



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	CANbus, SCI, SPI
Peripherals	LVD, POR, PWM
Number of I/O	53
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1.5K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 24x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	64-QFP
Supplier Device Package	64-QFP (14x14)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc908gz32vfue

Table of Contents

Chapter 1 General Description

1.1	Introduction	23
1.2	Features	23
1.2.1	Standard Features	23
1.2.2	Features of the CPU08	25
1.3	MCU Block Diagram	25
1.4	Pin Assignments	25
1.5	Pin Functions	28
1.5.1	Power Supply Pins (V_{DD} and V_{SS})	28
1.5.2	Oscillator Pins (OSC1 and OSC2)	29
1.5.3	External Reset Pin (\overline{RST})	29
1.5.4	External Interrupt Pin (\overline{IRQ})	29
1.5.5	CGM Power Supply Pins (V_{DDA} and V_{SSA})	29
1.5.6	External Filter Capacitor Pin (CGMXFC)	29
1.5.7	ADC Power Supply/Reference Pins (V_{DDAD}/V_{REFH} and V_{SSAD}/V_{REFL})	29
1.5.8	Port A Input/Output (I/O) Pins (PTA7/KBD7/AD15–PTA0/KBD0/AD8)	30
1.5.9	Port B I/O Pins (PTB7/AD7–PTB0/AD0)	30
1.5.10	Port C I/O Pins (PTC6–PTC0/CANTX)	30
1.5.11	Port D I/O Pins (PTD7/T2CH1–PTD0/SS)	30
1.5.12	Port E I/O Pins (PTE5–PTE2, PTE1/RxD, and PTE0/TxD)	30
1.5.13	Port F I/O Pins (PTF7/T2CH5–PTF0)	31
1.5.14	Port G I/O Pins (PTG7/AD23–PTBG0/AD16)	31
1.5.15	Unused Pin Termination	31

Chapter 2 Memory

2.1	Introduction	33
2.2	Unimplemented Memory Locations	33
2.3	Reserved Memory Locations	33
2.4	Input/Output (I/O) Section	33
2.5	Random-Access Memory (RAM)	45
2.6	FLASH-1 Memory (FLASH-1)	45
2.6.1	Functional Description	45
2.6.2	FLASH-1 Control and Block Protect Registers	46
2.6.2.1	FLASH-1 Control Register	46
2.6.2.2	FLASH-1 Block Protect Register	47
2.6.3	FLASH-1 Block Protection	48
2.6.4	FLASH-1 Mass Erase Operation	48
2.6.5	FLASH-1 Page Erase Operation	49

Memory

- \$FF81; FLASH-2 block protect register, FL2BPR
- \$FF88; FLASH-1 control register, FL1CR

Data registers are shown in Figure 2-2. Table 2-1 is a list of vector locations.

\$0000 ↓ \$003F	I/O REGISTERS 64 BYTES	\$FE00	SIM BREAK STATUS REGISTER (BSR)
\$0040 ↓ \$043F	RAM-1 1024 BYTES	\$FE01	SIM RESET STATUS REGISTER (SRSR)
\$0440 ↓ \$0461	I/O REGISTERS 34 BYTES	\$FE02	RESERVED
\$0462 ↓ \$04FF	FLASH-2 158 BYTES	\$FE03	SIM BREAK FLAG CONTROL REGISTER (BFCR)
\$0500 ↓ \$057F	MSCAN CONTROL AND MESSAGE BUFFER 128 BYTES	\$FE04	INTERRUPT STATUS REGISTER 1 (INT1)
\$0580 ↓ \$097F	RAM-2 1024 BYTES	\$FE05	INTERRUPT STATUS REGISTER 2 (INT2)
\$0980 ↓ \$1B7F	FLASH-2 4608 BYTES	\$FE06	INTERRUPT STATUS REGISTER 3 (INT3)
\$1B80 ↓ \$1DFF	RESERVED 640 BYTES	\$FE07	INTERRUPT STATUS REGISTER 4 (INT4)
\$1E00 ↓ \$1E0F	MONITOR ROM 16 BYTES	\$FE08	FLASH-2 CONTROL REGISTER (FL2CR)
\$1E10 ↓ \$1E1F	RESERVED 16 BYTES	\$FE09	BREAK ADDRESS REGISTER HIGH (BRKH)
\$1E20 ↓ \$7FFF	FLASH-2 25,056 BYTES	\$FE0A	BREAK ADDRESS REGISTER LOW (BRKL)
\$8000 ↓ \$FDFF	FLASH-1 32,256 BYTES	\$FE0B	BREAK STATUS AND CONTROL REGISTER (BRKSCR)
		\$FE0C	LVI STATUS REGISTER (LVISR)
		\$FE0D	FLASH-2 TEST CONTROL REGISTER (FLT2CR)
		\$FE0E	FLASH-1 TEST CONTROL REGISTER (FLT1CR)
		\$FE0F	UNIMPLEMENTED
		\$FE10 ↓ \$FE1F	UNIMPLEMENTED 16 BYTES RESERVED FOR COMPATIBILITY WITH MONITOR CODE FOR A-FAMILY PART
		\$FE20 ↓ \$FF7F	MONITOR ROM 352 BYTES
		\$FF80	FLASH-1 BLOCK PROTECT REGISTER (FL1BPR)
		\$FF81	FLASH-2 BLOCK PROTECT REGISTER (FL2BPR)
		\$FF82 ↓ \$FF87	RESERVED 6 BYTES
		\$FF88	FLASH-1 CONTROL REGISTER (FL1CR)
		\$FF89 ↓ \$FFCB	RESERVED 67 BYTES
		\$FFCC ↓ \$FFFF ⁽¹⁾	FLASH-1 VECTORS 52 BYTES

1. \$FFF6–\$FFFD used for eight security bytes

Figure 2-1. MC68HC908GZ60 Memory Map

Memory

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
\$003C	ADC Status and Control Register (ADSCR) See page 68.	Read:	COCO	AIEN	ADCO	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0
		Write:	R							
		Reset:	0	0	0	1	1	1	1	1
\$003D	ADC Data High Register (ADRH) See page 70.	Read:	0	0	0	0	0	0	AD9	AD8
		Write:								
		Reset:	Unaffected by reset							
\$003E	ADC Data Low Register (ADRL) See page 70.	Read:	AD7	AD6	AD5	AD4	A3	AD2	AD1	AD0
		Write:								
		Reset:	Unaffected by reset							
\$003F	ADC Clock Register (ADCLK) See page 72.	Read:	ADIV2	ADIV1	ADIV0	ADICLK	MODE1	MODE0	R	0
		Write:								
		Reset:	0	0	0	0	0	1	0	0
\$0440	Port F Data Register (PTF) See page 185.	Read:	PTF7	PTF6	PTF5	PTF4	PTAF3	PTF2	PTF1	PTF0
		Write:								
		Reset:	Unaffected by reset							
\$0441	Port G Data Register (PTG) See page 186.	Read:	PTG7	PTG6	PTG5	PTG4	PTG3	PTG2	PTG1	PTG0
		Write:								
		Reset:	Unaffected by reset							
\$0444	Data Direction Register F (DDRF) See page 185.	Read:	DDRF7	DDRF6	DDRF5	DDRF4	DDRF3	DDRF2	DDRF1	DDRF0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0445	Data Direction Register G (DDRG) See page 187.	Read:	DDRG7	DDRG6	DDRG5	DDRG4	DDRG3	DDRG2	DDRG1	DDRG0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0448	Keyboard Interrupt Polarity Register (INTKBIPR) See page 121.	Read:	KBIP7	KBIP6	KBIP5	KBIP4	KBIP3	KBIP2	KBIP1	KBIP0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0456	TIM2 Channel 2 Status and Control Register (T2SC2) See page 297.	Read:	CH2F	CH2IE	MS2B	MS2A	ELS2B	ELS2A	TOV2	CH2MAX
		Write:	0							
		Reset:	0	0	0	0	0	0	0	0
\$0457	TIM2 Channel 2 Register High (T2CH2H) See page 297.	Read:	Bit 15	14	13	12	11	10	9	Bit 8
		Write:								
		Reset:	Indeterminate after reset							
\$0458	TIM2 Channel 2 Register Low (T2CH2L) See page 297.	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
		Reset:	Indeterminate after reset							

= Unimplemented R = Reserved U = Unaffected

Figure 2-2. Control, Status, and Data Registers (Sheet 6 of 9)

The following conditions apply when in manual mode:

- \overline{ACQ} is a writable control bit that controls the mode of the filter. Before turning on the PLL in manual mode, the \overline{ACQ} bit must be clear.
- Before entering tracking mode ($\overline{ACQ} = 1$), software must wait a given time, t_{ACQ} (See 4.8 Acquisition/Lock Time Specifications.), after turning on the PLL by setting PLLON in the PLL control register (PCTL).
- Software must wait a given time, t_{AL} , after entering tracking mode before selecting the PLL as the clock source to CGMOUT (BCS = 1).
- The LOCK bit is disabled.
- CPU interrupts from the CGM are disabled.

4.3.6 Programming the PLL

Use the following procedure to program the PLL. For reference, the variables used and their meaning are shown in Table 4-1.

Table 4-1. Variable Definitions

Variable	Definition
f_{BUSDES}	Desired bus clock frequency
$f_{VCLKDES}$	Desired VCO clock frequency
f_{RCLK}	Chosen reference crystal frequency
f_{VCLK}	Calculated VCO clock frequency
f_{BUS}	Calculated bus clock frequency
f_{NOM}	Nominal VCO center frequency
f_{VRS}	Programmed VCO center frequency

NOTE

The round function in the following equations means that the real number should be rounded to the nearest integer number.

1. Choose the desired bus frequency, f_{BUSDES} .
2. Calculate the desired VCO frequency (four times the desired bus frequency).

$$f_{VCLKDES} = 4 \times f_{BUSDES}$$

3. Choose a practical PLL (crystal) reference frequency, f_{RCLK} . Typically, the reference crystal is 1–8 MHz.

Frequency errors to the PLL are corrected at a rate of f_{RCLK} .

For stability and lock time reduction, this rate must be as fast as possible. The VCO frequency must be an integer multiple of this rate. The relationship between the VCO frequency, f_{VCLK} , and the reference frequency, f_{RCLK} , is:

$$f_{VCLK} = (N) (f_{RCLK})$$

N, the range multiplier, must be an integer.

4.8 Acquisition/Lock Time Specifications

The acquisition and lock times of the PLL are, in many applications, the most critical PLL design parameters. Proper design and use of the PLL ensures the highest stability and lowest acquisition/lock times.

4.8.1 Acquisition/Lock Time Definitions

Typical control systems refer to the acquisition time or lock time as the reaction time, within specified tolerances, of the system to a step input. In a PLL, the step input occurs when the PLL is turned on or when it suffers a noise hit. The tolerance is usually specified as a percent of the step input or when the output settles to the desired value plus or minus a percent of the frequency change. Therefore, the reaction time is constant in this definition, regardless of the size of the step input. For example, consider a system with a 5 percent acquisition time tolerance. If a command instructs the system to change from 0 Hz to 1 MHz, the acquisition time is the time taken for the frequency to reach 1 MHz \pm 50 kHz. Fifty kHz = 5% of the 1-MHz step input. If the system is operating at 1 MHz and suffers a -100-kHz noise hit, the acquisition time is the time taken to return from 900 kHz to 1 MHz \pm 5 kHz. Five kHz = 5% of the 100-kHz step input.

Other systems refer to acquisition and lock times as the time the system takes to reduce the error between the actual output and the desired output to within specified tolerances. Therefore, the acquisition or lock time varies according to the original error in the output. Minor errors may not even be registered. Typical PLL applications prefer to use this definition because the system requires the output frequency to be within a certain tolerance of the desired frequency regardless of the size of the initial error.

4.8.2 Parametric Influences on Reaction Time

Acquisition and lock times are designed to be as short as possible while still providing the highest possible stability. These reaction times are not constant, however. Many factors directly and indirectly affect the acquisition time.

The most critical parameter which affects the reaction times of the PLL is the reference frequency, f_{RCLK} . This frequency is the input to the phase detector and controls how often the PLL makes corrections. For stability, the corrections must be small compared to the desired frequency, so several corrections are required to reduce the frequency error. Therefore, the slower the reference the longer it takes to make these corrections. This parameter is under user control via the choice of crystal frequency f_{XCLK} . (See 4.3.3 PLL Circuits and 4.3.6 Programming the PLL.)

Another critical parameter is the external filter network. The PLL modifies the voltage on the VCO by adding or subtracting charge from capacitors in this network. Therefore, the rate at which the voltage changes for a given frequency error (thus change in charge) is proportional to the capacitance. The size of the capacitor also is related to the stability of the PLL. If the capacitor is too small, the PLL cannot make small enough adjustments to the voltage and the system cannot lock. If the capacitor is too large, the PLL may not be able to adjust the voltage in a reasonable time. (See 4.8.3 Choosing a Filter.)

Also important is the operating voltage potential applied to V_{DDA} . The power supply potential alters the characteristics of the PLL. A fixed value is best. Variable supplies, such as batteries, are acceptable if they vary within a known range at very slow speeds. Noise on the power supply is not acceptable, because it causes small frequency errors which continually change the acquisition time of the PLL.

Temperature and processing also can affect acquisition time because the electrical characteristics of the PLL change. The part operates as specified as long as these influences stay within the specified limits.

Chapter 6

Computer Operating Properly (COP) Module

6.1 Introduction

The computer operating properly (COP) module contains a free-running counter that generates a reset if allowed to overflow. The COP module helps software recover from runaway code. Prevent a COP reset by clearing the COP counter periodically. The COP module can be disabled through the COPD bit in the CONFIG register.

6.2 Functional Description

Figure 6-1 shows the structure of the COP module.

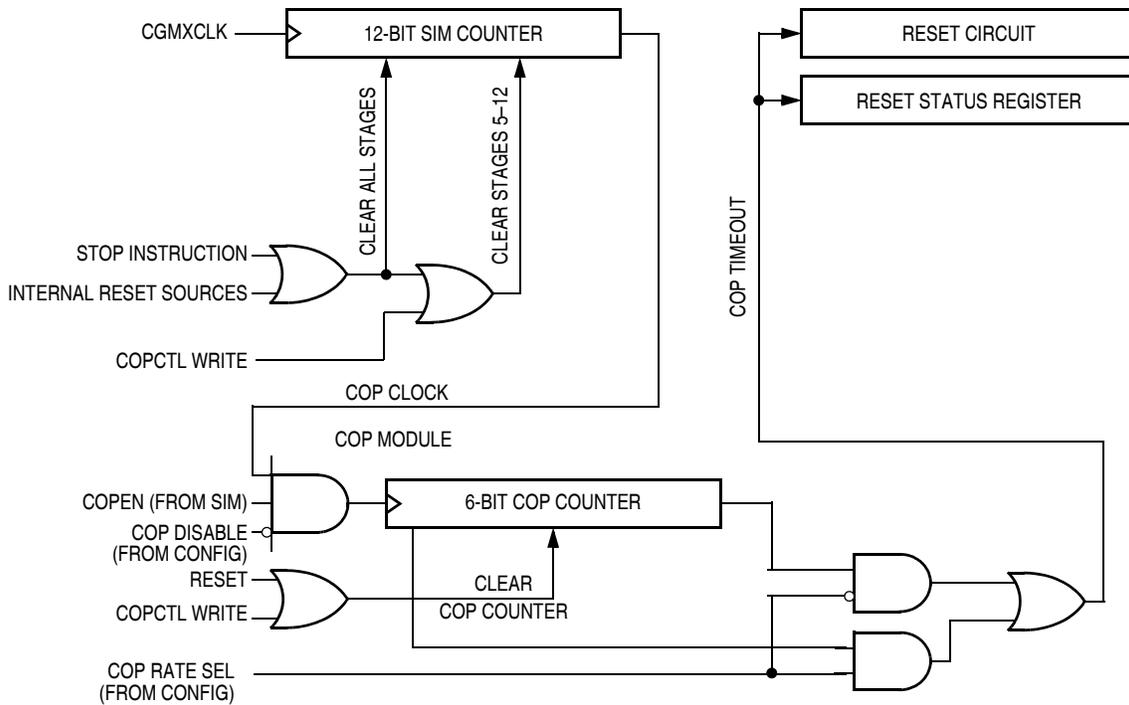


Figure 6-1. COP Block Diagram

Chapter 9

Keyboard Interrupt Module (KBI)

9.1 Introduction

The keyboard interrupt module (KBI) provides eight independently maskable external interrupts which are accessible via PTA0–PTA7. When a port pin is enabled for keyboard interrupt function, an internal pullup/pulldown device is also enabled on the pin.

9.2 Features

Features include:

- Eight keyboard interrupt pins with separate keyboard interrupt enable bits and one keyboard interrupt mask
- Hysteresis buffers
- Programmable edge-only or edge- and level- interrupt sensitivity
- Edge detect programmable for rising or falling edges
- Level detect programmable for high or low levels
- Exit from low-power modes
- Pullup/pulldown device automatically configured based on polarity of edge/level selection

9.3 Functional Description

Writing to the KBIE7–KBIE0 bits in the keyboard interrupt enable register independently enables or disables each port A pin as a keyboard interrupt pin. Enabling a keyboard interrupt pin also enables its internal pullup/pulldown device. On falling edge or low level selection a pullup device is configured. On rising edge or high level selection a pulldown device is configured.

- A falling edge is detected when an enabled keyboard input signal is seen as a 1 (the deasserted level) during one bus cycle and then a 0 (the asserted level) during the next cycle.
- A rising edge is detected when the input signal is seen as a 0 during one bus cycle and then a 1 during the next cycle.

A keyboard interrupt is latched when one or more keyboard pins are asserted. The MODEK bit in the keyboard status and control register controls the triggering mode of the keyboard interrupt.

The KBIP7–KBIP0 bits determine the polarity of the keyboard pin detection. These bits along with the MODEK bit determine whether a logic level (0 or 1) and/or a falling (or rising) edge is being detected.

- If the keyboard interrupt is edge-sensitive only, a falling (or rising) edge on a keyboard pin does not latch an interrupt request if another keyboard pin is already asserted. To prevent losing an interrupt request on one pin because another pin is still asserted, software can disable the latter pin while it is asserted.
- If the keyboard interrupt is edge and level sensitive, an interrupt request is present as long as any keyboard interrupt pin is asserted and the pin is keyboard interrupt enabled.

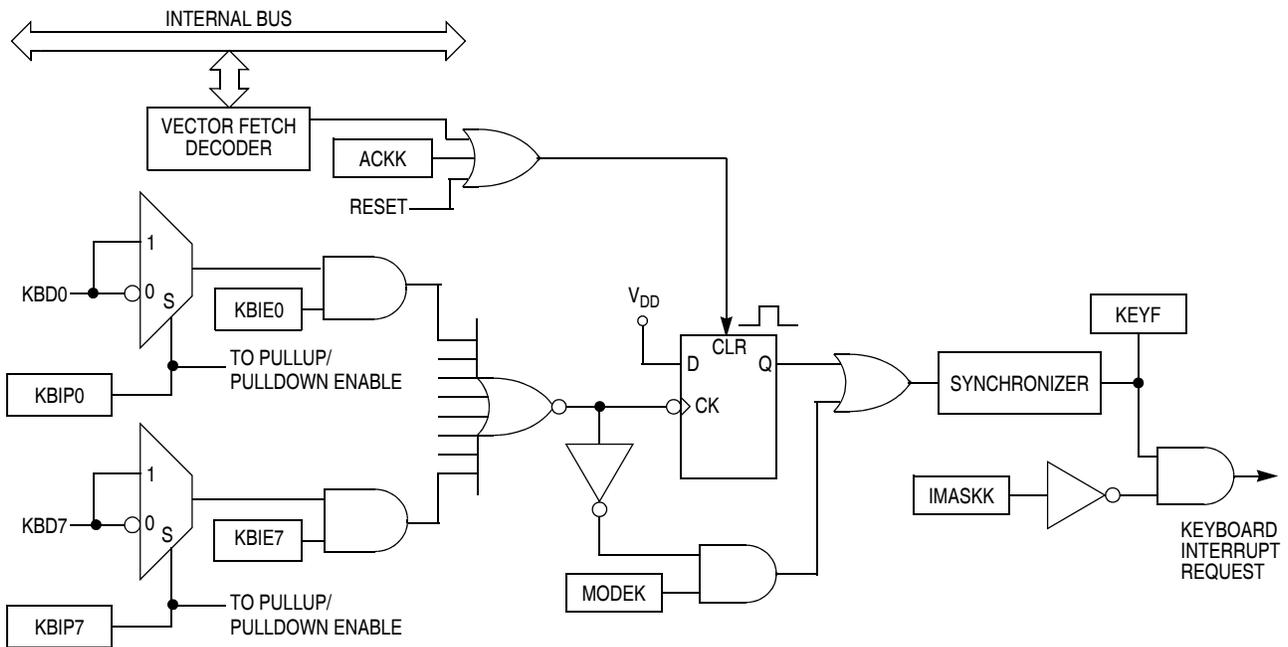


Figure 9-2. Keyboard Module Block Diagram

Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0		
\$001A	Keyboard Status and Control Register (INTKBSCR) See page 120.	Read:	0	0	0	0	KEYF	0	IMASKK	MODEK	
		Write:									ACKK
		Reset:	0	0	0	0	0	0	0	0	0
\$001B	Keyboard Interrupt Enable Register (INTKBIER) See page 121.	Read:	KBIE7	KBIE6	KBIE5	KBIE4	KBIE3	KBIE2	KBIE1	KBIE0	
		Write:									
		Reset:	0	0	0	0	0	0	0	0	0
\$0448	Keyboard Interrupt Polarity Register (INTKBIPR) See page 121.	Read:	KBIP7	KBIP6	KBIP5	KBIP4	KBIP3	KBIP2	KBIP1	KBIP0	
		Write:									
		Reset:	0	0	0	0	0	0	0	0	0

= Unimplemented

Figure 9-3. I/O Register Summary

Table 12-1. MSCAN08 Interrupt Vector Addresses

Function	Source	Local Mask	Global Mask
Wakeup	WUPIF	WUPIE	1 bit
Error interrupts	RWRNIF	RWRNIE	
	TWRNIF	TWRNIE	
	RERRIF	RERRIE	
	TERRIF	TERRIE	
	BOFFIF	BOFFIE	
	OVRIF	OVRIE	
Receive	RXF	RXFIE	
Transmit	TXE0	TXEIE0	
	TXE1	TXEIE1	
	TXE2	TXEIE2	

12.7 Protocol Violation Protection

The MSCAN08 will protect the user from accidentally violating the CAN protocol through programming errors. The protection logic implements the following features:

- The receive and transmit error counters cannot be written or otherwise manipulated.
- All registers which control the configuration of the MSCAN08 can not be modified while the MSCAN08 is on-line. The SFTRES bit in the MSCAN08 module control register (see 12.13.1 MSCAN08 Module Control Register 0) serves as a lock to protect the following registers:
 - MSCAN08 module control register 1 (CMCR1)
 - MSCAN08 bus timing register 0 and 1 (CBTR0 and CBTR1)
 - MSCAN08 identifier acceptance control register (CIDAC)
 - MSCAN08 identifier acceptance registers (CIDAR0–3)
 - MSCAN08 identifier mask registers (CIDMR0–3)
- The CAN_{TX} pin is forced to recessive when the MSCAN08 is in any of the low-power modes.

12.8 Low-Power Modes

In addition to normal mode, the MSCAN08 has three modes with reduced power consumption: sleep, soft reset, and power down. In sleep and soft reset mode, power consumption is reduced by stopping all clocks except those to access the registers. In power-down mode, all clocks are stopped and no power is consumed.

The WAIT and STOP instructions put the MCU in low-power consumption stand-by modes. Table 12-2 summarizes the combinations of MSCAN08 and CPU modes. A particular combination of modes is entered for the given settings of the bits SLPK and SFTRES. For all modes, an MSCAN08 wakeup interrupt can occur only if SLPK = WUPIE = 1.

Table 13-2. Port A Pin Functions

PTAPUE Bit	DDRA Bit	PTA Bit	I/O Pin Mode	Accesses to DDRA	Accesses to PTA	
				Read/Write	Read	Write
1	0	X ⁽¹⁾	Input, V _{DD} ⁽²⁾	DDRA7–DDRA0	Pin	PTA7–PTA0 ⁽³⁾
0	0	X	Input, Hi-Z ⁽⁴⁾	DDRA7–DDRA0	Pin	PTA7–PTA0 ⁽³⁾
X	1	X	Output	DDRA7–DDRA0	PTA7–PTA0	PTA7–PTA0

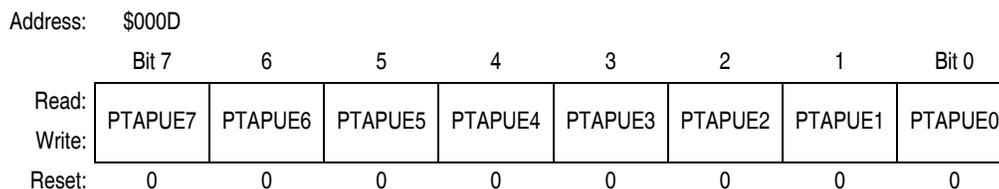
1. X = Don't care
2. I/O pin pulled up to V_{DD} by internal pullup device
3. Writing affects data register, but does not affect input.
4. Hi-Z = High impedance

13.3.3 Port A Input Pullup Enable Register

The port A input pullup enable register (PTAPUE) contains a software configurable pullup device for each of the eight port A pins. Each bit is individually configurable and requires that the data direction register, DDRA, bit be configured as an input. Each pullup is automatically and dynamically disabled when a port bit's DDRA is configured for output mode.

NOTE

Pullup or pulldown resistors are automatically selected for keyboard interrupt pins depending on the bit settings in the keyboard interrupt polarity register (INTKBIPR) see 9.7.3 Keyboard Interrupt Polarity Register.


Figure 13-5. Port A Input Pullup Enable Register (PTAPUE)

PTAPUE7–PTAPUE0 — Port A Input Pullup Enable Bits

These writable bits are software programmable to enable pullup devices on an input port bit.

- 1 = Corresponding port A pin configured to have internal pullup
- 0 = Corresponding port A pin has internal pullup disconnected

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
\$0009	ESCI Prescaler Register (SCPSC) See page 214.	Read:	PDS2	PDS1	PDS0	PSSB4	PSSB3	PSSB2	PSSB1	PSSB0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$000A	ESCI Arbiter Control Register (SCIACTL) See page 217.	Read:	AM1	Alost	AM0	ACLK	AFIN	ARUN	AROVFL	ARD8
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$000B	ESCI Arbiter Data Register (SCIADAT) See page 218.	Read:	ARD7	ARD6	ARD5	ARD4	ARD3	ARD2	ARD1	ARD0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0013	ESCI Control Register 1 (SCC1) See page 204.	Read:	LOOPS	ENSCI	TXINV	M	WAKE	ILTY	PEN	PTY
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0014	ESCI Control Register 2 (SCC2) See page 206.	Read:	SCTIE	TCIE	SCRIE	ILIE	TE	RE	RWU	SBK
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0015	ESCI Control Register 3 (SCC3) See page 208.	Read:	R8	T8	R	R	ORIE	NEIE	FEIE	PEIE
		Write:								
		Reset:	U	0	0	0	0	0	0	0
\$0016	ESCI Status Register 1 (SCS1) See page 209.	Read:	SCTE	TC	SCRf	IDLE	OR	NF	FE	PE
		Write:								
		Reset:	1	1	0	0	0	0	0	0
\$0017	ESCI Status Register 2 (SCS2) See page 211.	Read:	0	0	0	0	0	0	BKF	RPF
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0018	ESCI Data Register (SCDR) See page 212.	Read:	R7	R6	R5	R4	R3	R2	R1	R0
		Write:	T7	T6	T5	T4	T3	T2	T1	T0
		Reset:	Unaffected by reset							
\$0019	ESCI Baud Rate Register (SCBR) See page 212.	Read:	LINT	LINR	SCP1	SCP0	R	SCR2	SCR1	SCR0
		Write:								
		Reset:	0	0	0	0	0	0	0	0

= Unimplemented
 R = Reserved

Figure 14-4. ESCI I/O Register Summary

14.5.2 Stop Mode

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

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

14.6 ESCI During Break Module Interrupts

The BCFE bit in the break flag control register (SBFCR) enables software to clear status bits during the break state. See 20.2 Break Module (BRK).

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

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

14.7 I/O Signals

Port E shares two of its pins with the ESCI module. The two ESCI I/O pins are:

- PTE0/TxD — transmit data
- PTE1/RxD — receive data

14.7.1 PTE0/TxD (Transmit Data)

The PTE0/TxD pin is the serial data output from the ESCI transmitter. The ESCI shares the PTE0/TxD pin with port E. When the ESCI is enabled, the PTE0/TxD pin is an output regardless of the state of the DDRE0 bit in data direction register E (DDRE).

14.7.2 PTE1/RxD (Receive Data)

The PTE1/RxD pin is the serial data input to the ESCI receiver. The ESCI shares the PTE1/RxD pin with port E. When the ESCI is enabled, the PTE1/RxD pin is an input regardless of the state of the DDRE1 bit in data direction register E (DDRE).

14.8 I/O Registers

These I/O registers control and monitor ESCI operation:

- ESCI control register 1, SCC1
- ESCI control register 2, SCC2
- ESCI control register 3, SCC3
- ESCI status register 1, SCS1
- ESCI status register 2, SCS2
- ESCI data register, SCDR

IDLE — Receiver Idle Bit

This clearable, read-only bit is set when 10 or 11 consecutive 1s appear on the receiver input. IDLE generates an ESCI receiver CPU interrupt request if the ILIE bit in SCC2 is also set. Clear the IDLE bit by reading SCS1 with IDLE set and then reading the SCDR. After the receiver is enabled, it must receive a valid character that sets the SCRF bit before an idle condition can set the IDLE bit. Also, after the IDLE bit has been cleared, a valid character must again set the SCRF bit before an idle condition can set the IDLE bit. Reset clears the IDLE bit.

- 1 = Receiver input idle
- 0 = Receiver input active (or idle since the IDLE bit was cleared)

OR — Receiver Overrun Bit

This clearable, read-only bit is set when software fails to read the SCDR before the receive shift register receives the next character. The OR bit generates an ESCI error CPU interrupt request if the ORIE bit in SCC3 is also set. The data in the shift register is lost, but the data already in the SCDR is not affected. Clear the OR bit by reading SCS1 with OR set and then reading the SCDR. Reset clears the OR bit.

- 1 = Receive shift register full and SCRF = 1
- 0 = No receiver overrun

Software latency may allow an overrun to occur between reads of SCS1 and SCDR in the flag-clearing sequence. Figure 14-14 shows the normal flag-clearing sequence and an example of an overrun caused by a delayed flag-clearing sequence. The delayed read of SCDR does not clear the OR bit because OR was not set when SCS1 was read. Byte 2 caused the overrun and is lost. The next flag-clearing sequence reads byte 3 in the SCDR instead of byte 2.

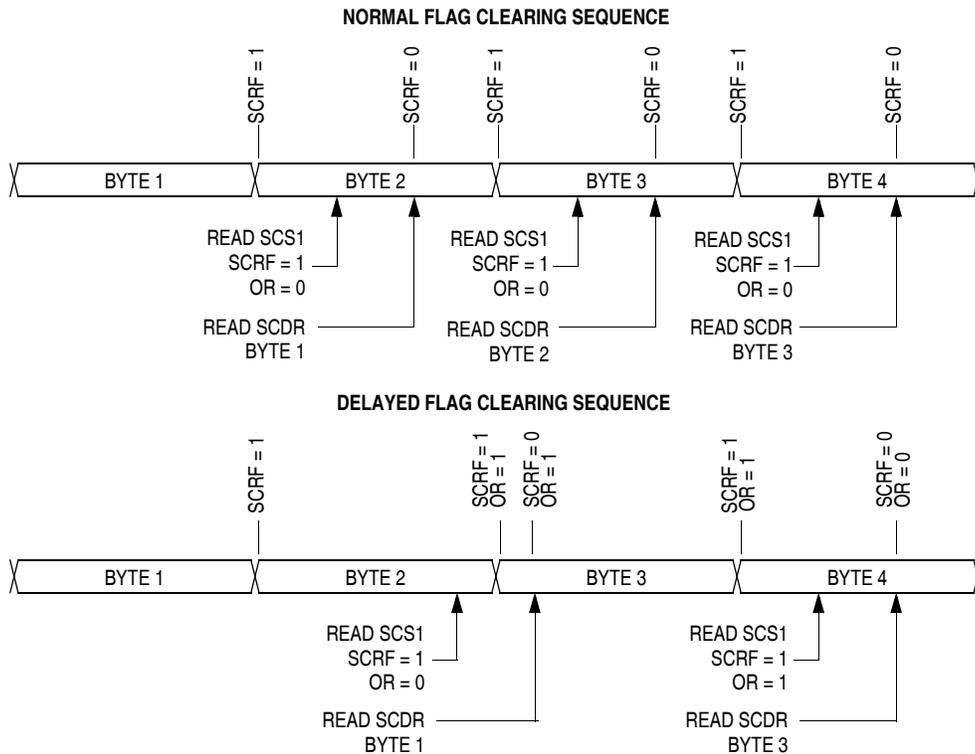


Figure 14-14. Flag Clearing Sequence

In applications that are subject to software latency or in which it is important to know which byte is lost due to an overrun, the flag-clearing routine can check the OR bit in a second read of SCS1 after reading the data register.

NF — Receiver Noise Flag Bit

This clearable, read-only bit is set when the ESCI detects noise on the RxD pin. NF generates an NF CPU interrupt request if the NEIE bit in SCC3 is also set. Clear the NF bit by reading SCS1 and then reading the SCDR. Reset clears the NF bit.

- 1 = Noise detected
- 0 = No noise detected

FE — Receiver Framing Error Bit

This clearable, read-only bit is set when a 0 is accepted as the stop bit. FE generates an ESCI error CPU interrupt request if the FEIE bit in SCC3 also is set. Clear the FE bit by reading SCS1 with FE set and then reading the SCDR. Reset clears the FE bit.

- 1 = Framing error detected
- 0 = No framing error detected

PE — Receiver Parity Error Bit

This clearable, read-only bit is set when the ESCI detects a parity error in incoming data. PE generates a PE CPU interrupt request if the PEIE bit in SCC3 is also set. Clear the PE bit by reading SCS1 with PE set and then reading the SCDR. Reset clears the PE bit.

- 1 = Parity error detected
- 0 = No parity error detected

14.8.5 ESCI Status Register 2

ESCI status register 2 (SCS2) contains flags to signal these conditions:

- Break character detected
- Incoming data

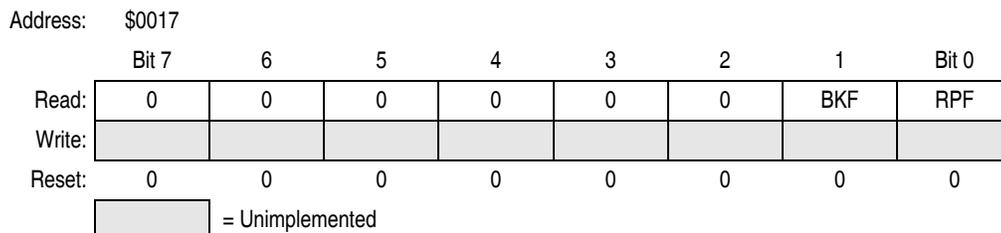


Figure 14-15. ESCI Status Register 2 (SCS2)

BKF — Break Flag Bit

This clearable, read-only bit is set when the ESCI detects a break character on the RxD pin. In SCS1, the FE and SCRF bits are also set. In 9-bit character transmissions, the R8 bit in SCC3 is cleared. BKF does not generate a CPU interrupt request. Clear BKF by reading SCS2 with BKF set and then reading the SCDR. Once cleared, BKF can become set again only after 1s again appear on the RxD pin followed by another break character. Reset clears the BKF bit.

- 1 = Break character detected
- 0 = No break character detected

RPF — Reception in Progress Flag Bit

This read-only bit is set when the receiver detects a 0 during the RT1 time period of the start bit search. RPF does not generate an interrupt request. RPF is reset after the receiver detects false start bits (usually from noise or a baud rate mismatch), or when the receiver detects an idle character. Polling RPF before disabling the ESCI module or entering stop mode can show whether a reception is in progress.

- 1 = Reception in progress
- 0 = No reception in progress

14.8.6 ESCI Data Register

The ESCI data register (SCDR) is the buffer between the internal data bus and the receive and transmit shift registers. Reset has no effect on data in the ESCI data register.

Address: \$0018

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	R7	R6	R5	R4	R3	R2	R1	R0
Write:	T7	T6	T5	T4	T3	T2	T1	T0
Reset:	Unaffected by reset							

Figure 14-16. ESCI Data Register (SCDR)

R7/T7:R0/T0 — Receive/Transmit Data Bits

Reading address \$0018 accesses the read-only received data bits, R7:R0. Writing to address \$0018 writes the data to be transmitted, T7:T0. Reset has no effect on the ESCI data register.

NOTE

Do not use read-modify-write instructions on the ESCI data register.

14.8.7 ESCI Baud Rate Register

The ESCI baud rate register (SCBR) together with the ESCI prescaler register selects the baud rate for both the receiver and the transmitter.

NOTE

There are two prescalers available to adjust the baud rate. One in the ESCI baud rate register and one in the ESCI prescaler register.

Address: \$0019

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	LINT	LINR	SCP1	SCP0	R	SCR2	SCR1	SCR0
Write:								
Reset:	0	0	0	0	0	0	0	0

R = Reserved

Figure 14-17. ESCI Baud Rate Register (SCBR)

LINT — LIN Transmit Enable

This read/write bit selects the enhanced ESCI features for the local interconnect network (LIN) protocol as shown in Table 14-6. Reset clears LINT.

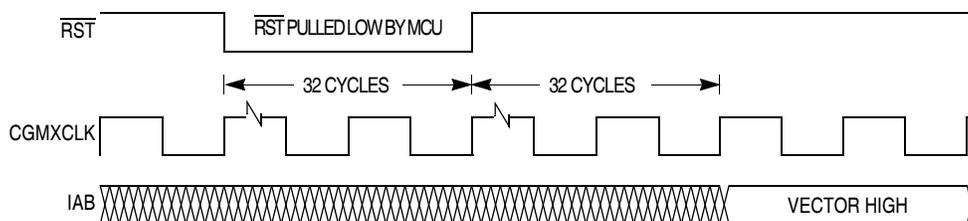


Figure 15-5. Internal Reset Timing

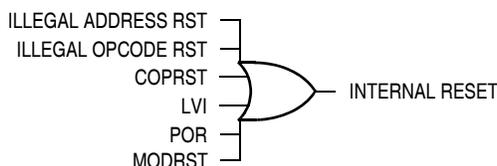


Figure 15-6. Sources of Internal Reset

Table 15-2. Reset Recovery

Reset Recovery Type	Actual Number of Cycles
POR/LVI	4163 (4096 + 64 + 3)
All others	67 (64 + 3)

15.3.2.1 Power-On Reset

When power is first applied to the MCU, the power-on reset module (POR) generates a pulse to indicate that power-on has occurred. The external reset pin (\overline{RST}) is held low while the SIM counter counts out 4096 + 32 CGMXCLK cycles. Thirty-two CGMXCLK cycles later, the CPU and memories are released from reset to allow the reset vector sequence to occur.

At power-on, these events occur:

- A POR pulse is generated.
- The internal reset signal is asserted.
- The SIM enables CGMOUT.
- Internal clocks to the CPU and modules are held inactive for 4096 CGMXCLK cycles to allow stabilization of the oscillator.
- The \overline{RST} pin is driven low during the oscillator stabilization time.
- The POR bit of the SIM reset status register (SRSR) is set.

15.3.2.2 Computer Operating Properly (COP) Reset

An input to the SIM is reserved for the COP reset signal. The overflow of the COP counter causes an internal reset and sets the COP bit in the SIM reset status register (SRSR) if the COPD bit in the CONFIG1 register is cleared. The SIM actively pulls down the \overline{RST} pin for all internal reset sources.

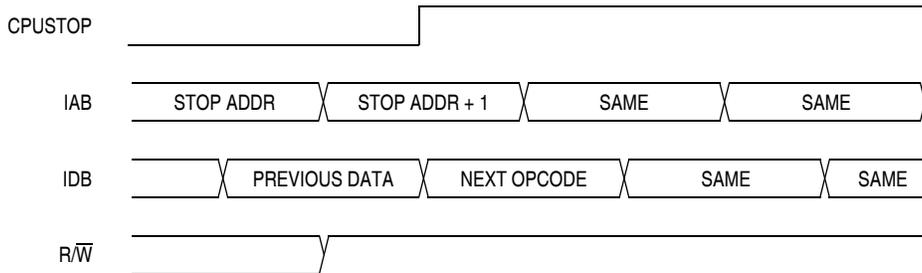
The COP module is disabled if the \overline{RST} pin or the \overline{IRQ} pin is held at V_{TST} while the MCU is in monitor mode. During a break state, V_{TST} on the \overline{RST} pin disables the COP module.

System Integration Module (SIM)

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 15-19 shows stop mode entry timing. Figure 15-20 shows stop mode recovery time from interrupt.

NOTE

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



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

Figure 15-19. Stop Mode Entry Timing

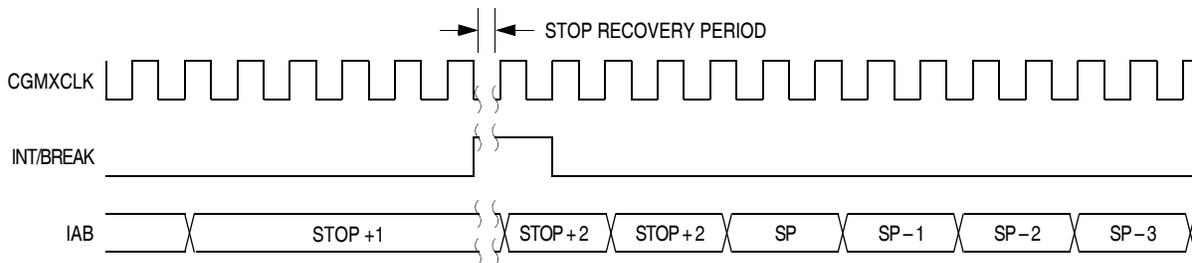


Figure 15-20. Stop Mode Recovery from Interrupt

15.7 SIM Registers

The SIM has three memory-mapped registers. Table 15-4 shows the mapping of these registers.

Table 15-4. SIM Registers

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

19.3.3.1 Unbuffered Output Compare

Any output compare channel can generate unbuffered output compare pulses as described in 19.3.3 Output Compare. The pulses are unbuffered because changing the output compare value requires writing the new value over the old value currently in the TIM2 channel registers.

An unsynchronized write to the TIM2 channel registers to change an output compare value could cause incorrect operation for up to two counter overflow 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 counter overflow period. Also, using a TIM2 overflow interrupt routine to write a new, smaller output compare value may cause the compare to be missed. The TIM2 may pass the new value before it is written.

Use the following methods to synchronize unbuffered changes in the output compare value on channel x:

- When changing to a smaller value, 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 output compare pulse. The interrupt routine has until the end of the counter overflow period to write the new value.
- When changing to a larger output compare value, enable TIM2 overflow interrupts and write the new value in the TIM2 overflow interrupt routine. The TIM2 overflow interrupt occurs at the end of the current counter overflow 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 counter overflow period.

19.3.3.2 Buffered Output Compare

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

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

Channels 2 and 3 can be linked to form a buffered output compare channel whose output appears on the T2CH2 pin. The TIM2 channel registers of the linked pair alternately control the output.

Setting the MS2B bit in TIM2 channel 2 status and control register (T2SC2) links channel 2 and channel 3. The output compare value in the TIM2 channel 2 registers initially controls the output on the T2CH2 pin. Writing to the TIM2 channel 3 registers enables the TIM2 channel 3 registers to synchronously control the output after the TIM2 overflows. At each subsequent overflow, the TIM2 channel registers (2 or 3) that control the output are the ones written to last. T2SC2 controls and monitors the buffered output compare function, and TIM2 channel 3 status and control register (T2SC3) is unused. While the MS2B bit is set, the channel 3 pin, T2CH3, is available as a general-purpose I/O pin.

Channels 4 and 5 can be linked to form a buffered output compare channel whose output appears on the T2CH4 pin. The TIM2 channel registers of the linked pair alternately control the output.

Setting the MS4B bit in TIM2 channel 4 status and control register (T2SC4) links channel 4 and channel 5. The output compare value in the TIM2 channel 4 registers initially controls the output on the T2CH4 pin.

20.3.1.1 Normal Monitor Mode

If V_{TST} is applied to \overline{IRQ} and PTB4 is low upon monitor mode entry, the bus frequency is a divide-by-two of the input clock. If PTB4 is high with V_{TST} applied to \overline{IRQ} upon monitor mode entry, the bus frequency will be a divide-by-four of the input clock. Holding the PTB4 pin low when entering monitor mode causes a bypass of a divide-by-two stage at the oscillator *only if V_{TST} is applied to \overline{IRQ}* . In this event, the CGMOUT frequency is equal to the CGMXCLK frequency, and the OSC1 input directly generates internal bus clocks. In this case, the OSC1 signal must have a 50% duty cycle at maximum bus frequency.

When monitor mode was entered with V_{TST} on \overline{IRQ} , the computer operating properly (COP) is disabled as long as V_{TST} is applied to either \overline{IRQ} or \overline{RST} .

This condition states that as long as V_{TST} is maintained on the \overline{IRQ} pin after entering monitor mode, or if V_{TST} is applied to \overline{RST} after the initial reset to get into monitor mode (when V_{TST} was applied to \overline{IRQ}), then the COP will be disabled. In the latter situation, after V_{TST} is applied to the \overline{RST} pin, V_{TST} can be removed from the \overline{IRQ} pin in the interest of freeing the \overline{IRQ} for normal functionality in monitor mode.

20.3.1.2 Forced Monitor Mode

If entering monitor mode without high voltage on \overline{IRQ} , then all port B pin requirements and conditions, including the PTB4 frequency divisor selection, are not in effect. This is to reduce circuit requirements when performing in-circuit programming.

NOTE

If the reset vector is blank and monitor mode is entered, the chip will see an additional reset cycle after the initial power-on reset (POR). Once the reset vector has been programmed, the traditional method of applying a voltage, V_{TST} , to \overline{IRQ} must be used to enter monitor mode.

An external oscillator of 8 MHz is required for a baud rate of 7200, as the internal bus frequency is automatically set to the external frequency divided by four.

When the forced monitor mode is entered the COP is always disabled regardless of the state of \overline{IRQ} or \overline{RST} .

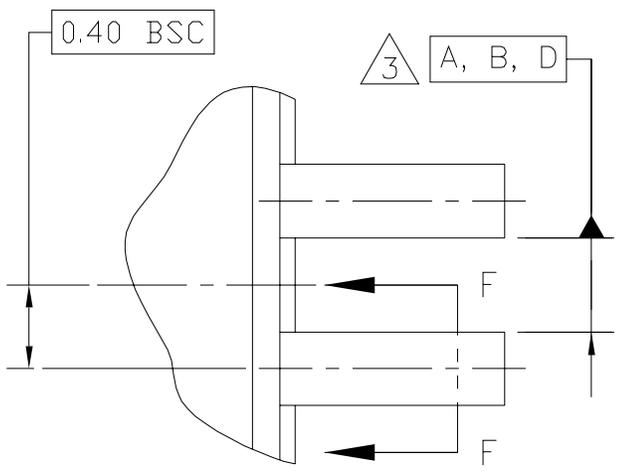
20.3.1.3 Monitor Vectors

In monitor mode, the MCU uses different vectors for reset, SWI (software interrupt), and break interrupt than those for user mode. The alternate vectors are in the \$FE page instead of the \$FF page and allow code execution from the internal monitor firmware instead of user code.

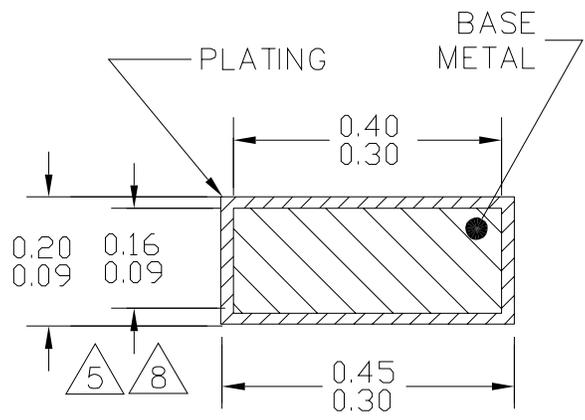
Table 20-2 summarizes the differences between user mode and monitor mode.

Table 20-2. Mode Differences

Modes	Functions					
	Reset Vector High	Reset Vector Low	Break Vector High	Break Vector Low	SWI Vector High	SWI Vector Low
User	\$FFFE	\$FFFF	\$FFFC	\$FFFD	\$FFFC	\$FFFD
Monitor	\$FEFE	\$FEFF	\$FEFC	\$FEFD	\$FEFC	\$FEFD

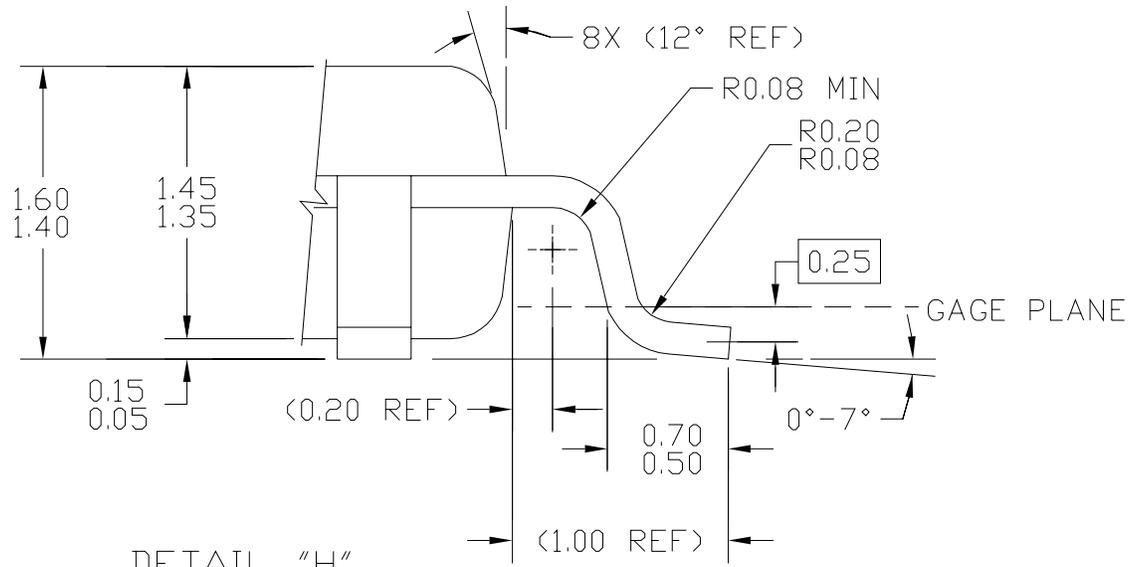


DETAIL G



M C A-B D

SECTION F-F
ROTATED 90°CW
32 PLACES



DETAIL "H"

© FREESCALE SEMICONDUCTOR, INC. ALL RIGHTS RESERVED.	MECHANICAL OUTLINE	PRINT VERSION NOT TO SCALE	
TITLE: LOW PROFILE QUAD FLAT PACK (LQFP) 32 LEAD, 0.8 PITCH (7 X 7 X 1.4)	DOCUMENT NO: 98ASH70029A	REV: C	
	CASE NUMBER: 873A-04	01 APR 2005	
	STANDARD: JEDEC MS-026 BBA		