digitally adjusted using the oscillator trimming feature of the OSCTRIM register (see 10.3.2.1 Internal Oscillator Trimming).

10.3.1.5 Bus Clock Times 4 (BUSCLKX4)

BUSCLKX4 is the same frequency as the input clock (XTALCLK, RCCLK, or INTCLK). This signal is driven to the SIM module and is used during recovery from reset and stop and is the clock source for the COP module.

10.3.1.6 Bus Clock Times 2 (BUSCLKX2)

The frequency of this signal is equal to half of the BUSCLKX4. This signal is driven to the SIM for generation of the bus clocks used by the CPU and other modules on the MCU. BUSCLKX2 will be divided by two in the SIM. The internal bus frequency is one fourth of the XTALCLK, RCCLK, or INTCLK frequency.

10.3.2 Internal Oscillator

The internal oscillator circuit is designed for use with no external components to provide a clock source with a tolerance of less than 25%untrimmed. An 8-bit register (OSCTRIM) allows the digital adjustment to a tolerance of ACC_{INT}. See the oscillator characteristics in the Electrical section of this data sheet.

The internal oscillator is capable of generating clocks of 25.6 MHz, 12.8 MHz, 8.0 MHz, or 4.0 MHz (INTCLK) resulting in a bus frequency (INTCLK divided by 4) of 6.4 MHz, 3.2 MHz, 2.0 MHz, or 1.0 MHz respectively. The bus clock is software selectable and defaults to the 1.0-MHz bus out of reset. Users can increase the bus frequency based on the voltage range of their application.

Figure 10-3 shows how BUSCLKX4 is derived from INTCLK and OSC2 can output BUSCLKX4 by setting OSC2EN.

10.3.2.1 Internal Oscillator Trimming

OSCTRIM allows a clock period adjustment of +127 and –128 steps. Increasing the OSCTRIM value increases the clock period, which decreases the clock frequency. Trimming allows the internal clock frequency to be fine tuned to the target frequency.

All devices are factory programmed with a trim value that is stored in FLASH memory at location $FFC0. This trim value is not automatically loaded into OSCTRIM register. User software must copy the trim value

- Framing error (FE) — The FE bit in SCS1 is set when a 0 occurs where the receiver expects a stop bit. The framing error interrupt enable bit, FEIE, in SCC3 enables FE to generate ESCI error interrupt requests.
- Parity error (PE) — The PE bit in SCS1 is set when the ESCI detects a parity error in incoming data. The parity error interrupt enable bit, PEIE, in SCC3 enables PE to generate ESCI error interrupt requests.

13.5 Low-Power Modes

The WAIT and STOP instructions put the MCU in low power-consumption standby modes.

13.5.1 Wait Mode

The ESCI module remains active in wait mode. Any enabled interrupt request from the ESCI module can bring the MCU out of wait mode.

If ESCI module functions are not required during wait mode, reduce power consumption by disabling the module before executing the WAIT instruction.

13.5.2 Stop Mode

The ESCI module is inactive in stop mode. The STOP instruction does not affect ESCI register states. ESCI module operation resumes after the MCU exits stop mode.

Because the internal clock is inactive during stop mode, entering stop mode during an ESCI transmission or reception results in invalid data.

13.6 ESCI During Break Interrupts

The system integration module (SIM) controls whether status bits in other modules can be cleared during the break state. The BCFE bit in the break flag control register (BFCR) enables software to clear status bits during the break state. See BFCR in the SIM section of this data sheet.

To allow software to clear status bits during a break interrupt, write a 1 to BCFE. If a status bit is cleared during the break state, it remains cleared when the MCU exits the break state.

To protect status bits during the break state, write a 0 to BCFE. With BCFE cleared (its default state), software can read and write registers during the break state without affecting status bits. Some status bits have a two-step read/write clearing procedure. If software does the first step on such a bit before the break, the bit cannot change during the break state as long as BCFE is cleared. After the break, doing the second step clears the status bit.

13.7 I/O Signals

The ESCI module can share its pins with the general-purpose I/O pins. See Figure 13-1 for the port pins that are shared.

13.7.1 ESCI Transmit Data (TxD)

The TxD pin is the serial data output from the ESCI transmitter. When the ESCI is enabled, the TxD pin becomes an output.

input ($\overline{SS}$) is low, so that only the selected slave drives to the master. The $\overline{SS}$ pin of the master is not shown but is assumed to be inactive. The $\overline{SS}$ pin of the master must be high or must be reconfigured as general-purpose I/O not affecting the SPI. (See 15.3.6.2 Mode Fault Error.) When CPHA = 0, the first SPSCK edge is the MSB capture strobe. Therefore, the slave must begin driving its data before the first SPSCK edge, and a falling edge on the $\overline{SS}$ pin is used to start the slave data transmission. The slave's $\overline{SS}$ pin must be toggled back to high and then low again between each byte transmitted as shown in Figure 15-5.

When CPHA = 0 for a slave, the falling edge of $\overline{SS}$ indicates the beginning of the transmission. This causes the SPI to leave its idle state and begin driving the MISO pin with the MSB of its data. After the transmission begins, no new data is allowed into the shift register from the transmit data register. Therefore, the SPI data register of the slave must be loaded with transmit data before the falling edge of $\overline{SS}$. Any data written after the falling edge is stored in the transmit data register and transferred to the shift register after the current transmission.

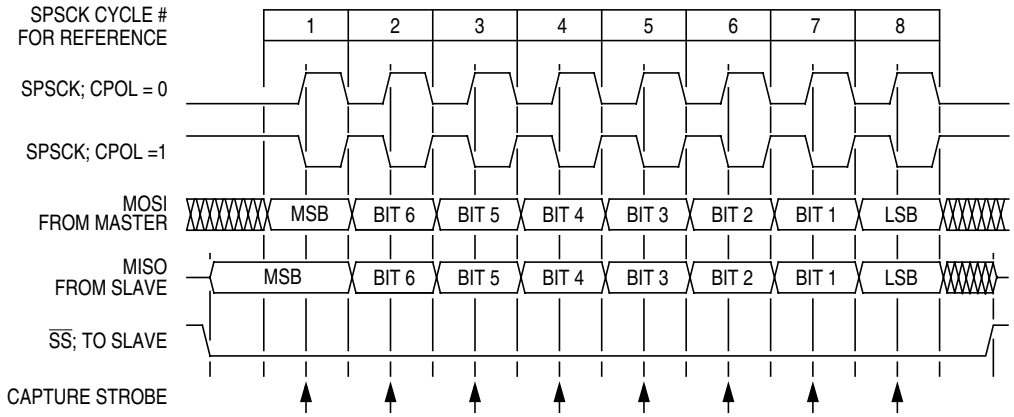


Figure 15-4. Transmission Format (CPHA = 0)

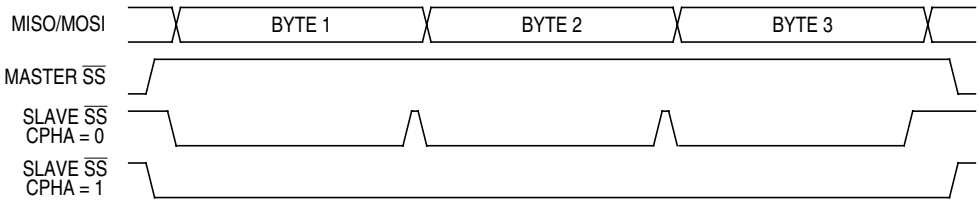


Figure 15-5. CPHA/$\overline{SS}$ Timing

15.3.3.3 Transmission Format When CPHA = 1

Figure 15-6 shows an SPI transmission in which CPHA = 1. The figure should not be used as a replacement for data sheet parametric information. Two waveforms are shown for SPSCK: one for CPOL = 0 and another for CPOL = 1. The diagram may be interpreted as a master or slave timing diagram because the serial clock (SPSCK), master in/slave out (MISO), and master out/slave in (MOSI) pins are directly connected between the master and the slave. The MISO signal is the output from the slave, and the MOSI signal is the output from the master. The $\overline{SS}$ line is the slave select input to the slave. The slave SPI drives its MISO output only when its slave select input ($\overline{SS}$) is low, so that only the selected slave drives to the master. The $\overline{SS}$ pin of the master is not shown but is assumed to be inactive. The $\overline{SS}$

16.6 TIM1 During Break Interrupts

A break interrupt stops the counter and inhibits input captures.

The system integration module (SIM) controls whether status bits in other modules can be cleared during the break state. The BCFE bit in the break flag control register (BFCR) enables software to clear status bits during the break state. See BFCR in the SIM section of this data sheet.

To allow software to clear status bits during a break interrupt, write a 1 to BCFE. If a status bit is cleared during the break state, it remains cleared when the MCU exits the break state.

To protect status bits during the break state, write a 0 to BCFE. With BCFE cleared (its default state), software can read and write registers during the break state without affecting status bits. Some status bits have a two-step read/write clearing procedure. If software does the first step on such a bit before the break, the bit cannot change during the break state as long as BCFE is cleared. After the break, doing the second step clears the status bit.

16.7 I/O Signals

The TIM1 module can share its pins with the general-purpose I/O pins. See Figure 16-1 for the port pins that are shared.

16.7.1 TIM1 Channel I/O Pins (T1CH3:T1CH0)

Each channel I/O pin is programmable independently as an input capture pin or an output compare pin. T1CH0 and T1CH2 can be configured as buffered output compare or buffered PWM pins.

16.7.2 TIM1 Clock Pin (T1CLK)

T1CLK is an external clock input that can be the clock source for the counter instead of the prescaled internal bus clock. Select the T1CLK input by writing 1s to the three prescaler select bits, PS[2:0]. The Timer Interface Module Characteristics table in the Electricals section. The maximum T1CLK frequency is the least of 4 MHz or bus frequency ÷ 2.

16.8 Registers

The following registers control and monitor operation of the TIM1:

- TIM1 status and control register (T1SC)
- TIM1 control registers (T1CNTH:T1CNTL)
- TIM1 counter modulo registers (T1MODH:T1MODL)
- TIM1 channel status and control registers (T1SC0 through T1SC3)
- TIM1 channel registers (T1CH0H:T1CH0L through T1CH3H:T1CH3L)

16.8.1 TIM1 Status and Control Register

The TIM1 status and control register (T1SC) does the following:

- Enables TIM1 overflow interrupts
- Flags TIM1 overflows
- Stops the counter
- Resets the counter
- Prescales the counter clock

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	CH0F	CH0IE	MS0B	MS0A	ELS0B	ELS0A	TOV0	CH0MAX
Write:	0							
Reset:	0	0	0	0	0	0	0	0

Figure 16-9. TIM1 Channel 0 Status and Control Register (T1SC0)

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	CH1F	CH1IE	0	MS1A	ELS1B	ELS1A	TOV1	CH1MAX
Write:	0							
Reset:	0	0	0	0	0	0	0	0

Figure 16-10. TIM1 Channel 1 Status and Control Register (T1SC1)

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	CH2F	CH2IE	MS2B	MS2A	ELS2B	ELS2A	TOV2	CH2MAX
Write:	0							
Reset:	0	0	0	0	0	0	0	0

Figure 16-11. TIM1 Channel 2 Status and Control Register (T1SC2)

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	CH3F	CH3IE	0	MS3A	ELS3B	ELS3A	TOV3	CH3MAX
Write:	0							
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 16-12. TIM1 Channel 3 Status and Control Register (T1SC3)

CHxF — Channel x Flag Bit

When channel x is an input capture channel, this read/write bit is set when an active edge occurs on the channel x pin. When channel x is an output compare channel, CHxF is set when the value in the counter registers matches the value in the TIM1 channel x registers.

Clear CHxF by reading the T1SCx register with CHxF set and then writing a 0 to CHxF. If another interrupt request occurs before the clearing sequence is complete, then writing 0 to CHxF has no effect. Therefore, an interrupt request cannot be lost due to inadvertent clearing of CHxF.

Writing a 1 to CHxF has no effect.

1 = Input capture or output compare on channel x
0 = No input capture or output compare on channel x

CHxIE — Channel x Interrupt Enable Bit

This read/write bit enables TIM1 interrupt service requests on channel x.

1 = Channel x interrupt requests enabled
0 = Channel x interrupt requests disabled

17.3.4.2 Buffered PWM Signal Generation

Channels 0 and 1 can be linked to form a buffered PWM channel whose output appears on the T2CH0 pin. The TIM2 channel registers of the linked pair alternately control the output.

Setting the MS0B bit in TIM2 channel 0 status and control register (T2SC0) links channel 0 and channel 1. The TIM2 channel 0 registers initially control the pulse width on the T2CH0 pin. Writing to the TIM2 channel 1 registers enables the TIM2 channel 1 registers to synchronously control the pulse width at the beginning of the next PWM period. At each subsequent overflow, the TIM2 channel registers (0 or 1) that control the pulse width are the ones written to last. T2SC0 controls and monitors the buffered PWM function, and TIM2 channel 1 status and control register (T2SC1) is unused. While the MS0B bit is set, the channel 1 pin, T2CH1, is available as a general-purpose I/O pin.

NOTE

In buffered PWM signal generation, do not write new pulse width values to the currently active channel registers. User software should track the currently active channel to prevent writing a new value to the active channel. Writing to the active channel registers is the same as generating unbuffered PWM signals.

17.3.4.3 PWM Initialization

To ensure correct operation when generating unbuffered or buffered PWM signals, use the following initialization procedure:

1. In the TIM2 status and control register (T2SC):
   a. Stop the counter by setting the TIM2 stop bit, TSTOP.
   b. Reset the counter and prescaler by setting the TIM2 reset bit, TRST.
2. In the TIM2 counter modulo registers (T2MODH:T2MODL), write the value for the required PWM period.
3. In the TIM2 channel x registers (T2CHxH:T2CHxL), write the value for the required pulse width.
4. In TIM2 channel x status and control register (T2SCx):
   a. Write 0:1 (for unbuffered output compare or PWM signals) or 1:0 (for buffered output compare or PWM signals) to the mode select bits, MSxB:MSxA. See Table 17-2.
   b. Write 1 to the toggle-on-overflow bit, TOVx.
   c. Write 1:0 (polarity 1 — to clear output on compare) or 1:1 (polarity 0 — to set output on compare) to the edge/level select bits, ELSxB:ELSxA. The output action on compare must force the output to the complement of the pulse width level. See Table 17-2.

NOTE

In PWM signal generation, do not program the PWM channel to toggle on output compare. Toggling on output compare prevents reliable 0% duty cycle generation and removes the ability of the channel to self-correct in the event of software error or noise. Toggling on output compare can also cause incorrect PWM signal generation when changing the PWM pulse width to a new, much larger value.

5. In the TIM2 status control register (T2SC), clear the TIM2 stop bit, TSTOP.

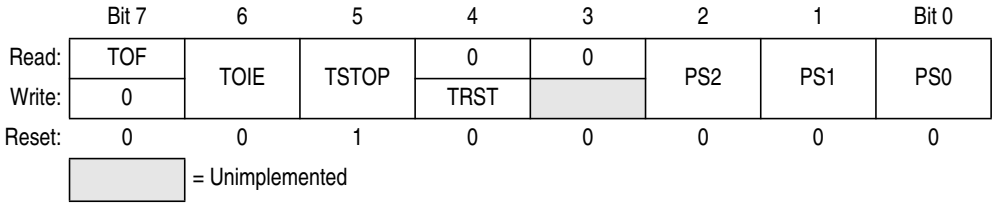


Figure 17-4. TIM2 Status and Control Register (T2SC)

TOF — TIM2 Overflow Flag Bit

This read/write flag is set when the counter reaches the modulo value programmed in the TIM2 counter modulo registers. Clear TOF by reading the T2SC register when TOF is set and then writing a 0 to TOF. If another TIM2 overflow occurs before the clearing sequence is complete, then writing 0 to TOF has no effect. Therefore, a TOF interrupt request cannot be lost due to inadvertent clearing of TOF. Writing a 1 to TOF has no effect.

1 = Counter has reached modulo value
0 = Counter has not reached modulo value

TOIE — TIM2 Overflow Interrupt Enable Bit

This read/write bit enables TIM2 overflow interrupts when the TOF bit becomes set.

1 = TIM2 overflow interrupts enabled
0 = TIM2 overflow interrupts disabled

TSTOP — TIM2 Stop Bit

This read/write bit stops the counter. Counting resumes when TSTOP is cleared. Reset sets the TSTOP bit, stopping the counter until software clears the TSTOP bit.

1 = Counter stopped
0 = Counter active

NOTE

Do not set the TSTOP bit before entering wait mode if the TIM2 is required to exit wait mode. Also, when the TSTOP bit is set and the timer is configured for input capture operation, input captures are inhibited until the TSTOP bit is cleared.

When using TSTOP to stop the timer counter, see if any timer flags are set. If a timer flag is set, it must be cleared by clearing TSTOP, then clearing the flag, then setting TSTOP again.

TRST — TIM2 Reset Bit

Setting this write-only bit resets the counter and the TIM2 prescaler. Setting TRST has no effect on any other timer registers. Counting resumes from $0000. TRST is cleared automatically after the counter is reset and always reads as 0.

1 = Prescaler and counter cleared
0 = No effect

NOTE

Setting the TSTOP and TRST bits simultaneously stops the counter at a value of $0000.

BCFE — Break Clear Flag Enable Bit

This read/write bit enables software to clear status bits by accessing status registers while the MCU is in a break state. To clear status bits during the break state, the BCFE bit must be set.

1 = Status bits clearable during break
0 = Status bits not clearable during break

18.2.3 Low-Power Modes

The WAIT and STOP instructions put the MCU in low power-consumption standby modes. If enabled, the break module will remain enabled in wait and stop modes. However, since the internal address bus does not increment in these modes, a break interrupt will never be triggered.

18.3 Monitor Module (MON)

The monitor module allows debugging and programming of the microcontroller unit (MCU) through a single-wire interface with a host computer. Monitor mode entry can be achieved without use of the higher test voltage, V_{TST}, as long as vector addresses $FFFE and $FFFF are blank, thus reducing the hardware requirements for in-circuit programming.

Features include:

- Normal user-mode pin functionality
- One pin dedicated to serial communication between MCU and host computer
- Standard non-return-to-zero (NRZ) communication with host computer
- Standard communication baud rate (7200 @ 2-MHz bus frequency)
- Execution of code in random-access memory (RAM) or FLASH
- FLASH memory security feature[1]
- FLASH memory programming interface
- Use of external 9.8304 MHz oscillator to generate internal frequency of 2.4576 MHz
- Simple internal oscillator mode of operation (no external clock or high voltage)
- Monitor mode entry without high voltage, V_{TST}, if reset vector is blank ($FFFE and $FFFF contain $FF)
- Normal monitor mode entry if V_{TST} is applied to $\overline{IRQ}$

18.3.1 Functional Description

Figure 18-9 shows a simplified diagram of monitor mode entry.

The monitor module receives and executes commands from a host computer. Figure 18-10, Figure 18-11, and Figure 18-12 show example circuits used to enter monitor mode and communicate with a host computer via a standard RS-232 interface.

1. No security feature is absolutely secure. However, Freescale's strategy is to make reading or copying the FLASH difficult for unauthorized users.

19.16 Timer Interface Module Characteristics

Characteristic	Symbol	Min	Max	Unit
Timer input capture pulse width(1)	t_{TH}, t_{TL}	2	—	t_{CYC}
Timer input capture period	t_{TLTL}	Note(2)	—	t_{CYC}
Timer input clock pulse width(1)	t_{TCL}, t_{TCH}	t_{CYC} + 5	—	ns

1. Values are based on characterization results, not tested in production.
2. The minimum period is the number of cycles it takes to execute the interrupt service routine plus 1 t_{CYC}.

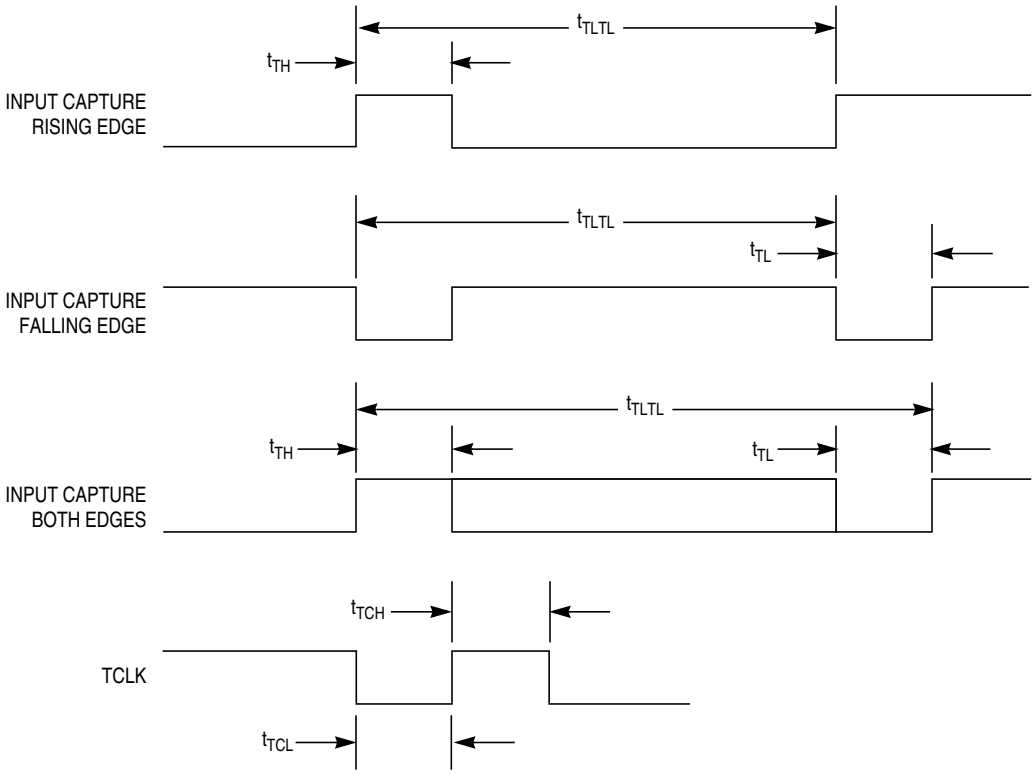


Figure 19-13. Timer Input Timing

NOTES:

1. CONTROLLING DIMENSION: MILLIMETER

2. DIMENSIONS AND TOLERANCES PER ANSI Y14.5M-1982.

3. DIMENSION DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS. MOLD FLASH OR GATE BURRS SHALL NOT EXCEED 0.15 PER SIDE.

4. DIMENSION DOES NOT INCLUDE INTERLEAD FLASH OR PROTRUSION. INTERLEAD FLASH OR PROTRUSION SHALL NOT EXCEED 0.25 PER SIDE.

5. DIMENSION DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08 TOTAL IN EXCESS OF THE DIMENSION AT MAXIMUM MATERIAL CONDITION.

6. TERMINAL NUMBERS ARE SHOWN FOR REFERENCE ONLY.

7. DIMENSIONS ARE TO BE DETERMINED AT DATUM PLANE -W-.

MECHANICAL OUTLINE	PRINT VERSION NOT TO SCALE

TITLE:	DOCUMENT NO: 98ASH70169A	REV: C
20 LD TSSOP, PITCH 0.65MM	CASE NUMBER: 948E-02	25 MAY 2005
	STANDARD: JEDEC	