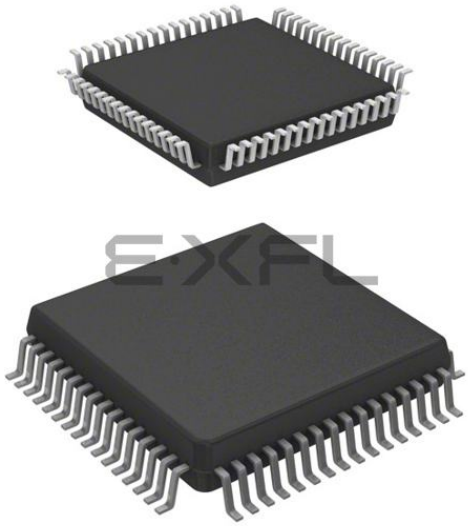


Welcome to E-XFL.COM

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	Not For New Designs
Core Processor	HC08
Core Size	8-Bit
Speed	8MHz
Connectivity	SCI, SPI
Peripherals	LVD, POR, PWM
Number of I/O	44
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	768 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 10x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-QFP
Supplier Device Package	64-QFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc908mr32cfue

18.3 Monitor ROM (MON) 255
18.3.1 Functional Description 256
18.3.1.1 Entering Monitor Mode 256
18.3.1.2 Normal Monitor Mode 256
18.3.1.3 Forced Monitor Mode 259
18.3.1.4 Data Format 259
18.3.1.5 Echoing 260
18.3.1.6 Break Signal 260
18.3.1.7 Commands 260
18.3.1.8 Baud Rate 263
18.3.2 Security 263

Chapter 19
Electrical Specifications

19.1 Introduction 265
19.2 Absolute Maximum Ratings 265
19.3 Functional Operating Range 266
19.4 Thermal Characteristics 266
19.5 DC Electrical Characteristics 267
19.6 FLASH Memory Characteristics 268
19.7 Control Timing 268
19.8 Serial Peripheral Interface Characteristics 269
19.9 TImer Interface Module Characteristics 272
19.10 Clock Generation Module Component Specifications 272
19.11 CGM Operating Conditions 272
19.12 CGM Acquisition/Lock Time Specifications 273
19.13 Analog-to-Digital Converter (ADC) Characteristics 274

Chapter 20
Ordering Information and Mechanical Specifications

20.1 Introduction 275
20.2 Order Numbers 275
20.3 64-Pin Plastic Quad Flat Pack (QFP) 276
20.4 56-Pin Shrink Dual In-Line Package (SDIP) 277

Appendix A
MC68HC908MR16

Table 2-1 is a list of vector locations.

Table 2-1. Vector Addresses

Low

Priority

Address	Vector
$FFD2	SCI transmit vector (high)
$FFD3	SCI transmit vector (low)
$FFD4	SCI receive vector (high)
$FFD5	SCI receive vector (low)
$FFD6	SCI error vector (high)
$FFD7	SCI error vector (low)
$FFD8	SPI transmit vector (high)(1)
$FFD9	SPI transmit vector (low)(1)
$FFDA	SPI receive vector (high)(1)
$FFDB	SPI receive vector (low)(1)
$FFDC	A/D vector (high)
$FFDD	A/D vector (low)
$FFDE	TIMB overflow vector (high)
$FFDF	TIMB overflow vector (low)
$FFE0	TIMB channel 1 vector (high)
$FFE1	TIMB channel 1 vector (low)
$FFE2	TIMB channel 0 vector (high)
$FFE3	TIMB channel 0 vector (low)
$FFE4	TIMA overflow vector (high)
$FFE5	TIMA overflow vector (low)
$FFE6	TIMA channel 3 vector (high)
$FFE7	TIMA channel 3 vector (low)
$FFE8	TIMA channel 2 vector (high)
$FFE9	TIMA channel 2 vector (low)
$FFEA	TIMA channel 1 vector (high)
$FFEB	TIMA channel 1 vector (low)
$FFEC	TIMA channel 0 vector (high)
$FFED	TIMA channel 0 vector (low)

1. The SPI module is not available in the 56-pin SDIP package.

During a subroutine call, the CPU uses two bytes of the stack to store the return address. The stack pointer decrements during pushes and increments during pulls.

NOTE

Be careful when using nested subroutines. The CPU may overwrite data in the RAM during a subroutine or during the interrupt stacking operation.

2.8 FLASH Memory (FLASH)

The FLASH memory is an array of 32,256 bytes with an additional 46 bytes of user vectors and one byte of block protection.

NOTE

An erased bit reads as a 1 and a programmed bit reads as a 0.

Program and erase operations are facilitated through control bits in a memory mapped register. Details for these operations appear later in this section.

Memory in the FLASH array is organized into two rows per page. The page size is 128 bytes per page. The minimum erase page size is 128 bytes. Programming is performed on a row basis, 64 bytes at a time.

The address ranges for the user memory and vectors are:

- $8000–$FDFF, user memory
- $FF7E, block protect register (FLBPR)
- $FE08, FLASH control register (FLCR)
- $FFD2–$FFFF, reserved for user-defined interrupt and reset vectors

Programming tools are available from Freescale. Contact a local Freescale representative for more information.

NOTE

A security feature prevents viewing of the FLASH contents.[1]

2.8.1 FLASH Control Register

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

Address: $FE08

	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)

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

3.7.1 ADC Status and Control Register

This section describes the function of the ADC status and control register (ADSCR). Writing ADSCR aborts the current conversion and initiates a new conversion.

Address: $0040

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	COCO	AIEN	ADCO	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0
Write:	R							
Reset:	0	0	0	1	1	1	1	1

R = Reserved

Figure 3-4. ADC Status and Control Register (ADSCR)

COCO — Conversions Complete Bit

In non-interrupt mode (AIEN = 0), COCO is a read-only bit that is set at the end of each conversion. COCO will stay set until cleared by a read of the ADC data register. Reset clears this bit.

In interrupt mode (AIEN = 1), COCO is a read-only bit that is not set at the end of a conversion. It always reads as a 0.

1 = Conversion completed (AIEN = 0)
0 = Conversion not completed (AIEN = 0) or CPU interrupt enabled (AIEN = 1)

NOTE

The write function of the COCO bit is reserved. When writing to the ADSCR register, always have a 0 in the COCO bit position.

AIEN — ADC Interrupt Enable Bit

When this bit is set, an interrupt is generated at the end of an ADC conversion. The interrupt signal is cleared when the data register is read or the status/control register is written. Reset clears the AIEN bit.

1 = ADC interrupt enabled
0 = ADC interrupt disabled

ADCO — ADC Continuous Conversion Bit

When set, the ADC will convert samples continuously and update the ADR register at the end of each conversion. Only one conversion is allowed when this bit is cleared. Reset clears the ADCO bit.

1 = Continuous ADC conversion
0 = One ADC conversion

ADCH[4:0] — ADC Channel Select Bits

ADCH4, ADCH3, ADCH2, ADCH1, and ADCH0 form a 5-bit field which is used to select one of 10 ADC channels. The ADC channels are detailed in Table 3-1.

NOTE

Take care to prevent switching noise from corrupting the analog signal when simultaneously using a port pin as both an analog and digital input.

The ADC subsystem is turned off when the channel select bits are all set to 1. This feature allows for reduced power consumption for the MCU when the ADC is not used.

NOTE

Recovery from the disabled state requires one conversion cycle to stabilize.

Chapter 5
Configuration Register (CONFIG)

5.1 Introduction

This section describes the configuration register (CONFIG). This register contains bits that configure these options:

- Resets caused by the low-voltage inhibit (LVI) module
- Power to the LVI module
- Computer operating properly (COP) module
- Top-side pulse-width modulator (PWM) polarity
- Bottom-side PWM polarity
- Edge-aligned versus center-aligned PWMs
- Six independent PWMs versus three complementary PWM pairs

5.2 Functional Description

The configuration register (CONFIG) is used in the initialization of various options. The configuration register can be written once after each reset. All of the configuration register bits are cleared during reset. Since the various options affect the operation of the microcontroller unit (MCU), it is recommended that this register be written immediately after reset. The configuration register is located at $001F and may be read at anytime.

NOTE

On a FLASH device, the options are one-time writeable by the user after each reset. The registers are not in the FLASH memory but are special registers containing one-time writeable latches after each reset. Upon a reset, the configuration register defaults to predetermined settings as shown in Figure 5-1.

If the LVI module and the LVI reset signal are enabled, a reset occurs when V_{DD} falls to a voltage, V_{LVRx}, and remains at or below that level for at least nine consecutive central processor unit (CPU) cycles. Once an LVI reset occurs, the MCU remains in reset until V_{DD} rises to a voltage, V_{LVRX}.

10.2 Port A

Port A is an 8-bit, general-purpose, bidirectional I/O port.

10.2.1 Port A Data Register

The port A data register (PTA) contains a data latch for each of the eight port A pins.

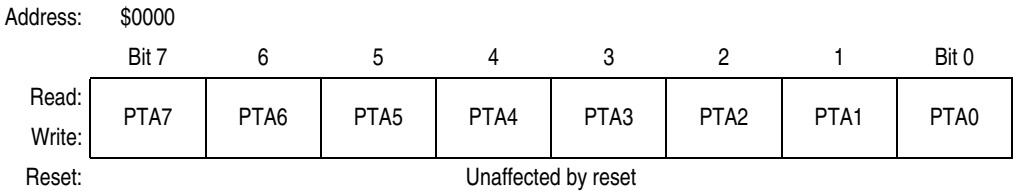


Figure 10-2. Port A Data Register (PTA)

PTA[7:0] — Port A Data Bits

These read/write bits are software programmable. Data direction of each port A pin is under the control of the corresponding bit in data direction register A. Reset has no effect on port A data.

10.2.2 Data Direction Register A

Data direction register A (DDRA) determines whether each port A pin is an input or an output. Writing a logic 1 to a DDRA bit enables the output buffer for the corresponding port A pin; a logic 0 disables the output buffer.

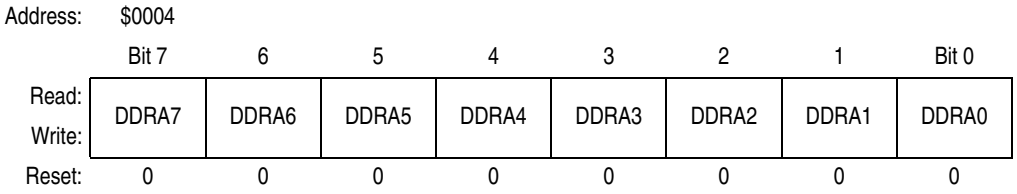


Figure 10-3. Data Direction Register A (DDRA)

DDRA[7:0] — Data Direction Register A Bits

These read/write bits control port A data direction. Reset clears DDRA[7:0], configuring all port A pins as inputs.

1 = Corresponding port A pin configured as output
0 = Corresponding port A pin configured as input

NOTE

Avoid glitches on port A pins by writing to the port A data register before changing data direction register A bits from 0 to 1.

Figure 10-4 shows the port A I/O logic.

Chapter 11
Power-On Reset (POR)

11.1 Introduction

This section describes the power-on reset (POR) module.

11.2 Functional Description

The POR module provides a known, stable signal to the microcontroller unit (MCU) at power-on. This signal tracks V_{DD} until the MCU generates a feedback signal to indicate that it is properly initialized. At this time, the POR drives its output low.

The POR is not a brown-out detector, low-voltage detector, or glitch detector. V_{DD} at the POR must go completely to 0 to reset the microcontroller unit (MCU). To detect power-loss conditions, use a low-voltage inhibit module (LVI) or other suitable circuit.

When complementary operation is used, two additional features are provided:

- Dead-time insertion
- Separate top/bottom pulse width correction to correct for distortions caused by the motor drive characteristics

If independent operation is chosen, each PWM has its own PWM value register.

12.5.2 Dead-Time Insertion

As shown in Figure 12-13, in complementary mode, each PWM pair can be used to drive top-side/bottom-side transistors.

When controlling dc-to-ac inverters such as this, the top and bottom PWMs in one pair should ***never*** be active at the same time. In Figure 12-13, if PWM1 and PWM2 were on at the same time, large currents would flow through the two transistors as they discharge the bus capacitor. The IGBTs could be weakened or destroyed.

Simply forcing the two PWMs to be inversions of each other is not always sufficient. Since a time delay is associated with turning off the transistors in the motor drive, there must be a dead-time between the deactivation of one PWM and the activation of the other.

A dead-time can be specified in the dead-time write-once register. This 8-bit value specifies the number of CPU clock cycles to use for the dead-time. The dead-time is not affected by changes in the PWM period caused by the prescaler.

Dead-time insertion is achieved by feeding the top PWM outputs of the PWM generator into dead-time generators, as shown in Figure 12-14. Current sensing determines which PWM value of a PWM generator pair to use for the top PWM in the next PWM cycle. See 12.5.3 Top/Bottom Correction with Motor Phase Current Polarity Sensing. When output control is enabled, the odd OUT bits, rather than the PWM generator outputs, are fed into the dead-time generators. See 12.5.5 PWM Output Port Control**.**

Whenever an input to a dead-time generator transitions, a dead-time is inserted (for example, both PWMs in the pair are forced to their inactive state). The bottom PWM signal is generated from the top PWM and the dead-time. In the case of output control enabled, the odd OUTx bits control the top PWMs, the even OUTx bits control the bottom PWMs *with respect to the odd OUTx bits* (see Table 12-6). Figure 12-15 shows the effects of the dead-time insertion.

As seen in Figure 12-15, some pulse width distortion occurs when the dead-time is inserted. The active pulse widths are reduced. For example, in Figure 12-15, when the PWM value register is equal to two, the ideal waveform (with no dead-time) has pulse widths equal to four. However, the actual pulse widths shrink to two after a dead-time of two was inserted. In this example, with the prescaler set to divide by one and center-aligned operation selected, this distortion can be compensated for by adding or subtracting half the dead-time value to or from the PWM register value. This correction is further described in 12.5.3 Top/Bottom Correction with Motor Phase Current Polarity Sensing.

Further examples of dead-time insertion are shown in Figure 12-16 and Figure 12-17. Figure 12-16 shows the effects of dead-time insertion at the duty cycle boundaries (near 0 percent and 100 percent duty cycles). Figure 12-17 shows the effects of dead-time insertion on pulse widths smaller than the dead-time.

12.5.5 PWM Output Port Control

Conditions may arise in which the PWM pins need to be individually controlled. This is made possible by the PWM output control register (PWMOUT) shown in Figure 12-22.

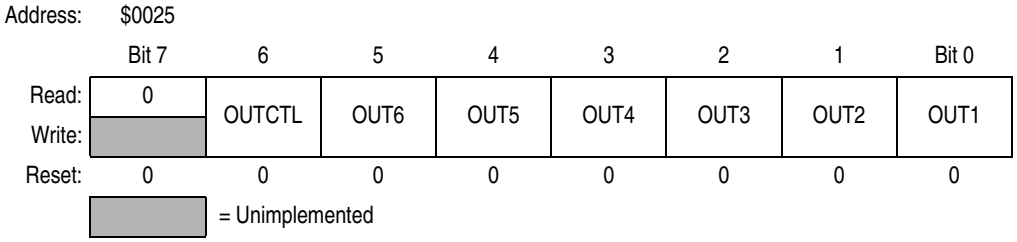


Figure 12-22. PWM Output Control Register (PWMOUT)

If the OUTCTL bit is set, the PWM pins can be controlled by the OUTx bits. These bits behave according to Table 12-6.

Table 12-6. OUTx Bits

OUTx Bit	Complementary Mode	Independent Mode
OUT1	1 — PWM1 is active. 0 — PWM1 is inactive.	1 — PWM1 is active. 0 — PWM1 is inactive.
OUT2	1 — PWM2 is complement of PWM 1. 0 — PWM2 is inactive.	1 — PWM2 is active. 0 — PWM2 is inactive.
OUT3	1 — PWM3 is active. 0 — PWM3 is inactive.	1 — PWM3 is active. 0 — PWM3 is inactive.
OUT4	1 — PWM4 is complement of PWM 3. 0 — PWM4 is inactive.	1 — PWM4 is active. 0 — PWM4 is inactive.
OUT5	1 — PWM5 is active. 0 — PWM5 is inactive.	1 — PWM5 is active. 0 — PWM5 is inactive.
OUT6	1 — PWM 6 is complement of PWM 5. 0 — PWM6 is inactive.	1 — PWM6 is active. 0 — PWM6 is inactive.

When OUTCTL is set, the polarity options TOPPOL and BOTPOL will still affect the outputs. In addition, if complementary operation is in use, the PWM pairs will not be allowed to be active simultaneously, and dead-time will still not be violated. When OUTCTL is set and complementary operation is in use, the odd OUTx bits are inputs to the dead-time generators as shown in Figure 12-15. Dead-time is inserted whenever the odd OUTx bit toggles as shown in Figure 12-23. Although dead-time is not inserted when the even OUTx bits change, there will be no dead-time violation as shown in Figure 12-24.

Setting the OUTCTL bit does not disable the PWM generator and current sensing circuitry. They continue to run, but are no longer controlling the output pins. In addition, OUTCTL will control the PWM pins even when PWMEN = 0. When OUTCTL is cleared, the outputs of the PWM generator become the inputs to the dead-time and output circuitry at the beginning of the next PWM cycle.

NOTE

To avoid an unexpected dead-time occurrence, it is recommended that the OUTx bits be cleared prior to entering and prior to exiting individual PWM output control mode.

ENSCI — Enable SCI Bit

This read/write bit enables the SCI and the SCI baud rate generator. Clearing ENSCI sets the SCTE and TC bits in SCI status register 1 and disables transmitter interrupts. Reset clears the ENSCI bit.

1 = SCI enabled
0 = SCI disabled

TXINV — Transmit Inversion Bit

This read/write bit reverses the polarity of transmitted data. Reset clears the TXINV bit.

1 = Transmitter output inverted
0 = Transmitter output not inverted

NOTE

Setting the TXINV bit inverts all transmitted values, including idle, break, start, and stop bits.

M — Mode (Character Length) Bit

This read/write bit determines whether SCI characters are eight or nine bits long. See Table 13-4. The ninth bit can serve as an extra stop bit, as a receiver wakeup signal, or as a parity bit. Reset clears the M bit.

1 = 9-bit SCI characters
0 = 8-bit SCI characters

WAKE — Wakeup Condition Bit

This read/write bit determines which condition wakes up the SCI: a 1 (address mark) in the most significant bit (MSB) position of a received character or an idle condition on the PTF4/RxD pin. Reset clears the WAKE bit.

1 = Address mark wakeup
0 = Idle line wakeup

ILTY — Idle Line Type Bit

This read/write bit determines when the SCI starts counting 1s as idle character bits. The counting begins either after the start bit or after the stop bit. If the count begins after the start bit, then a string of 1s preceding the stop bit may cause false recognition of an idle character. Beginning the count after the stop bit avoids false idle character recognition, but requires properly synchronized transmissions. Reset clears the ILTY bit.

1 = Idle character bit count begins after stop bit.
0 = Idle character bit count begins after start bit.

PEN — Parity Enable Bit

This read/write bit enables the SCI parity function. See Table 13-4. When enabled, the parity function inserts a parity bit in the most significant bit position. See Figure 13-4. Reset clears the PEN bit.

1 = Parity function enabled
0 = Parity function disabled

PTY — Parity Bit

This read/write bit determines whether the SCI generates and checks for odd parity or even parity. See Table 13-4. Reset clears the PTY bit.

1 = Odd parity
0 = Even parity

NOTE

Changing the PTY bit in the middle of a transmission or reception can generate a parity error.

signal on the $\overline{RST}$ or the $\overline{IRQ}$ pin. This prevents the COP from becoming disabled as a result of external noise. During a break state, V_{HI} on the $\overline{RST}$ pin disables the COP module.

14.3.2.3 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.

Because the MC68HC908MR32 has stop mode disabled, execution of the STOP instruction will cause an illegal opcode reset.

14.3.2.4 Illegal Address Reset

An opcode fetch from addresses other than FLASH or RAM addresses generates an illegal address reset (unimplemented locations within memory map). 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.

14.3.2.5 Forced Monitor Mode Entry Reset (MENRST)

The MENRST module monitors the reset vector fetches and will assert an internal reset if it detects that the reset vectors are erased ($FF). When the MCU comes out of reset, it is forced into monitor mode.

14.3.2.6 Low-Voltage Inhibit (LVI) Reset

The low-voltage inhibit (LVI) module asserts its output to the SIM when the V_{DD} voltage falls to the V_{LVRX} voltage and remains at or below that level for at least nine consecutive CPU cycles (see 19.5 DC Electrical Characteristics). The LVI bit in the SIM reset status register (SRSR) is set, and the external reset pin ($\overline{RST}$) is held low while the SIM counter counts out 4096 CGMXCLK cycles. Sixty-four CGMXCLK cycles later, the CPU is released from reset to allow the reset vector sequence to occur. The SIM actively pulls down the $\overline{RST}$ pin for all internal reset sources.

14.4 SIM Counter

The SIM counter is used by the power-on reset (POR) module 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 (COP) module. The SIM counter overflow supplies the clock for the COP module. The SIM counter is 13 bits long and is clocked by the falling edge of CGMXCLK.

14.4.1 SIM Counter During Power-On Reset

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

14.4.2 SIM Counter and Reset States

External reset has no effect on the SIM counter. The SIM counter is free-running after all reset states. For counter control and internal reset recovery sequences, see 14.3.2 Active Resets from Internal Sources.

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. Once 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.

15.5.3 Transmission Format When CPHA = 1

Figure 15-7 shows an SPI transmission in which CPHA is logic 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 since 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 at logic 0, 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 15.6.2 Mode Fault Error. When CPHA = 1, the master begins driving its MOSI pin on the first SPSCK edge. Therefore, the slave uses the first SPSCK edge as a start transmission signal. The $\overline{SS}$ pin can remain low between transmissions. This format may be preferable in systems having only one master and only one slave driving the MISO data line.

When CPHA = 1 for a slave, the first edge of the SPSCK 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. Once 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 first edge of SPSCK. Any data written after the first edge is stored in the transmit data register and transferred to the shift register after the current transmission.

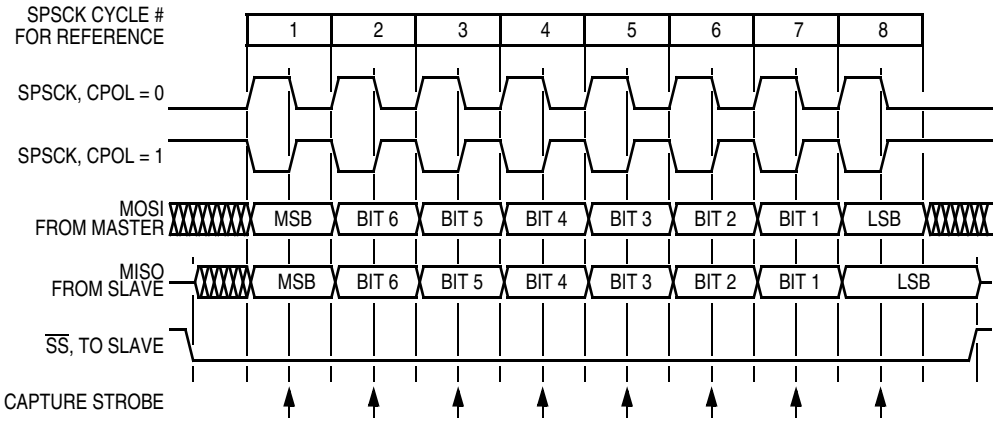


Figure 15-7. Transmission Format (CPHA = 1)

15.5.4 Transmission Initiation Latency

When the SPI is configured as a master (SPMSTR = 1), writing to the SPDR starts a transmission. CPHA has no effect on the delay to the start of the transmission, but it does affect the initial state of the SPSCK signal. When CPHA = 0, the SPSCK signal remains inactive for the first half of the first SPSCK cycle.

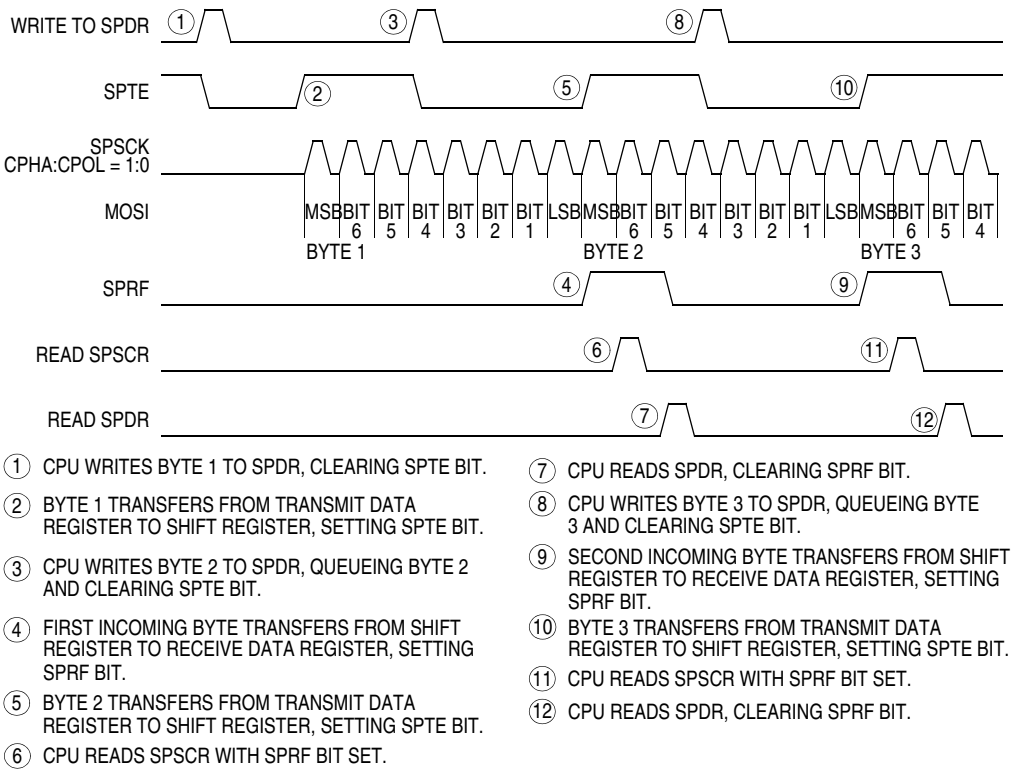


Figure 15-12. SPRF/SPTE CPU Interrupt Timing

15.10 Low-Power Mode

The WAIT instruction puts the MCU in a low power-consumption standby mode.

The SPI module remains active after the execution of a WAIT instruction. In wait mode the SPI module registers are not accessible by the CPU. Any enabled CPU interrupt request from the SPI module can bring the MCU out of wait mode.

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

To exit wait mode when an overflow condition occurs, enable the OVRF bit to generate CPU interrupt requests by setting the error interrupt enable bit (ERRIE). See 15.7 Interrupts.

Since the SPTE bit cannot be cleared during a break with the BCFE bit cleared, a write to the transmit data register in break mode does not initiate a transmission nor is this data transferred into the shift register. Therefore, a write to the SPDR in break mode with the BCFE bit cleared has no effect.

15.11 I/O Signals

The SPI module has five I/O pins and shares four of them with a parallel I/O port. The pins are:

- MISO — Data received
- MOSI — Data transmitted
- SPSCK — Serial clock
- $\overline{SS}$ — Slave select

16.7.3 TIMA Counter Modulo Registers

The read/write TIMA modulo registers contain the modulo value for the TIMA counter. When the TIMA counter reaches the modulo value, the overflow flag (TOF) becomes set, and the TIMA counter resumes counting from $0000 at the next timer clock. Writing to the high byte (TAMODH) inhibits the TOF bit and overflow interrupts until the low byte (TAMODL) is written. Reset sets the TIMA counter modulo registers.

Register Name and Address: TAMODH — $0011

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Reset:	1	1	1	1	1	1	1	1

Register Name and Address: TAMODL — $0012

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reset:	1	1	1	1	1	1	1	1

Figure 16-7. TIMA Counter Modulo Registers (TAMODH and TAMODL)

NOTE

Reset the TIMA counter before writing to the TIMA counter modulo registers.

16.7.4 TIMA Channel Status and Control Registers

Each of the TIMA channel status and control registers:

- Flags input captures and output compares
- Enables input capture and output compare interrupts
- Selects input capture, output compare, or PWM operation
- Selects high, low, or toggling output on output compare
- Selects rising edge, falling edge, or any edge as the active input capture trigger
- Selects output toggling on TIMA overflow
- Selects 0 percent and 100 percent PWM duty cycle
- Selects buffered or unbuffered output compare/PWM operation

Register Name and Address: TASC0 — $0013

	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

Register Name and Address: TASC1 — $0016

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

Register Name and Address: TASC2 — $0019

	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

Register Name and Address: TASC3 — $001C

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

R = Reserved

Figure 16-8. TIMA Channel Status and Control Registers (TASC0–TASC3)

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 TIMA counter registers matches the value in the TIMA channel x registers.

When CHxIE = 1, clear CHxF by reading TIMA channel x status and control 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.

Reset clears the CHxF bit. 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 TIMA CPU interrupts on channel x.

Reset clears the CHxIE bit.

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

The value in the TIMB channel registers determines the pulse width of the PWM output. The pulse width of an 8-bit PWM signal is variable in 256 increments. Writing $0080 (128) to the TIMB channel registers produces a duty cycle of 128/256 or 50 percent.

17.3.4.1 Unbuffered PWM Signal Generation

Any output compare channel can generate unbuffered PWM pulses as described in 17.3.4 Pulse-Width Modulation (PWM). The pulses are unbuffered because changing the pulse width requires writing the new pulse width value over the value currently in the TIMB channel registers.

An unsynchronized write to the TIMB channel registers to change a pulse width value could cause incorrect operation for up to two PWM periods. For example, writing a new value before the counter reaches the old value but after the counter reaches the new value prevents any compare during that PWM period. Also, using a TIMB overflow interrupt routine to write a new, smaller pulse width value may cause the compare to be missed. The TIMB may pass the new value before it is written to the TIMB channel registers.

Use this method to synchronize unbuffered changes in the PWM pulse width on channel x:

- When changing to a shorter pulse width, enable channel x output compare interrupts and write the new value in the output compare interrupt routine. The output compare interrupt occurs at the end of the current pulse. The interrupt routine has until the end of the PWM period to write the new value.
- When changing to a longer pulse width, enable TIMB overflow interrupts and write the new value in the TIMB overflow interrupt routine. The TIMB overflow interrupt occurs at the end of the current PWM period. Writing a larger value in an output compare interrupt routine (at the end of the current pulse) could cause two output compares to occur in the same PWM period.

NOTE

In PWM signal generation, do not program the PWM channel to toggle on output compare. Toggling on output compare prevents reliable 0 percent 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 also can cause incorrect PWM signal generation when changing the PWM pulse width to a new, much larger value.

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 PTE1/TCH0B pin. The TIMB channel registers of the linked pair alternately control the pulse width of the output.

Setting the MS0B bit in TIMB channel 0 status and control register (TBSC0) links channel 0 and channel 1. The TIMB channel 0 registers initially control the pulse width on the PTE1/TCH0B pin. Writing to the TIMB channel 1 registers enables the TIMB channel 1 registers to synchronously control the pulse width at the beginning of the next PWM period. At each subsequent overflow, the TIMB channel registers (0 or 1) that control the pulse width are the ones written to last. TBSC0 controls and monitors the buffered PWM function, and TIMB channel 1 status and control register (TBSC1) is unused. While the MS0B bit is set, the channel 1 pin, PTE2/TCH1B, 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

MSxB — Mode Select Bit B

This read/write bit selects buffered output compare/PWM operation. MSxB exists only in the TIMB channel 0.

Setting MS0B disables the channel 1 status and control register and reverts TCH1B to general-purpose I/O.
Reset clears the MSxB bit.

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 TCHx pin once PWM, input capture, or output compare operation is enabled. See Table 17-2. Reset clears the MSxA bit.

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 TIMB status and control register (TBSC).

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 port E, and pin PTEx/TCHxB is available as a general-purpose I/O pin. However, channel x is at a state determined by these bits and becomes transparent to the respective pin when PWM, input capture, or output compare mode is enabled. Table 17-2 shows how ELSxB and ELSxA work. Reset clears the ELSxB and ELSxA bits.

NOTE

Before enabling a TIMB channel register for input capture operation, make sure that the PTEx/TBCHx pin is stable for at least two bus clocks.

TOVx — Toggle-On-Overflow Bit

When channel x is an output compare channel, this read/write bit controls the behavior of the channel x output when the TIMB counter overflows. When channel x is an input capture channel, TOVx has no effect. Reset clears the TOVx bit.

1 = Channel x pin toggles on TIMB counter overflow.
0 = Channel x pin does not toggle on TIMB counter overflow.

18.2.1.2 CPU During Break Interrupts

The CPU starts a break interrupt by:

- Loading the instruction register with the SWI instruction
- Loading the program counter with $FFFC and $FFFD ($FEFC and $FEFD in monitor mode)

The break interrupt begins after completion of the CPU instruction in progress. If the break address register match occurs on the last cycle of a CPU instruction, the break interrupt begins immediately.

18.2.1.3 TIM1 and TIM2 During Break Interrupts

A break interrupt stops the timer counters.

18.2.1.4 COP During Break Interrupts

The COP is disabled during a break interrupt when V_{TST} is present on the $\overline{RST}$ pin.

18.2.2 Low-Power Modes

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

18.2.2.1 Wait Mode

If enabled, the break module is active in wait mode. In the break routine, the user can subtract one from the return address on the stack if SBSW is set. Clear the BW bit by writing logic 0 to it.

18.2.2.2 Stop Mode

The break module is inactive in stop mode. The STOP instruction does not affect break module register states.

18.2.3 Break Module Registers

These registers control and monitor operation of the break module:

- Break status and control register (BRKSCR)
- Break address register high (BRKH)
- Break address register low (BRKL)
- SIM break status register (SBSR)
- SIM break flag control register (SBFCR)

18.3.1.8 Baud Rate

With a 4.9152-MHz crystal and the PTC2 pin at logic 1 during reset, data is transferred between the monitor and host at 4800 baud. If the PTC2 pin is at logic 0 during reset, the monitor baud rate is 9600. See Table 18-9.

Table 18-9. Monitor Baud Rate Selection

	VCO Frequency Multiplier (N)					
	1	2	3	4	5	6
Monitor baud rate	4800	9600	14,400	19,200	24,000	28,800

18.3.2 Security

A security feature discourages unauthorized reading of FLASH locations while in monitor mode. The host can bypass the security feature at monitor mode entry by sending eight security bytes that match the bytes at locations $FFF6–$FFFD. Locations $FFF6–$FFFD contain user-defined data.

NOTE

Do not leave locations $FFF6–$FFFD blank. For security reasons, program locations $FFF6–$FFFD even if they are not used for vectors.

During monitor mode entry, the MCU waits after the power-on reset for the host to send the eight security bytes on pin PTA0. If the received bytes match those at locations $FFF6–$FFFD, the host bypasses the security feature and can read all FLASH locations and execute code from FLASH. Security remains bypassed until a power-on reset occurs. If the reset was not a power-on reset, security remains bypassed and security code entry is not required. (See Figure 18-13.)

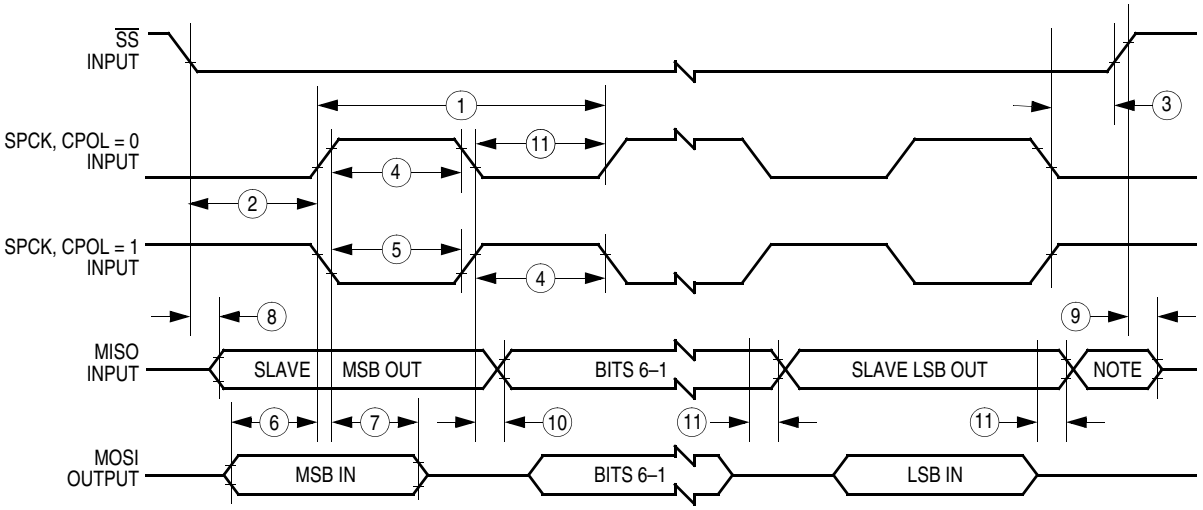
Upon power-on reset, if the received bytes of the security code do not match the data at locations $FFF6–$FFFD, the host fails to bypass the security feature. The MCU remains in monitor mode, but reading a FLASH location returns an invalid value and trying to execute code from FLASH causes an illegal address reset. After receiving the eight security bytes from the host, the MCU transmits a break character, signifying that it is ready to receive a command.

NOTE

The MCU does not transmit a break character until after the host sends the eight security bytes.

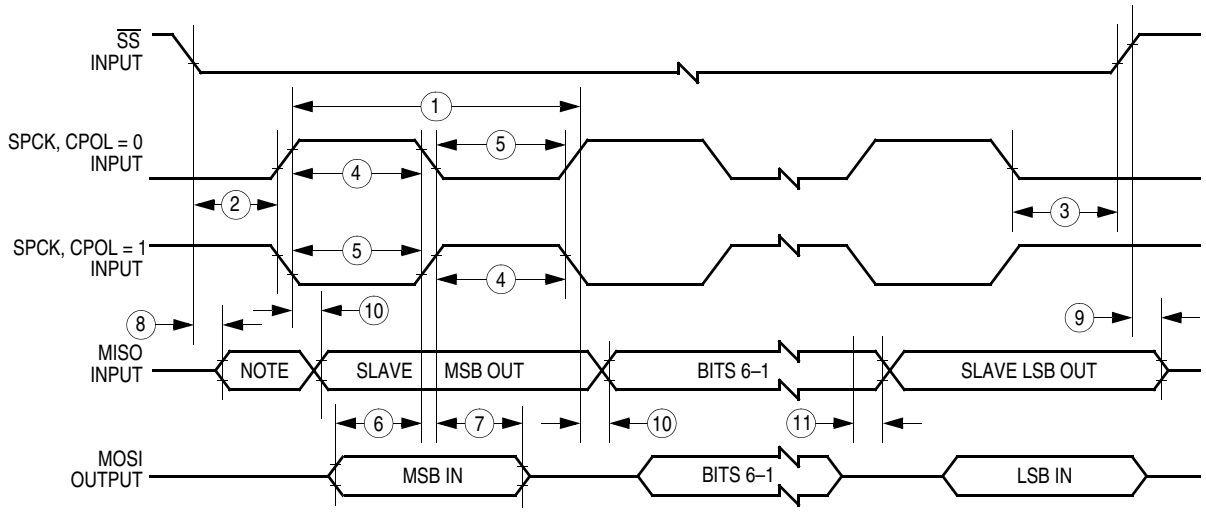
To determine whether the security code entered is correct, check to see if bit 6 of RAM address $60 is set. If it is, then the correct security code has been entered and FLASH can be accessed.

If the security sequence fails, the device can be reset (via power-pin reset only) and brought up in monitor mode to attempt another entry. After failing the security sequence, the FLASH mode can also be bulk erased by executing an erase routine that was downloaded into internal RAM. The bulk erase operation clears the security code locations so that all eight security bytes become $FF.



Note: Not defined, but normally MSB of character just received

a) SPI Slave Timing (CPHA = 0)



Note: Not defined, but normally LSB of character previously transmitted

b) SPI Slave Timing (CPHA = 1)

Figure 19-2. SPI Slave Timing