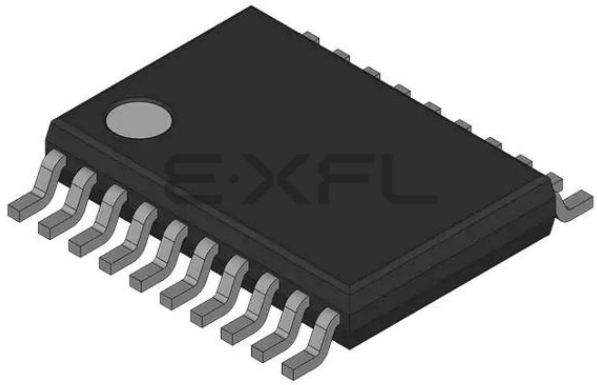


Welcome to E-XFL.COM

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/pro/item?MUrl=&PartUrl=mc908qc16mdse

MC68HC908QC16
MC68HC908QC8
MC68HC908QC4

Data Sheet

M68HC08
Microcontrollers

MC68HC908QC16
Rev. 5
4/2008

freescale.com

freescale™
semiconductor

Table of Contents

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0019	ESCI Arbiter Data Register (SCIADAT) See page 150.	Read:	ARD7	ARD6	ARD5	ARD4	ARD3	ARD2	ARD1	ARD0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$001A	Keyboard Status and Control Register (KBSCR) See page 90.	Read:	0	0	0	0	KEYF	0	IMASKK	MODEK
		Write:						ACKK		
		Reset:	0	0	0	0	0	0	0	0
$001B	Keyboard Interrupt Enable Register (KBIER) See page 90.	Read:	0	R	KBIE5	KBIE4	KBIE3	KBIE2	KBIE1	KBIE0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$001C	Keyboard Interrupt Polarity Register (KBIPR) See page 91.	Read:	0	0	KBIP5	KBIP4	KBIP3	KBIP2	KBIP1	KBIP0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$001D	IRQ Status and Control Register (INTSCR) See page 83.	Read:	0	0	0	0	IRQF	0	IMASK	MODE
		Write:						ACK		
		Reset:	0	0	0	0	0	0	0	0
$001E	Configuration Register 2 (CONFIG2)(1) See page 59.	Read:	IRQPUD	IRQEN	0	0	TIM2POS	ESCIBDSRC	OSCENIN-STOP	RSTEN
		Write:								
		Reset:	0	0	0	0	0	0	0	0(2)

1. One-time writable register after each reset.
2. RSTEN reset to 0 by a power-on reset (POR) only.

$001F	Configuration Register 1 (CONFIG1)(1) See page 60.	Read:	COPRS	LVISTOP	LVIRSTD	LVIPWRD	LVITRIP	SSREC	STOP	COPD
		Write:								
		Reset:	0	0	0	0	0(2)	0	0	0

1. One-time writable register after each reset.
2. LVI5OR3 reset to 0 by a power-on reset (POR) only.

$0020	TIM1 Status and Control Register (T1SC) See page 198.	Read:	TOF	TOIE	TSTOP	0	0	PS2	PS1	PS0
		Write:	0			TRST				
		Reset:	0	0	1	0	0	0	0	0
$0021	TIM1 Counter Register High (T1CNTH) See page 199.	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
$0022	TIM1 Counter Register Low (T1CNTL) See page 199.	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0023	TIM1 Counter Modulo Register High (T1MODH) See page 200.	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	1	1	1	1	1	1	1	1

= Unimplemented R = Reserved U = Unaffected

Figure 2-2. Control, Status, and Data Registers (Sheet 3 of 8)

2.6 FLASH Memory (FLASH)

The FLASH memory is intended primarily for program storage. In-circuit programming allows the operating program to be loaded into the FLASH memory after final assembly of the application product. It is possible to program the entire array through the single-wire monitor mode interface. Because no special voltages are needed for FLASH erase and programming operations, in-application programming is also possible through other software-controlled communication paths.

This subsection describes the operation of the embedded FLASH memory. The FLASH memory can be read, programmed, and erased from the internal V_{DD} supply. The program and erase operations are enabled through the use of an internal charge pump.

The minimum size of FLASH memory that can be erased is 64 bytes; and the maximum size of FLASH memory that can be programmed in a program cycle is 32 bytes (a row). Program and erase operations are facilitated through control bits in the FLASH control register (FLCR). Details for these operations appear later in this section.

NOTE

An erased bit reads as a 1 and a programmed bit reads as a 0. A security feature prevents viewing of the FLASH contents.[1]

2.6.1 FLASH Control Register

The FLASH control register (FLCR) controls FLASH program and erase operations.

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	HVEN	MASS	ERASE	PGM
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 2-3. FLASH Control Register (FLCR)

HVEN — High Voltage Enable Bit

This read/write bit enables high voltage from the charge pump to the memory for either program or erase operation. It can only be set if either PGM =1 or ERASE =1 and the proper sequence for program or erase is followed.

1 = High voltage enabled to array and charge pump on
0 = High voltage disabled to array and charge pump off

MASS — Mass Erase Control Bit

This read/write bit configures the memory for mass erase operation.

1 = Mass erase operation selected
0 = Mass erase operation unselected

ERASE — Erase Control Bit

This read/write bit configures the memory for erase operation. ERASE is interlocked with the PGM bit such that both bits cannot be equal to 1 or set to 1 at the same time.

1 = Erase operation selected
0 = Erase operation unselected

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

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.

2.6.5 FLASH Protection

Due to the ability of the on-board charge pump to erase and program the FLASH memory in the target application, provision is made to protect blocks of memory from unintentional erase or program operations due to system malfunction. This protection is done by use of a FLASH block protect register (FLBPR). The FLBPR determines the range of the FLASH memory which is to be protected. The range of the protected area starts from a location defined by FLBPR and ends to the bottom of the FLASH memory ($FFFF). When the memory is protected, the HVEN bit cannot be set in either ERASE or PROGRAM operations.

NOTE

In performing a program or erase operation, the FLASH block protect register must be read after setting the PGM or ERASE bit and before asserting the HVEN bit.

When the FLBPR is programmed with all 0 s, the entire memory is protected from being programmed and erased. When all the bits are erased (all 1's), the entire memory is accessible for program and erase.

When bits within the FLBPR are programmed, they lock a block of memory. The address ranges are shown in 2.6.6 FLASH Block Protect Register. Once the FLBPR is programmed with a value other than $FF, any erase or program of the FLBPR or the protected block of FLASH memory is prohibited. Mass erase is disabled whenever any block is protected (FLBPR does not equal $FF). The FLBPR itself can be erased or programmed only with an external voltage, V_{TST}, present on the $\overline{IRQ}$ pin. This voltage also allows entry from reset into the monitor mode.

2.6.6 FLASH Block Protect Register

The FLASH block protect register is implemented as a byte within the FLASH memory, and therefore can only be written during a programming sequence of the FLASH memory. The value in this register determines the starting address of the protected range within the FLASH memory.

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	BPR7	BPR6	BPR5	BPR4	BPR3	BPR2	BPR1	BPR0
Reset:	Unaffected by reset. Initial value from factory is 1.							

Write to this register is by a programming sequence to the FLASH memory.

Figure 2-5. FLASH Block Protect Register (FLBPR)

BPR[7:0] — FLASH Protection Register Bits [7:0]

These eight bits in FLBPR represent bits [13:6] of a 16-bit memory address. Bits [15:14] are 1s and bits [5:0] are 0s.

The resultant 16-bit address is used for specifying the start address of the FLASH memory for block protection. The FLASH is protected from this start address to the end of FLASH memory, at $FFFF. With this mechanism, the protect start address can be $XX00, $XX40, $XX80, or $XXC0 within the FLASH memory. See Figure 2-6 and Table 2-2.

8.8.3 Keyboard Interrupt Polarity Register (KBIPR)

KBIPR determines the polarity of the enabled keyboard interrupt pin and enables the appropriate pullup or pulldown device.

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	KBIP5	KBIP4	KBIP3	KBIP2	KBIP1	KBIP0
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 8-5. Keyboard Interrupt Polarity Register (KBIPR)

KBIP5–KBIP0 — Keyboard Interrupt Polarity Bits

Each of these read/write bits enables the polarity of the keyboard interrupt detection.

1 = Keyboard polarity is high level and/or rising edge

0 = Keyboard polarity is low level and/or falling edge

11.6.3 Port D Input Pullup Enable Register

The port D input pullup enable register (PTDPUE) contains a software configurable pullup device for each of the eight port D pins. Each bit is individually configurable and requires the corresponding data direction register, DDRDx, be configured as input. Each pullup device is automatically and dynamically disabled when its corresponding DDRDx bit is configured as output.

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	PTDPUE7	PTDPUE6	PTDPUE5	PTDPUE4	PTDPUE3	PTDPUE2	PTDPUE1	PTDPUE0
Reset:	0	0	0	0	0	0	0	0

Figure 11-16. Port D Input Pullup Enable Register (PTDPUE)

PTDPUE[7:0] — Port D Input Pullup Enable Bits

These read/write bits are software programmable to enable pullup devices on port D pins

1 = Corresponding port D pin configured to have internal pull if its DDRD bit is set to 0

0 = Pullup device is disconnected on the corresponding port D pin regardless of the state of its DDRD bit.

Table 11-4 summarizes the operation of the port D pins.

Table 11-4. Port D Pin Functions

PTDPUE Bit	DDRD Bit	PTD Bit	I/O Pin Mode	Accesses to DDRD	Accesses to PTD	
				Read/Write	Read	Write
1	0	X(1)	Input, V_{DD}(2)	DDRD7–DDRD0	Pin	PTD7–PTD0(3)
0	0	X	Input, Hi-Z(4)	DDRD7–DDRD0	Pin	PTD7–PTD0(3)
X	1	X	Output	DDRD7–DDRD0	PTD7–PTD0	PTD7–PTD0

1. X = don't care
2. I/O pin pulled to V_{DD} by internal pullup.
3. Writing affects data register, but does not affect input.
4. Hi-Z = high impedance

Chapter 12
Periodic Wakeup Module (PWU)

12.1 Introduction

This section describes the periodic wakeup (PWU) module.The PWU is available in all modes of operation (run, wait, and stop) and performs two main functions:

- Generate periodic wakeup requests to bring the microcontroller unit (MCU) out of stop mode.
- Generate periodic interrupt requests during run and wait modes.

12.2 Features

Features of the periodic wakeup module include:

- Interrupt with separate interrupt enable bit, interrupt vector and interrupt mask bit
- Exit from low-power stop mode without external signals
- Programmable clock input
- Selectable timeout periods (40 μs to 3 minutes with an adjustment resolution of better than 1% for periods over 4 ms)
- Dedicated low-power 32 kHz internal oscillator separate from the main system clock sources
- Option to allow bus clock source to run the PWU
- Accessible in all modes of operation (run, wait, and stop)

12.3 Functional Description

Figure 12-1 is a block diagram of the PWU.

The PWU module consists of a PWU counter whose count is reset once it equals the value stored in the PWU modulo register (PWUMOD). The PWU counter clock, PWUCLOCK, frequency is selectable using the PWU prescaler register (PWUP). The prescaler clock source can be selected using the PWUCLKSEL bit in the PWU status and control register (PWUSC) and can either be the internal RC oscillator or the BUSCLKX4 clock.The PWUCLKSEL bit, PWUMOD and PWUP registers can only be written to when the PWUON bit is clear.

The PWUON bit in PWUSC is used to enable the PWU module. The SMODE bit in PWUSC is used to allow an enabled PWU module to continue running in stop mode. BUSCLKX4 must be enabled to run in stop mode if used as the clock source for the PWU when SMODE is set. See Chapter 4 Configuration Registers (CONFIG1 and CONFIG2) on enabling BUSCLKX4 to run in stop mode.

The PWU counter when enabled will count until it reaches the value stored in the PWU modulo register (PWUMOD), when they are equal the read-only PWU flag (PWUF) in PWUSC is latched. The PWU interrupt enable bit, PWUIE, in PWUSC enables a PWU interrupt request. The PWUF can be cleared by writing to the PWU acknowledge bit, PWUACK in PWUSC or by a PWU interrupt vector fetch.

14.4.2.3 Illegal Opcode Reset

The SIM decodes signals from the CPU to detect illegal instructions. An illegal instruction sets the ILOP bit in the SIM reset status register (SRSR) and causes a reset.

If the stop enable bit, STOP, in the mask option register is 0, the SIM treats the STOP instruction as an illegal opcode and causes an illegal opcode reset. The SIM actively pulls down the $\overline{\text{RST}}$ pin for all internal reset sources.

14.4.2.4 Illegal Address Reset

An opcode fetch from an unmapped address generates an illegal address reset. The SIM verifies that the CPU is fetching an opcode prior to asserting the ILAD bit in the SIM reset status register (SRSR) and resetting the MCU. A data fetch from an unmapped address does not generate a reset. The SIM actively pulls down the $\overline{\text{RST}}$ pin for all internal reset sources. See Figure 2-1. Memory Map for memory ranges.

14.4.2.5 Low-Voltage Inhibit (LVI) Reset

The LVI asserts its output to the SIM when the V_{DD} voltage falls to the LVI trip voltage V_{TRIPF}. The LVI bit in the SIM reset status register (SRSR) is set, and the external reset pin ($\overline{\text{RST}}$) is held low while the SIM counter counts out 4096 BUSCLKX4 cycles after V_{DD} rises above V_{TRIPR}. Sixty-four BUSCLKX4 cycles later, the CPU and memories are released from reset to allow the reset vector sequence to occur. The SIM actively pulls down the ($\overline{\text{RST}}$) pin for all internal reset sources.

14.5 SIM Counter

The SIM counter is used by the power-on reset module (POR) and in stop mode recovery to allow the oscillator time to stabilize before enabling the internal bus (IBUS) clocks. The SIM counter also serves as a prescaler for the computer operating properly module (COP). The SIM counter uses 12 stages for counting, followed by a 13th stage that triggers a reset of SIM counters and supplies the clock for the COP module. The SIM counter is clocked by the falling edge of BUSCLKX4.

14.5.1 SIM Counter During Power-On Reset

The power-on reset module (POR) detects power applied to the MCU. At power-on, the POR circuit asserts the signal PORRST. Once the SIM is initialized, it enables the oscillator to drive the bus clock state machine.

14.5.2 SIM Counter During Stop Mode Recovery

The SIM counter also is used for stop mode recovery. The STOP instruction clears the SIM counter. After an interrupt, break, or reset, the SIM senses the state of the short stop recovery bit, SSREC, in the configuration register 1 (CONFIG1). If the SSREC bit is a 1, then the stop recovery is reduced from the normal delay of 4096 BUSCLKX4 cycles down to 32 BUSCLKX4 cycles. This is ideal for applications using canned oscillators that do not require long start-up times from stop mode. External crystal applications should use the full stop recovery time, that is, with SSREC cleared in the configuration register 1 (CONFIG1).

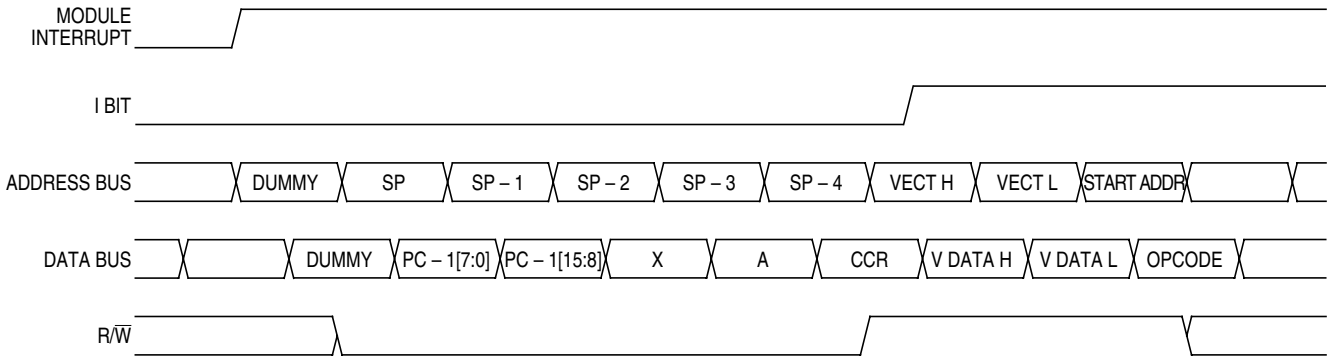


Figure 14-8. Interrupt Entry

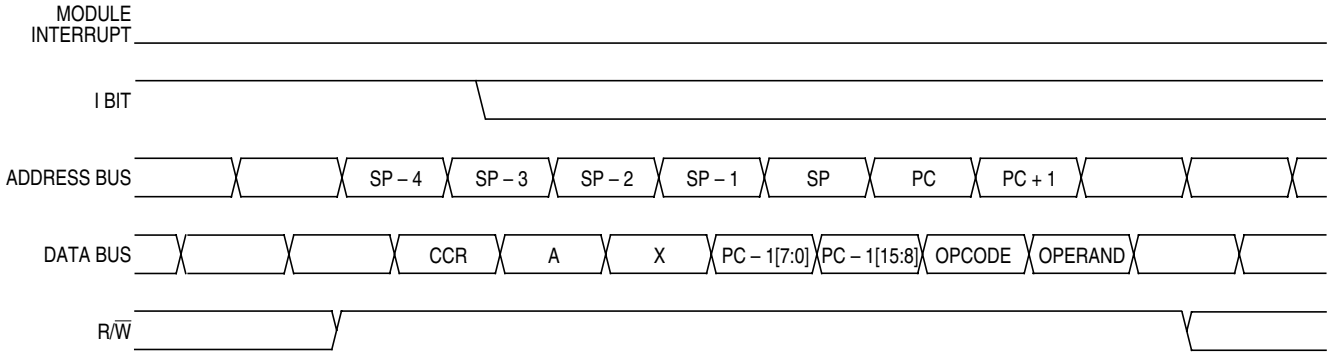


Figure 14-9. Interrupt Recovery

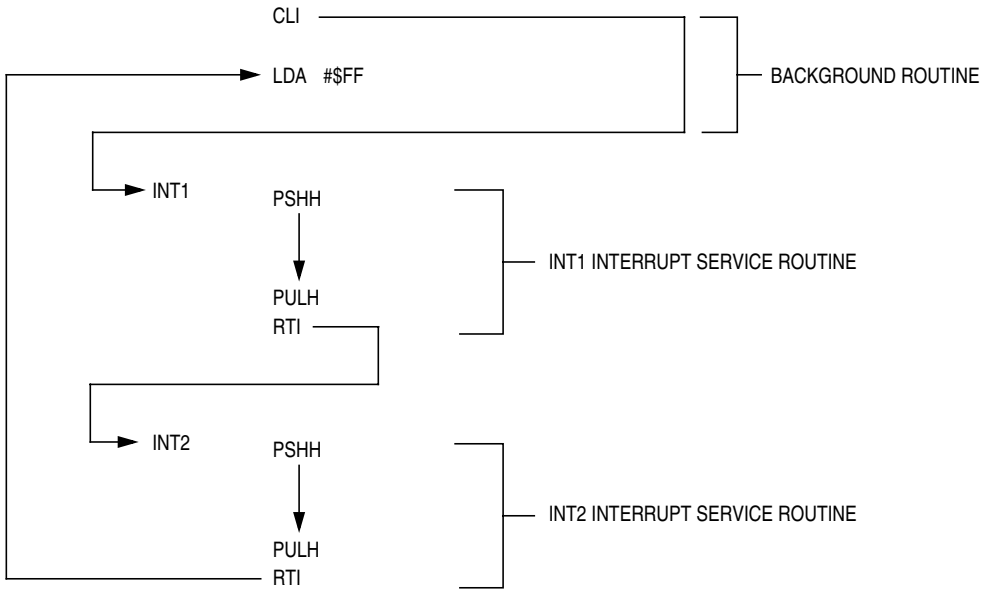
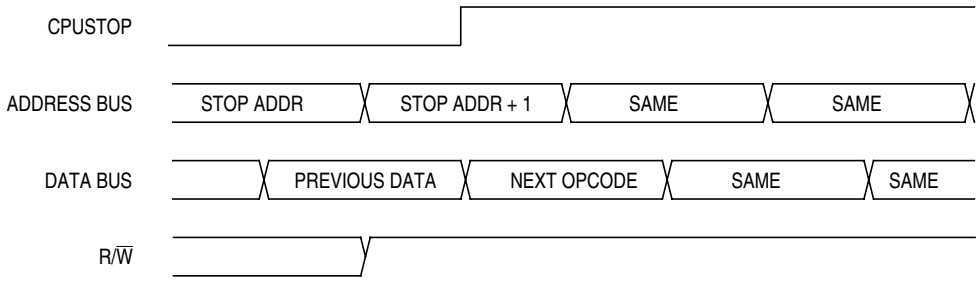


Figure 14-10. Interrupt Recognition Example

The SIM counter is held in reset from the execution of the STOP instruction until the beginning of stop recovery. It is then used to time the recovery period. Figure 14-17 shows stop mode entry timing and Figure 14-18 shows the stop mode recovery time from interrupt or break

NOTE

To minimize stop current, all pins configured as inputs should be driven to a logic 1 or logic 0.



NOTE: Previous data can be operand data or the STOP opcode, depending on the last instruction.

Figure 14-17. Stop Mode Entry Timing

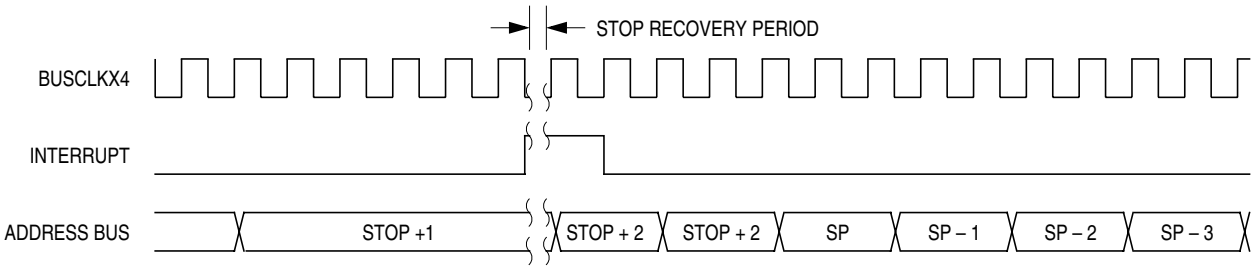


Figure 14-18. Stop Mode Recovery from Interrupt

14.8 SIM Registers

The SIM has three memory mapped registers. Table 14-4 shows the mapping of these registers.

Table 14-4. SIM Registers

Address	Register	Access Mode
$FE00	BSR	User
$FE01	SRSR	User
$FE03	BFCR	User

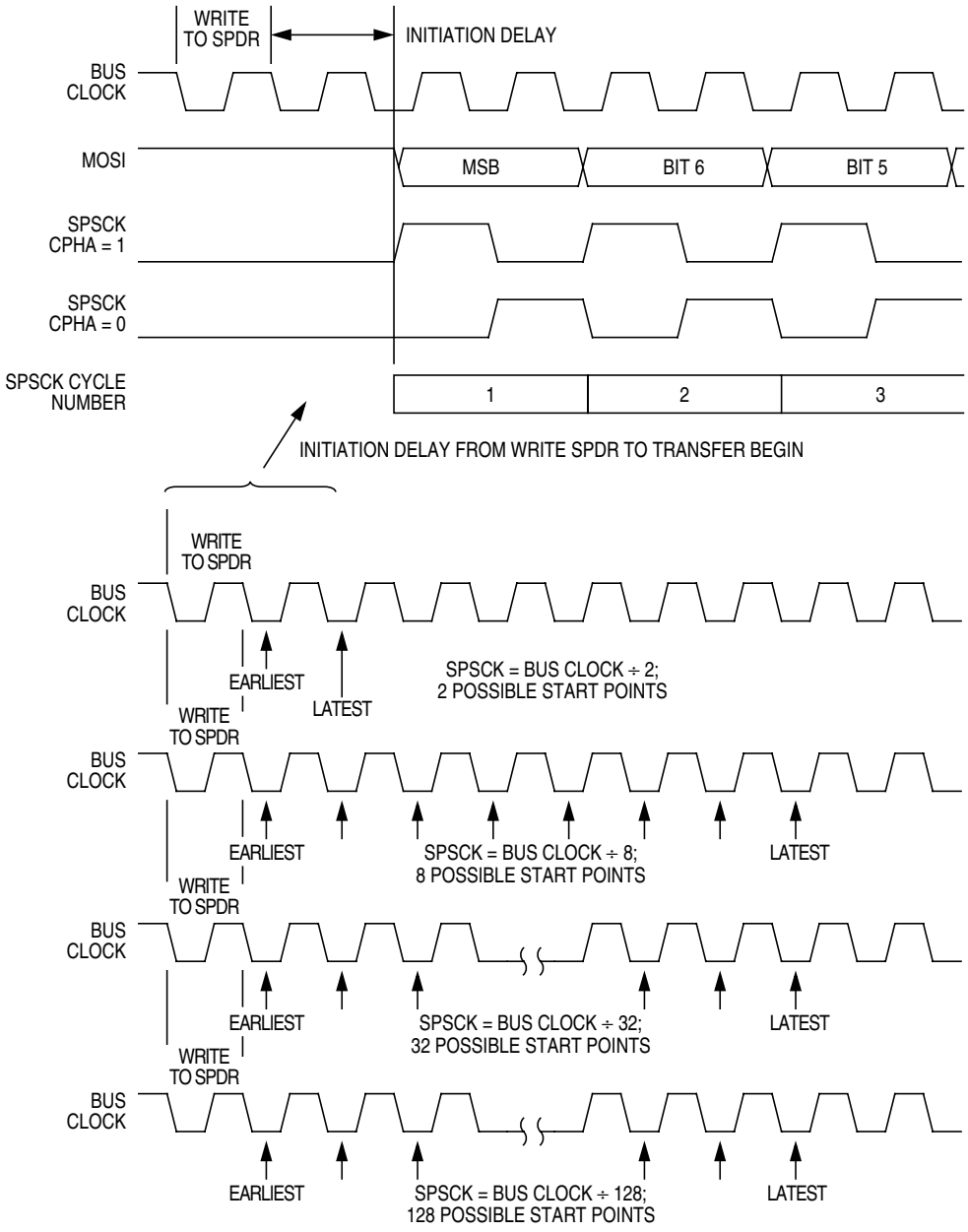


Figure 15-7. Transmission Start Delay (Master)

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	SPRIE	R	SPMSTR	CPOL	CPHA	SPWOM	SPE	SPTIE
Reset:	0	0	1	0	1	0	0	0
	R	= Reserved						

Figure 15-13. SPI Control Register (SPCR)

SPRIE — SPI Receiver Interrupt Enable Bit

This read/write bit enables interrupt requests generated by the SPRF bit. The SPRF bit is set when a byte transfers from the shift register to the receive data register.

1 = SPRF interrupt requests enabled
0 = SPRF interrupt requests disabled

SPMSTR — SPI Master Bit

This read/write bit selects master mode operation or slave mode operation.

1 = Master mode
0 = Slave mode

CPOL — Clock Polarity Bit

This read/write bit determines the logic state of the SPSCK pin between transmissions. (See Figure 15-4 and Figure 15-6.) To transmit data between SPI modules, the SPI modules must have identical CPOL values.

CPHA — Clock Phase Bit

This read/write bit controls the timing relationship between the serial clock and SPI data. (See Figure 15-4 and Figure 15-6.) To transmit data between SPI modules, the SPI modules must have identical CPHA values. When CPHA = 0, the $\overline{SS}$ pin of the slave SPI module must be high between bytes. (See Figure 15-12.)

SPWOM — SPI Wired-OR Mode Bit

This read/write bit configures pins SPSCK, MOSI, and MISO so that these pins become open-drain outputs.

1 = Wired-OR SPSCK, MOSI, and MISO pins
0 = Normal push-pull SPSCK, MOSI, and MISO pins

SPE — SPI Enable

This read/write bit enables the SPI module. Clearing SPE causes a partial reset of the SPI. (See 15.3.5 Resetting the SPI.)

1 = SPI module enabled
0 = SPI module disabled

SPTIE— SPI Transmit Interrupt Enable

This read/write bit enables interrupt requests generated by the SPTE bit. SPTE is set when a byte transfers from the transmit data register to the shift register.

1 = SPTE interrupt requests enabled
0 = SPTE interrupt requests disabled

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	TOF	TOIE	TSTOP	0	0	PS2	PS1	PS0
Write:	0			TRST				
Reset:	0	0	1	0	0	0	0	0

= Unimplemented

Figure 16-4. TIM1 Status and Control Register (T1SC)

TOF — TIM1 Overflow Flag Bit

This read/write flag is set when the counter reaches the modulo value programmed in the TIM1 counter modulo registers. Clear TOF by reading the T1SC register when TOF is set and then writing a 0 to TOF. If another TIM1 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 — TIM1 Overflow Interrupt Enable Bit

This read/write bit enables TIM1 overflow interrupts when the TOF bit becomes set.

1 = TIM1 overflow interrupts enabled
0 = TIM1 overflow interrupts disabled

TSTOP — TIM1 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 TIM1 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 — TIM1 Reset Bit

Setting this write-only bit resets the counter and the TIM1 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.

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 TIM2 channel x registers.

Clear CHxF by reading the T2SCx 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 TIM2 interrupt service requests on channel x.

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

MSxB — Mode Select Bit B

This read/write bit selects buffered output compare/PWM operation. MSxB exists only in the T2SC0.

Setting MS0B causes the contents of T2SC1 to be ignored by the TIM2 and reverts T2CH1 to general-purpose I/O.

1 = Buffered output compare/PWM operation enabled
0 = Buffered output compare/PWM operation disabled

MSxA — Mode Select Bit A

When ELSxB:A ≠ 00, this read/write bit selects either input capture operation or unbuffered output compare/PWM operation. See Table 17-2.

1 = Unbuffered output compare/PWM operation
0 = Input capture operation

When ELSxB:A = 00, this read/write bit selects the initial output level of the T2CHx pin (see Table 17-2).

1 = Initial output level low
0 = Initial output level high

NOTE

Before changing a channel function by writing to the MSxB or MSxA bit, set the TSTOP and TRST bits in the TIM2 status and control register (T2SC).

ELSxB and ELSxA — Edge/Level Select Bits

When channel x is an input capture channel, these read/write bits control the active edge-sensing logic on channel x.

When channel x is an output compare channel, ELSxB and ELSxA control the channel x output behavior when an output compare occurs.

When ELSxB and ELSxA are both clear, channel x is not connected to an I/O port, and pin T2CHx is available as a general-purpose I/O pin. Table 17-2 shows how ELSxB and ELSxA work.

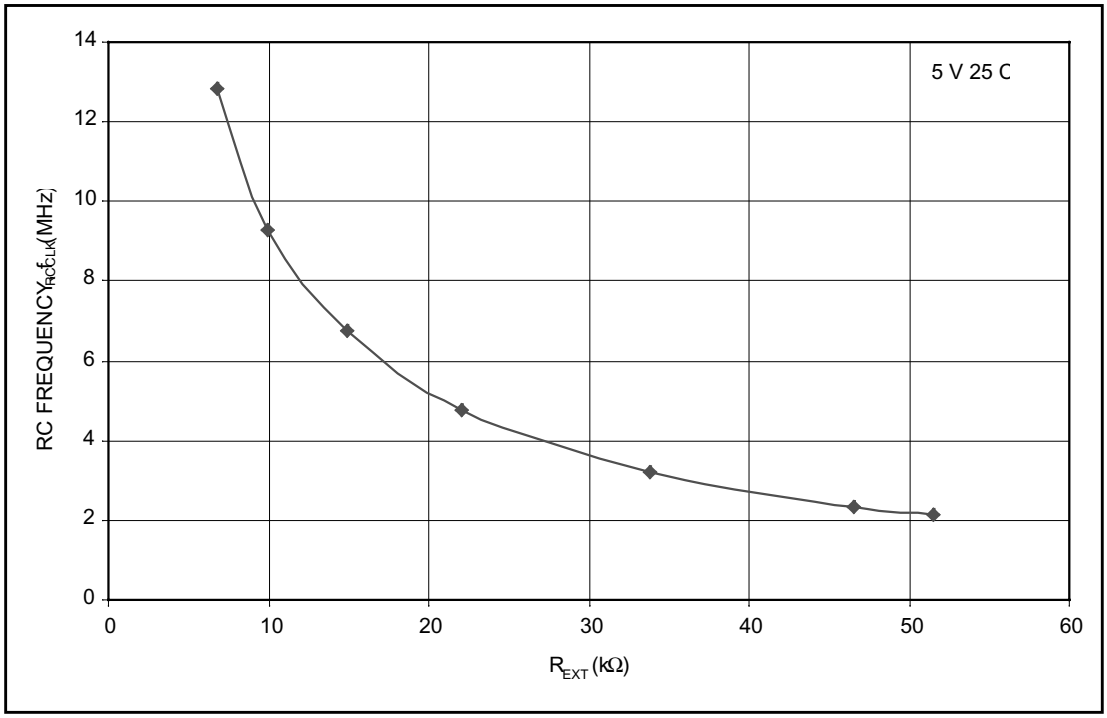


Figure 19-7. RC versus Frequency (5 Volts @ 25 C)

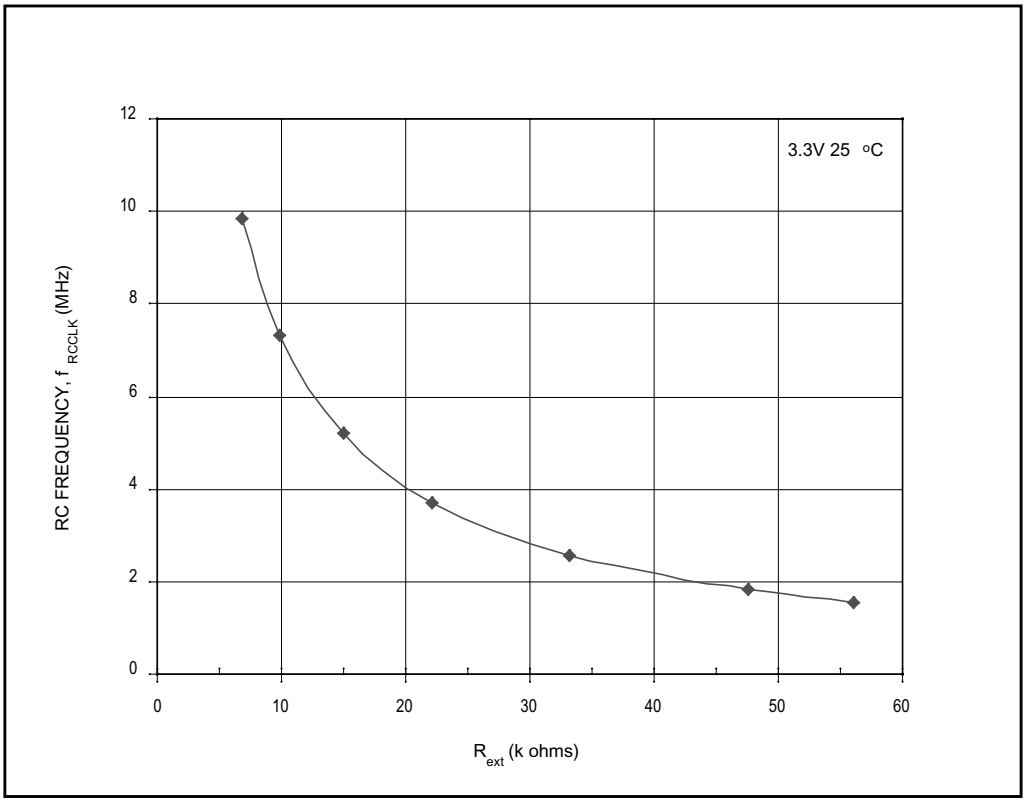


Figure 19-8. RC versus Frequency (3.3 Volts @ 25 C)

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}.

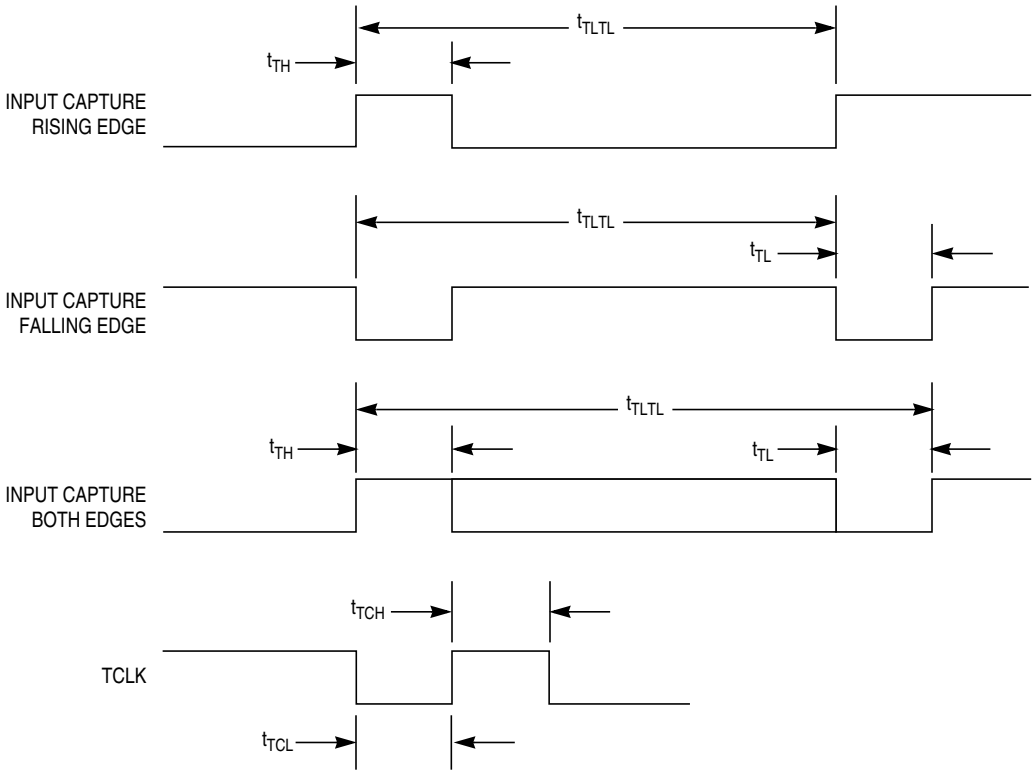


Figure 19-13. Timer Input Timing

NOTES:

1. DIMENSIONS AND TOLERANCING PER ASME Y14.5M–1994.

2. DIMENSIONS IN MILLIMETERS.

3. DIMENSION DOES NOT INCLUDE MOLD PROTRUSION. ALLOWABLE MOLD PROTRUSION IS 0.15 PER END.

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

5. DIMENSION DOES NOT INCLUDE DAM BAR PROTRUSIONS. DAM BAR PROTRUSION SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED 0.38.

MECHANICAL OUTLINE	PRINT VERSION NOT TO SCALE

TITLE:	DOCUMENT NO: 98ARS23923W	REV: C
28 LEAD 0.65 PITCH TSSOP 4.4 WIDE BODY, 1.2 MAX HEIGHT	CASE NUMBER: 1168-02	18 AUG 2006
	STANDARD: JEDEC MO-153 AE	