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	36
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	Through Hole
Package / Case	56-SDIP (0.600", 15.24mm)
Supplier Device Package	56-PSDIP
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc908mr32cbe

Chapter 10
Input/Output (I/O) Ports (PORTS)

10.1 Introduction . . . 101
10.2 Port A . . . 103
10.2.1 Port A Data Register . . . 103
10.2.2 Data Direction Register A . . . 103
10.3 Port B . . . 104
10.3.1 Port B Data Register . . . 104
10.3.2 Data Direction Register B . . . 105
10.4 Port C . . . 106
10.4.1 Port C Data Register . . . 106
10.4.2 Data Direction Register C . . . 106
10.5 Port D . . . 107
10.6 Port E . . . 108
10.6.1 Port E Data Register . . . 108
10.6.2 Data Direction Register E . . . 109
10.7 Port F . . . 110
10.7.1 Port F Data Register . . . 110
10.7.2 Data Direction Register F . . . 110

Chapter 11
Power-On Reset (POR)

11.1 Introduction . . . 113
11.2 Functional Description . . . 113

Chapter 12
Pulse-Width Modulator for Motor Control (PWMMC)

12.1 Introduction . . . 115
12.2 Features . . . 115
12.3 Timebase . . . 120
12.3.1 Resolution . . . 120
12.3.2 Prescaler . . . 122
12.4 PWM Generators . . . 122
12.4.1 Load Operation . . . 122
12.4.2 PWM Data Overflow and Underflow Conditions . . . 125
12.5 Output Control . . . 126
12.5.1 Selecting Six Independent PWMs or Three Complementary PWM Pairs . . . 126
12.5.2 Dead-Time Insertion . . . 127
12.5.3 Top/Bottom Correction with Motor Phase Current Polarity Sensing . . . 130
12.5.4 Output Polarity . . . 133
12.5.5 PWM Output Port Control . . . 135
12.6 Fault Protection . . . 137
12.6.1 Fault Condition Input Pins . . . 137
12.6.1.1 Fault Pin Filter . . . 139
12.6.1.2 Automatic Mode . . . 139
12.6.1.3 Manual Mode . . . 140

13.7.3 SCI Control Register 3 . . . 173
13.7.4 SCI Status Register 1 . . . 174
13.7.5 SCI Status Register 2 . . . 176
13.7.6 SCI Data Register . . . 177
13.7.7 SCI Baud Rate Register . . . 177

Chapter 14
System Integration Module (SIM)

14.1 Introduction . . . 181
14.2 SIM Bus Clock Control and Generation . . . 182
14.2.1 Bus Timing . . . 182
14.2.2 Clock Startup from POR or LVI Reset . . . 182
14.2.3 Clocks in Wait Mode . . . 183
14.3 Reset and System Initialization . . . 183
14.3.1 External Pin Reset . . . 183
14.3.2 Active Resets from Internal Sources . . . 184
14.3.2.1 Power-On Reset (POR) . . . 185
14.3.2.2 Computer Operating Properly (COP) Reset . . . 185
14.3.2.3 Illegal Opcode Reset . . . 186
14.3.2.4 Illegal Address Reset . . . 186
14.3.2.5 Forced Monitor Mode Entry Reset (MENRST) . . . 186
14.3.2.6 Low-Voltage Inhibit (LVI) Reset . . . 186
14.4 SIM Counter . . . 186
14.4.1 SIM Counter During Power-On Reset . . . 186
14.4.2 SIM Counter and Reset States . . . 186
14.5 Exception Control . . . 187
14.5.1 Interrupts . . . 187
14.5.1.1 Hardware Interrupts . . . 189
14.5.1.2 Software Interrupt (SWI) Instruction . . . 190
14.5.2 Reset . . . 190
14.6 Low-Power Mode . . . 190
14.6.1 Wait Mode . . . 190
14.6.2 Stop Mode . . . 191
14.7 SIM Registers . . . 191
14.7.1 SIM Break Status Register . . . 191
14.7.2 SIM Reset Status Register . . . 192
14.7.3 SIM Break Flag Control Register . . . 193

Chapter 15
Serial Peripheral Interface Module (SPI)

15.1 Introduction . . . 195
15.2 Features . . . 195
15.3 Pin Name Conventions . . . 195
15.4 Functional Description . . . 197
15.4.1 Master Mode . . . 198
15.4.2 Slave Mode . . . 199
15.5 Transmission Formats . . . 199

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0026	PWM Counter Register High (PCNTH) See page 143.	Read:	0	0	0	0	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0027	PWM Counter Register Low (PCNTL) See page 143.	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
$0028	PWM Counter Modulo Register High (PMODH) See page 144.	Read:	0	0	0	0	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	0	0	0	0	X	X	X	X
$0029	PWM Counter Modulo Register Low (PMODL) See page 144.	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	X	X	X	X	X	X	X	X
$002A	PWM 1 Value Register High (PVAL1H) See page 145.	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
$002B	PWM 1 Value Register Low (PVAL1L) See page 145.	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
$002C	PWM 2 Value Register High (PVAL2H) See page 145.	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
$002D	PWM 2 Value Register Low (PVAL2L) See page 145.	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
$002E	PWM 3 Value Register High (PVAL3H) See page 145.	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
$002F	PWM 3 Value Register Low (PVAL3L) See page 145.	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
$0030	PWM 4 Value Register High (PVAL4H) See page 145.	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
$0031	PWM 4 Value Register Low (PVAL4L) See page 145.	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

U = Unaffected X = Indeterminate R = Reserved **Bold** = Buffered = Unimplemented

Figure 2-2. Control, Status, and Data Registers Summary (Sheet 4 of 8)

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.

2.8.4 FLASH Program Operation

Use the following step-by-step procedure to program a row of FLASH memory. Figure 2-4 shows a flowchart of the programming algorithm.

NOTE

Only bytes which are currently $FF may be programmed.

1. Set the PGM bit. This configures the memory for program operation and enables the latching of address and data for programming.
2. Read the FLASH block protect register.
3. Write any data to any FLASH location within the address range desired.
4. Wait for a time, t_{NVS} (minimum 10 µs).
5. Set the HVEN bit.
6. Wait for a time, t_{PGS} (minimum 5 µs).
7. Write data to the FLASH address being programmed(1).
8. Wait for time, t_{PROG} (minimum 30 µs).
9. Repeat step 7 and 8 until all desired bytes within the row are programmed.
10. Clear the PGM bit(1).
11. Wait for time, t_{NVH} (minimum 5 µs).
12. Clear the HVEN bit.
13. After time, t_{RCV} (typical 1 µs), the memory can be accessed in read mode again.

NOTE

The COP register at location $FFFF should not be written between steps 5-12, when the HVEN bit is set. Since this register is located at a valid FLASH address, unpredictable behavior may occur if this location is written while HVEN is set.

This program sequence is repeated throughout the memory until all data is programmed.

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 shown, other unrelated operations may occur between the steps. Do not exceed t_{PROG} maximum, see 19.6 FLASH Memory Characteristics.

1. The time between each FLASH address change, or the time between the last FLASH address programmed to clearing PGM bit, must not exceed the maximum programming time, t_{PROG} maximum.

4.4.6 Crystal Output Frequency Signal (CGMXCLK)

CGMXCLK is the crystal oscillator output signal. It runs at the full speed of the crystal (f_{XCLK}) and comes directly from the crystal oscillator circuit. Figure 4-3 shows only the logical relation of CGMXCLK to OSC1 and OSC2 and may not represent the actual circuitry. The duty cycle of CGMXCLK is unknown and may depend on the crystal and other external factors. Also, the frequency and amplitude of CGMXCLK can be unstable at startup.

4.4.7 CGM Base Clock Output (CGMOUT)

CGMOUT is the clock output of the CGM. This signal goes to the SIM, which generates the MCU clocks. CGMOUT is a 50 percent duty cycle clock running at twice the bus frequency. CGMOUT is software programmable to be either the oscillator output, CGMXCLK, divided by two or the VCO clock, CGMVCLK, divided by two.

4.4.8 CGM CPU Interrupt (CGMINT)

CGMINT is the interrupt signal generated by the PLL lock detector.

4.5 CGM Registers

These registers control and monitor operation of the CGM:

- PLL control register (PCTL) — see 4.5.1 PLL Control Register
- PLL bandwidth control register (PBWC) — see 4.5.2 PLL Bandwidth Control Register
- PLL programming register (PPG) — see 4.5.3 PLL Programming Register

Figure 4-4 is a summary of the CGM registers.

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$005C	PLL Control Register (PCTL) See page 66.	Read:	PLLIE	PLLF	PLLON	BCS	1	1	1	1
		Write:		R			R	R	R	R
		Reset:	0	0	1	0	1	1	1	1
$005D	PLL Bandwidth Control Register (PBWC) See page 67.	Read:	AUTO	LOCK	$\overline{ACQ}$	XLD	0	0	0	0
		Write:		R			R	R	R	R
		Reset:	0	0	0	0	0	0	0	0
$005E	PLL Programming Register (PPG) See page 68.	Read:	MUL7	MUL6	MUL5	MUL4	VRS7	VRS6	VRS5	VRS4
		Write:								
		Reset:	0	1	1	0	0	1	1	0
			R	= Reserved						

Notes:
1. When AUTO = 0, PLLIE is forced to logic 0 and is read-only.
2. When AUTO = 0, PLLF and LOCK read as logic 0.
3. When AUTO = 1, $\overline{ACQ}$ is read-only.
4. When PLLON = 0 or VRS[7:4] = $0, BCS is forced to logic 0 and is read-only.
5. When PLLON = 1, the PLL programming register is read-only.
6. When BCS = 1, PLLON is forced set and is read-only.

Figure 4-4. CGM I/O Register Summary

Chapter 7
Central Processor Unit (CPU)

7.1 Introduction

The M68HC08 CPU (central processor unit) is an enhanced and fully object-code-compatible version of the M68HC05 CPU. The *CPU08 Reference Manual* (document order number CPU08RM/AD) contains a description of the CPU instruction set, addressing modes, and architecture.

7.2 Features

Features of the CPU include:

- Object code fully upward-compatible with M68HC05 Family
- 16-bit stack pointer with stack manipulation instructions
- 16-bit index register with x-register manipulation instructions
- 8-MHz CPU internal bus frequency
- 64-Kbyte program/data memory space
- 16 addressing modes
- Memory-to-memory data moves without using accumulator
- Fast 8-bit by 8-bit multiply and 16-bit by 8-bit divide instructions
- Enhanced binary-coded decimal (BCD) data handling
- Modular architecture with expandable internal bus definition for extension of addressing range beyond 64 Kbytes
- Low-power stop and wait modes

7.3 CPU Registers

Figure 7-1 shows the five CPU registers. CPU registers are not part of the memory map.

7.3.5 Condition Code Register

The 8-bit condition code register contains the interrupt mask and five flags that indicate the results of the instruction just executed. Bits 6 and 5 are set permanently to 1. The following paragraphs describe the functions of the condition code register.

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	V	1	1	H	I	N	Z	C
Reset:	X	1	1	X	1	X	X	X

X = Indeterminate

Figure 7-6. Condition Code Register (CCR)

V — Overflow Flag

The CPU sets the overflow flag when a two's complement overflow occurs. The signed branch instructions BGT, BGE, BLE, and BLT use the overflow flag.

1 = Overflow
0 = No overflow

H — Half-Carry Flag

The CPU sets the half-carry flag when a carry occurs between accumulator bits 3 and 4 during an add-without-carry (ADD) or add-with-carry (ADC) operation. The half-carry flag is required for binary-coded decimal (BCD) arithmetic operations. The DAA instruction uses the states of the H and C flags to determine the appropriate correction factor.

1 = Carry between bits 3 and 4
0 = No carry between bits 3 and 4

I — Interrupt Mask

When the interrupt mask is set, all maskable CPU interrupts are disabled. CPU interrupts are enabled when the interrupt mask is cleared. When a CPU interrupt occurs, the interrupt mask is set automatically after the CPU registers are saved on the stack, but before the interrupt vector is fetched.

1 = Interrupts disabled
0 = Interrupts enabled

NOTE

To maintain M6805 Family compatibility, the upper byte of the index register (H) is not stacked automatically. If the interrupt service routine modifies H, then the user must stack and unstack H using the PSHH and PULH instructions.

After the I bit is cleared, the highest-priority interrupt request is serviced first.

A return-from-interrupt (RTI) instruction pulls the CPU registers from the stack and restores the interrupt mask from the stack. After any reset, the interrupt mask is set and can be cleared only by the clear interrupt mask software instruction (CLI).

N — Negative Flag

The CPU sets the negative flag when an arithmetic operation, logic operation, or data manipulation produces a negative result, setting bit 7 of the result.

1 = Negative result
0 = Non-negative result

Once an LVI reset occurs, the MCU remains in reset until V_{DD} rises above a voltage, V_{LVRX} + V_{LVHX}. V_{DD} must be above V_{LVRX} + V_{LVHX} for only one CPU cycle to bring the MCU out of reset. See 14.3.2.6 Low-Voltage Inhibit (LVI) Reset. The output of the comparator controls the state of the LVIOUT flag in the LVI status register (LVISCR).

An LVI reset also drives the $\overline{RST}$ pin low to provide low-voltage protection to external peripheral devices. See 19.5 DC Electrical Characteristics.

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$FE0F	LVI Status and Control Register (LVISCR) See page 99.	Read:	LVIOUT	0	TRPSEL	0	0	0	0	0
		Write:	R	R		R	R	R	R	R
		Reset:	0	0	0	0	0	0	0	0

R = Reserved

Figure 9-2. LVI I/O Register Summary

9.3.1 Polled LVI Operation

In applications that can operate at V_{DD} levels below V_{LVRX}, software can monitor V_{DD} by polling the LVIOUT bit. In the configuration register, the LVIPWR bit must be 1 to enable the LVI module, and the LVIRST bit must be 0 to disable LVI resets. See Chapter 5 Configuration Register (CONFIG). TRPSEL in the LVISCR selects V_{LVRX}.

9.3.2 Forced Reset Operation

In applications that require V_{DD} to remain above V_{LVRX}, enabling LVI resets allows the LVI module to reset the MCU when V_{DD} falls to the V_{LVRX} level and remains at or below that level for nine or more consecutive CPU cycles. In the CONFIG register, the LVIPWR and LVIRST bits must be 1s to enable the LVI module and to enable LVI resets. TRPSEL in the LVISCR selects V_{LVRX}.

9.3.3 False Reset Protection

The V_{DD} pin level is digitally filtered to reduce false resets due to power supply noise. In order for the LVI module to reset the MCU, V_{DD} must remain at or below V_{LVRX} for nine or more consecutive CPU cycles. V_{DD} must be above V_{LVRX} + V_{LVHX} for only one CPU cycle to bring the MCU out of reset. TRPSEL in the LVISCR selects V_{LVRX} + V_{LVHX}.

9.3.4 LVI Trip Selection

The TRPSEL bit allows the user to chose between 5 percent and 10 percent tolerance when monitoring the supply voltage. The 10 percent option is enabled out of reset. Writing a 1 to TRPSEL will enable 5 percent option.

NOTE

The microcontroller is guaranteed to operate at a minimum supply voltage. The trip point (VLVR1 or VLVR2) may be lower than this. See 19.5 DC Electrical Characteristics.

9.4 LVI Status and Control Register

The LVI status register (LVISCR) flags V_{DD} voltages below the V_{LVRX} level.

Address: $FE0F

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

R = Reserved

Figure 9-3. LVI Status and Control Register (LVISCR)

LVIOUT — LVI Output Bit

This read-only flag becomes set when the V_{DD} voltage falls below the V_{LVRX} voltage for 32 to 40 CGMXCLK cycles. See Table 9-1. Reset clears the LVIOUT bit.

Table 9-1. LVIOUT Bit Indication

V_{DD}		LVIOUT
At Level:	**For Number of CGMXCLK Cycles:**	
$V_{DD} > V_{LVRX} + V_{LVHX}$	Any	0
$V_{DD} < V_{LVRX}$	< 32 CGMXCLK cycles	0
$V_{DD} < V_{LVRX}$	Between 32 & 40 CGMXCLK cycles	0 or 1
$V_{DD} < V_{LVRX}$	> 40 CGMXCLK cycles	1
$V_{LVRX} < V_{DD} < V_{LVRX} + V_{LVHX}$	Any	Previous value

TRPSEL — LVI Trip Select Bit

This bit selects the LVI trip point. Reset clears this bit.

1 = 5 percent tolerance. The trip point and recovery point are determined by V_{LVR1} and V_{LVH1}, respectively.

0 = 10 percent tolerance. The trip point and recovery point are determined by V_{LVR2} and V_{LVH2}, respectively.

NOTE

If LVIRST and LVIPWR are 0s, note that when changing the tolerance, LVI reset will be generated if the supply voltage is below the trip point.

9.5 LVI Interrupts

The LVI module does not generate interrupt requests.

9.6 Wait Mode

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

With the LVIPWR bit in the configuration register programmed to 1, the LVI module is active after a WAIT instruction.

With the LVIRST bit in the configuration register programmed to 1, the LVI module can generate a reset and bring the MCU out of wait mode.

9.7 Stop Mode

If enabled, the LVI module remains active in stop mode. If enabled to generate resets, the LVI module can generate a reset and bring the MCU out of stop mode.

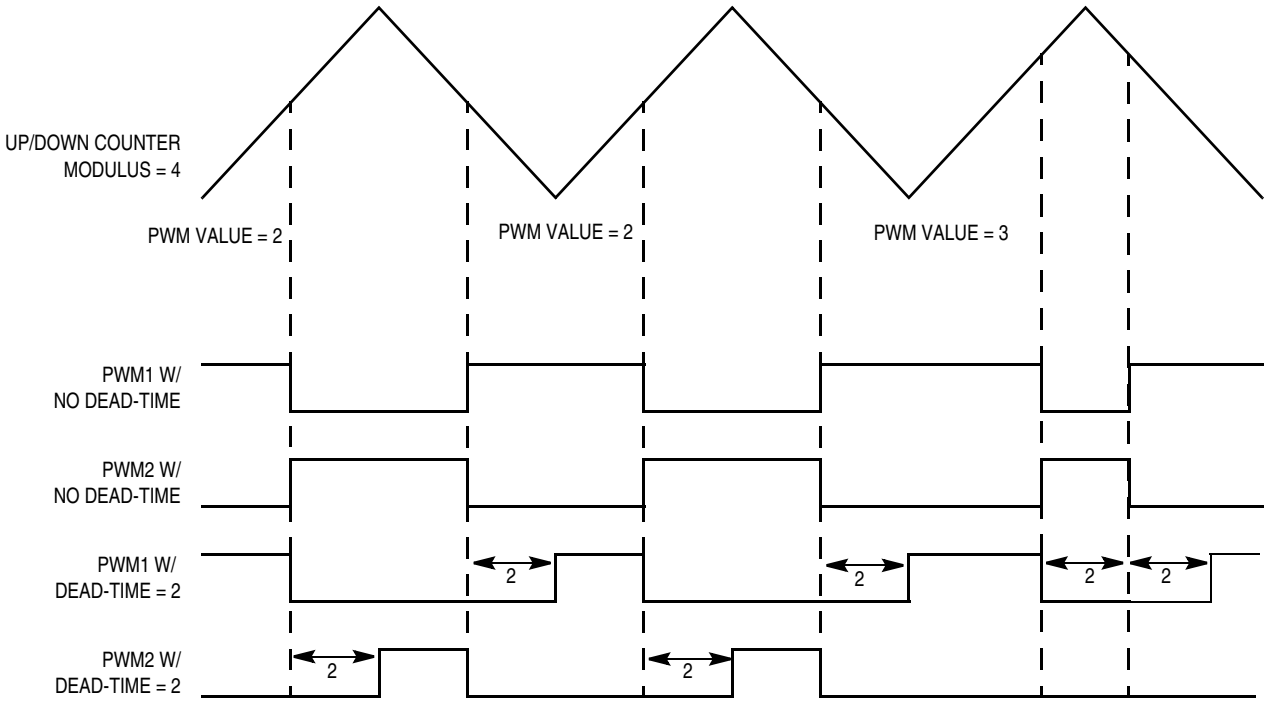

Figure 12-15. Effects of Dead-Time Insertion

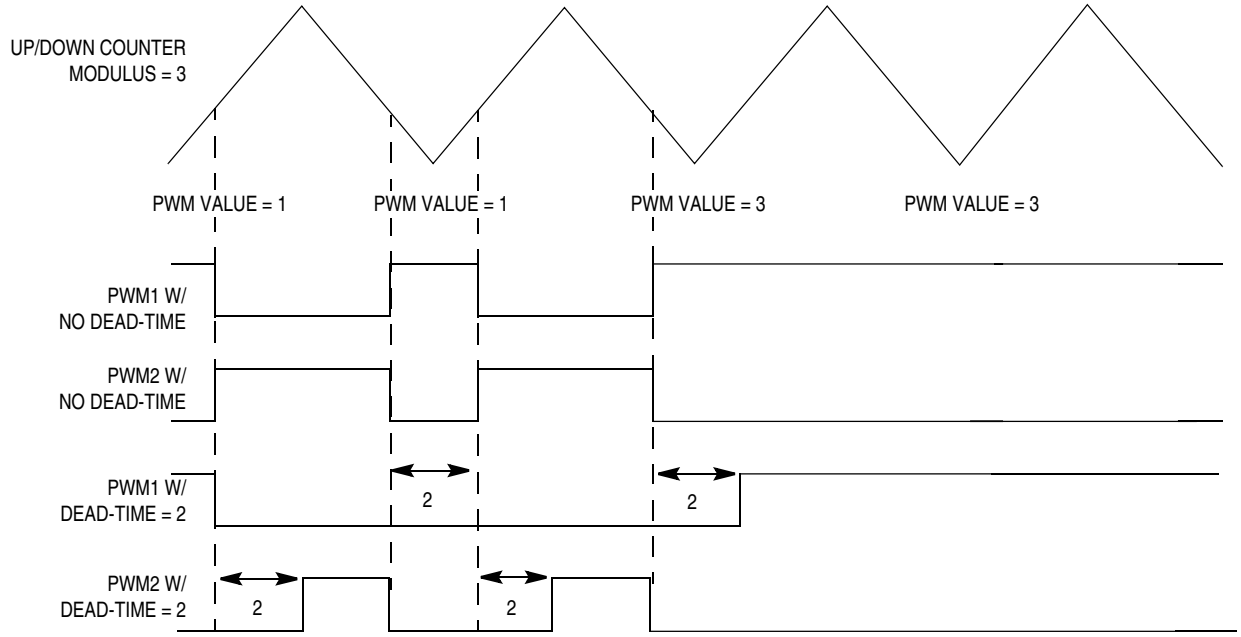

Figure 12-16. Dead-Time at Duty Cycle Boundaries

FMODE4 —Fault Mode Selection for Fault Pin 4 Bit (automatic versus manual mode)

This read/write bit allows the user to select between automatic and manual mode faults. For further descriptions of each mode, see 12.6 Fault Protection.

1 = Automatic mode
0 = Manual mode

FINT3 — Fault 3 Interrupt Enable Bit

This read/write bit allows the CPU interrupt caused by faults on fault pin 3 to be enabled. The fault protection circuitry is independent of this bit and will always be active. If a fault is detected, the PWM pins will still be disabled according to the disable mapping register.

1 = Fault pin 3 will cause CPU interrupts.
0 = Fault pin 3 will not cause CPU interrupts.

FMODE3 —Fault Mode Selection for Fault Pin 3 Bit (automatic versus manual mode)

This read/write bit allows the user to select between automatic and manual mode faults. For further descriptions of each mode, see 12.6 Fault Protection.

1 = Automatic mode
0 = Manual mode

FINT2 — Fault 2 Interrupt Enable Bit

This read/write bit allows the CPU interrupt caused by faults on fault pin 2 to be enabled. The fault protection circuitry is independent of this bit and will always be active. If a fault is detected, the PWM pins will still be disabled according to the disable mapping register.

1 = Fault pin 2 will cause CPU interrupts.
0 = Fault pin 2 will not cause CPU interrupts.

FMODE2 —Fault Mode Selection for Fault Pin 2 Bit (automatic versus manual mode)

This read/write bit allows the user to select between automatic and manual mode faults. For further descriptions of each mode, see 12.6 Fault Protection.

1 = Automatic mode
0 = Manual mode

FINT1 — Fault 1 Interrupt Enable Bit

This read/write bit allows the CPU interrupt caused by faults on fault pin 1 to be enabled. The fault protection circuitry is independent of this bit and will always be active. If a fault is detected, the PWM pins will still be disabled according to the disable mapping register.

1 = Fault pin 1 will cause CPU interrupts.
0 = Fault pin 1 will not cause CPU interrupts.

FMODE1 —Fault Mode Selection for Fault Pin 1 Bit (automatic versus manual mode)

This read/write bit allows the user to select between automatic and manual mode faults. For further descriptions of each mode, see 12.6 Fault Protection.

1 = Automatic mode
0 = Manual mode

14.7.3 SIM Break Flag Control Register

The SIM break control register (SBFCR) contains a bit that enables software to clear status bits while the MCU is in a break state.

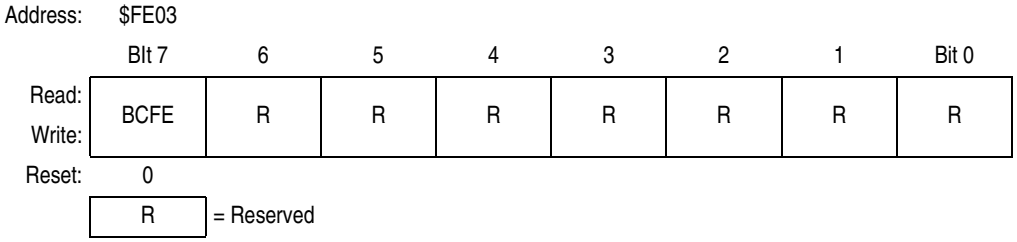

Figure 14-16. SIM Break Flag Control Register (SBFCR)

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

SPRF signals the end of a transmission. Software clears SPRF by reading the SPI status and control register with SPRF set and then reading the SPI data register. Writing to the SPI data register clears the SPTE bit.

15.4.2 Slave Mode

The SPI operates in slave mode when the SPMSTR bit is clear. In slave mode the SPSCK pin is the input for the serial clock from the master MCU. Before a data transmission occurs, the $\overline{SS}$ pin of the slave SPI must be at logic 0. $\overline{SS}$ must remain low until the transmission is complete. See 15.6.2 Mode Fault Error.

In a slave SPI module, data enters the shift register under the control of the serial clock from the master SPI module. After a byte enters the shift register of a slave SPI, it transfers to the receive data register, and the SPRF bit is set. To prevent an overflow condition, slave software then must read the receive data register before another full byte enters the shift register.

The maximum frequency of the SPSCK for an SPI configured as a slave is the bus clock speed (which is twice as fast as the fastest master SPSCK clock that can be generated). The frequency of the SPSCK for an SPI configured as a slave does not have to correspond to any SPI baud rate. The baud rate only controls the speed of the SPSCK generated by an SPI configured as a master. Therefore, the frequency of the SPSCK for an SPI configured as a slave can be any frequency less than or equal to the bus speed.

When the master SPI starts a transmission, the data in the slave shift register begins shifting out on the MISO pin. The slave can load its shift register with a new byte for the next transmission by writing to its transmit data register. The slave must write to its transmit data register at least one bus cycle before the master starts the next transmission. Otherwise, the byte already in the slave shift register shifts out on the MISO pin. Data written to the slave shift register during a transmission remains in a buffer until the end of the transmission.

When the clock phase bit (CPHA) is set, the first edge of SPSCK starts a transmission. When CPHA is clear, the falling edge of $\overline{SS}$ starts a transmission. See 15.5 Transmission Formats.

NOTE

If the write to the data register is late, the SPI transmits the data already in the shift register from the previous transmission.

SPSCK must be in the proper idle state before the slave is enabled to prevent SPSCK from appearing as a clock edge.

15.5 Transmission Formats

During an SPI transmission, data is simultaneously transmitted (shifted out serially) and received (shifted in serially). A serial clock synchronizes shifting and sampling on the two serial data lines. A slave select line allows selection of an individual slave SPI device; slave devices that are not selected do not interfere with SPI bus activities. On a master SPI device, the slave select line can optionally be used to indicate multiple-master bus contention.

15.5.1 Clock Phase and Polarity Controls

Software can select any of four combinations of serial clock (SPSCK) phase and polarity using two bits in the SPI control register (SPCR). The clock polarity is specified by the CPOL control bit, which selects an active high or low clock and has no significant effect on the transmission format.

4. In TIMA channel x status and control register (TSCx):
   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 16-2.)
   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 16-2.)

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 can also cause incorrect PWM signal generation when changing the PWM pulse width to a new, much larger value.

5. In the TIMA status control register (TASC), clear the TIMA stop bit, TSTOP.

Setting MS0B links channels 0 and 1 and configures them for buffered PWM operation. The TIMA channel 0 registers (TACH0H–TACH0L) initially control the buffered PWM output. TIMA status control register 0 (TASC0) controls and monitors the PWM signal from the linked channels. MS0B takes priority over MS0A.

Setting MS2B links channels 2 and 3 and configures them for buffered PWM operation. The TIMA channel 2 registers (TACH2H–TACH2L) initially control the buffered PWM output. TIMA status control register 2 (TASC2) controls and monitors the PWM signal from the linked channels. MS2B takes priority over MS2A.

Clearing the toggle-on-overflow bit, TOVx, inhibits output toggles on TIMA overflows. Subsequent output compares try to force the output to a state it is already in and have no effect. The result is a 0 percent duty cycle output.

Setting the channel x maximum duty cycle bit (CHxMAX) and setting the TOVx bit generates a 100 percent duty cycle output. (See 16.7.4 TIMA Channel Status and Control Registers.)

16.4 Interrupts

These TIMA sources can generate interrupt requests:

- TIMA overflow flag (TOF) — The timer overflow flag (TOF) bit is set when the TIMA counter reaches the modulo value programmed in the TIMA counter modulo registers. The TIMA overflow interrupt enable bit, TOIE, enables TIMA overflow interrupt requests. TOF and TOIE are in the TIMA status and control registers.
- TIMA channel flags (CH3F–CH0F) — The CHxF bit is set when an input capture or output compare occurs on channel x. Channel x TIMA CPU interrupt requests are controlled by the channel x interrupt enable bit, CHxIE.

16.5 Wait Mode

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

Register Name and Address: TACH2L — $001B

	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:	Indeterminate after reset							

Register Name and Address: TACH3H — $001D

	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:	Indeterminate after reset							

Register Name and Address: TACH3L — $001E

	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:	Indeterminate after reset							

Figure 16-10. TIMA Channel Registers (TACH0H/L–TACH3H/L) (Continued)

17.4 Interrupts

These TIMB sources can generate interrupt requests:

- TIMB overflow flag (TOF) — The timer overflow flag (TOF) bit is set when the TIMB counter reaches the modulo value programmed in the TIMB counter modulo registers. The TIMB overflow interrupt enable bit, TOIE, enables TIMB overflow interrupt requests. TOF and TOIE are in the TIMB status and control registers.
- TIMB channel flags (CH1F–CH0F) — The CHxF bit is set when an input capture or output compare occurs on channel x. Channel x TIMB CPU interrupt requests are controlled by the channel x interrupt enable bit, CHxIE.

17.5 Wait Mode

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

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

If TIMB functions are not required during wait mode, reduce power consumption by stopping the TIMB before executing the WAIT instruction.

17.6 I/O Signals

Port E shares three of its pins with the TIMB:

- PTE0/TCLKB is an external clock input to the TIMB prescaler.
- The two TIMB channel I/O pins are PTE1/TCH0B and PTE2/TCH1B.

17.6.1 TIMB Clock Pin (PTE0/TCLKB)

PTE0/TCLKB is an external clock input that can be the clock source for the TIMB counter instead of the prescaled internal bus clock. Select the PTE0/TCLKB input by writing 1s to the three prescaler select bits, PS[2:0]. See 17.7.1 TIMB Status and Control Register.

The maximum TCLK frequency is the least: 4 MHz or bus frequency ÷ 2.

PTE0/TCLKB is available as a general-purpose I/O pin or ADC channel when not used as the TIMB clock input. When the PTE0/TCLKB pin is the TIMB clock input, it is an input regardless of the state of the DDRE0 bit in data direction register E.

17.6.2 TIMB Channel I/O Pins (PTE1/TCH0B–PTE2/TCH1B)

Each channel I/O pin is programmable independently as an input capture pin or an output compare pin. PTE1/TCH0B and PTE2/TCH1B can be configured as buffered output compare or buffered PWM pins.

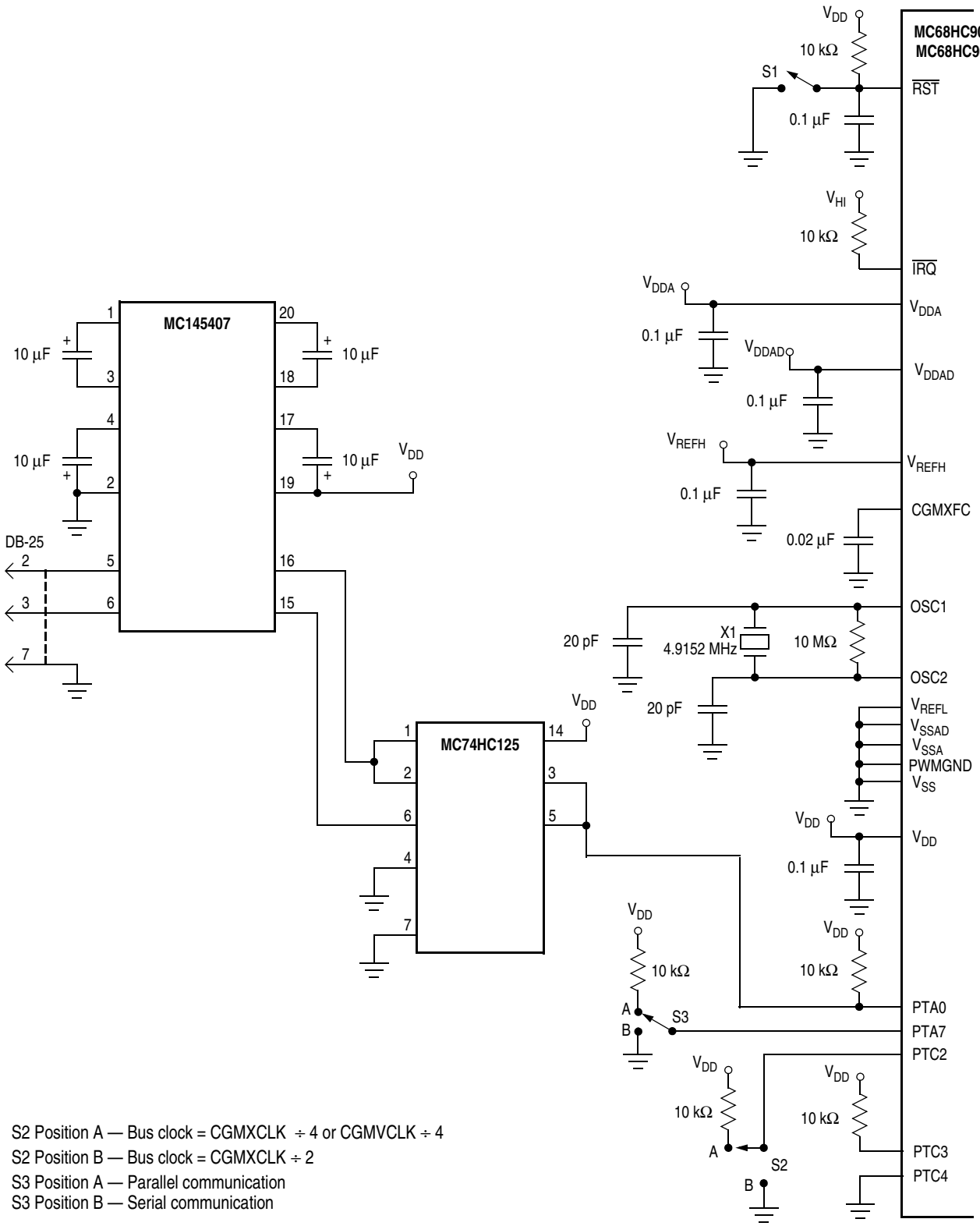

Figure 18-8. Monitor Mode Circuit

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

MC68HC908MR32
Rev. 6.1, 07/2005

freescale™
semiconductor