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	-
Peripherals	LVD, POR, PWM
Number of I/O	13
Program Memory Size	1.5KB (1.5K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 4x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	16-DIP (0.300", 7.62mm)
Supplier Device Package	16-PDIP
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc68hc908qy2cpe

- On-chip in-application programmable FLASH memory (with internal program/erase voltage generation)
  - MC68HC908QY4 and MC68HC908QT4 — 4096 bytes
  - MC68HC908QY2, MC68HC908QY1, MC68HC908QT2, and MC68HC908QT1 — 1536 bytes
- 128 bytes of on-chip random-access memory (RAM)
- 2-channel, 16-bit timer interface module (TIM)
- 4-channel, 8-bit analog-to-digital converter (ADC) on MC68HC908QY2, MC68HC908QY4, MC68HC908QT2, and MC68HC908QT4
- 5 or 13 bidirectional input/output (I/O) lines and one input only:
  - Six shared with keyboard interrupt function and ADC
  - Two shared with timer channels
  - One shared with external interrupt (IRQ)
  - Eight extra I/O lines on 16-pin package only
  - High current sink/source capability on all port pins
  - Selectable pullups on all ports, selectable on an individual bit basis
  - Three-state ability on all port pins
- 6-bit keyboard interrupt with wakeup feature (KBI)
- Low-voltage inhibit (LVI) module features:
  - Software selectable trip point in CONFIG register
- System protection features:
  - Computer operating properly (COP) watchdog
  - Low-voltage detection with reset
  - Illegal opcode detection with reset
  - Illegal address detection with reset
- External asynchronous interrupt pin with internal pullup ($\overline{\text{IRQ}}$) shared with general-purpose input pin
- Master asynchronous reset pin ($\overline{\text{RST}}$) shared with general-purpose input/output (I/O) pin
- Power-on reset
- Internal pullups on $\overline{\text{IRQ}}$ and $\overline{\text{RST}}$ to reduce external components
- Memory mapped I/O registers
- Power saving stop and wait modes
- MC68HC908QY4, MC68HC908QY2, and MC68HC908QY1 are available in these packages:
  - 16-pin plastic dual in-line package (PDIP)
  - 16-pin small outline integrated circuit (SOIC) package
  - 16-pin thin shrink small outline package (TSSOP)
- MC68HC908QT4, MC68HC908QT2, and MC68HC908QT1 are available in these packages:
  - 8-pin PDIP
  - 8-pin SOIC
  - 8-pin dual flat no lead (DFN) package

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$FE08	FLASH Control Register (FLCR) See page 34.	Read:	0	0	0	0	HVEN	MASS	ERASE	PGM
		Write:					HVEN	MASS	ERASE	PGM
		Reset:	0	0	0	0	0	0	0	0
$FE09	Break Address High Register (BRKH) See page 136.	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Reset:	0	0	0	0	0	0	0	0
$FE0A	Break Address low Register (BRKL) See page 136.	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0
$FE0B	Break Status and Control Register (BRKSCR) See page 136.	Read:	BRKE	BRKA	0	0	0	0	0	0
		Write:	BRKE	BRKA						
		Reset:	0	0	0	0	0	0	0	0
$FE0C	LVI Status Register (LVISR) See page 87.	Read:	LVIOUT	0	0	0	0	0	0	R
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$FE0D ↓ $FE0F	Reserved for FLASH Test		R	R	R	R	R	R	R	R
$FFBE	FLASH Block Protect Register (FLBPR) See page 39.	Read:	BPR7	BPR6	BPR5	BPR4	BPR3	BPR2	BPR1	BPR0
		Write:	BPR7	BPR6	BPR5	BPR4	BPR3	BPR2	BPR1	BPR0
		Reset:	Unaffected by reset							
$FFBF	Reserved		R	R	R	R	R	R	R	R
$FFC0	Internal Oscillator Trim (Factory Programmed, VDD = 5.0 V)	Read:	TRIM7	TRIM6	TRIM5	TRIM4	TRIM3	TRIM2	TRIM1	TRIM0
		Write:	TRIM7	TRIM6	TRIM5	TRIM4	TRIM3	TRIM2	TRIM1	TRIM0
		Reset:	Unaffected by reset							
$FFC1	Internal Oscillator Trim (Factory Programmed, VDD = 3.0 V)	Read:	TRIM7	TRIM6	TRIM5	TRIM4	TRIM3	TRIM2	TRIM1	TRIM0
		Write:	TRIM7	TRIM6	TRIM5	TRIM4	TRIM3	TRIM2	TRIM1	TRIM0
		Reset:	Unaffected by reset							
$FFFF	COP Control Register (COPCTL) See page 59.	Read:	LOW BYTE OF RESET VECTOR							
		Write:	WRITING CLEARS COP COUNTER (ANY VALUE)							
		Reset:	Unaffected by reset							

= Unimplemented   R = Reserved   U = Unaffected

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

2.6 FLASH Memory (FLASH)

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

The FLASH memory consists of an array of 4096 or 1536 bytes with an additional 48 bytes for user vectors. The minimum size of FLASH memory that can be erased is 64 bytes; and the maximum size of FLASH memory that can be programmed in a program cycle is 32 bytes (a row). Program and erase operations are facilitated through control bits in the FLASH control register (FLCR). Details for these operations appear later in this section. The address ranges for the user memory and vectors are:

- $EE00 – $FDFF; user memory, 4096 bytes: MC68HC908QY4 and MC68HC908QT4
- $F800 – $FDFF; user memory, 1536 bytes: MC68HC908QY2, MC68HC908QT2, MC68HC908QY1 and MC68HC908QT1
- $FFD0 – $FFFF; user interrupt vectors, 48 bytes.

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

2.6.1 FLASH Control Register

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

Address:	$FE08							
	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	HVEN	MASS	ERASE	PGM
Write:					HVEN	MASS	ERASE	PGM
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 2-3. FLASH Control Register (FLCR)

HVEN — High Voltage Enable Bit
This read/write bit enables high voltage from the charge pump to the memory for either program or erase operation. It can only be set if either PGM =1 or ERASE =1 and the proper sequence for program or erase is followed.
    1 = High voltage enabled to array and charge pump on
    0 = High voltage disabled to array and charge pump off

MASS — Mass Erase Control Bit
This read/write bit configures the memory for mass erase operation.
    1 = Mass erase operation selected
    0 = Mass erase operation unselected

---

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

2.6.3 FLASH Mass Erase Operation

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

1. Set both the ERASE bit and the MASS bit in the FLASH control register.
2. Read the FLASH block protect register.
3. Write any data to any FLASH address[1] within the FLASH memory address range.
4. Wait for a time, t_{NVS} (minimum 10 μs).
5. Set the HVEN bit.
6. Wait for a time, t_{MErase} (minimum 4 ms).
7. Clear the ERASE and MASS bits.

NOTE

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

8. Wait for a time, t_{NVHL} (minimum 100 μs).
9. Clear the HVEN bit.
10. After time, t_{RCV} (typical 1 μs), the memory can be accessed in read mode again.

NOTE

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

CAUTION

A mass erase will erase the internal oscillator trim values at $FFC0 and $FFC1.

2.6.4 FLASH Program Operation

Programming of the FLASH memory is done on a row basis. A row consists of 32 consecutive bytes starting from addresses $XX00, $XX20, $XX40, $XX60, $XX80, $XXA0, $XXC0, or $XXE0. 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[2].

---

1. When in monitor mode, with security sequence failed (see 15.3.2 Security), write to the FLASH block protect register instead of any FLASH address.

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 16.16 Memory Characteristics.*

2.6.5 FLASH Protection

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

> **NOTE**
> *In performing a program or erase operation, the FLASH block protect register must be read after setting the PGM or ERASE bit and before asserting the HVEN bit.*
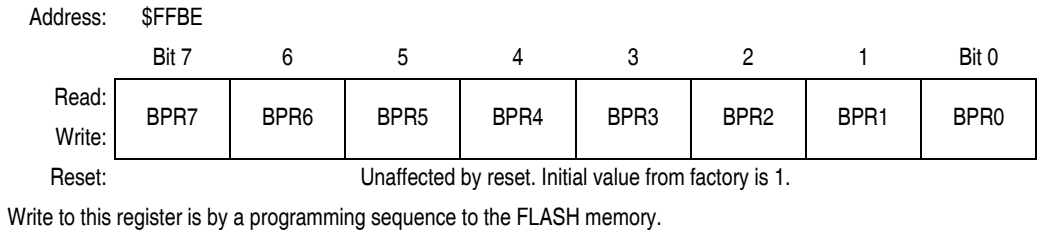
When the FLBPR is programmed with all 0 s, the entire memory is protected from being programmed and erased. When all the bits are erased (all 1's), the entire memory is accessible for program and erase.

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

---

2. 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.

2.6.6 FLASH Block Protect Register

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

Address: $FFBE

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

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

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

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

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

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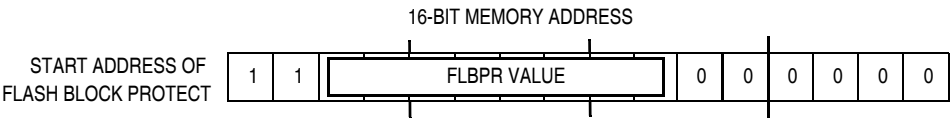
Figure 2-6. FLASH Block Protect Start Address

Table 2-2. Examples of Protect Start Address

BPR[7:0]	Start of Address of Protect Range
$00–$B8	The entire FLASH memory is protected.
$B9 (**1011 1001**)	$EE40 (11**10 1110 01**00 0000)
$BA (**1011 1010**)	$EE80 (11**10 1110 10**00 0000)
$BB (**1011 1011**)	$EEC0 (11**10 1110 11**00 0000)
$BC (**1011 1100**)	$EF00 (11**10 1111 00**00 0000)
and so on...	
$DE (**1101 1110**)	$F780 (11**11 0111 10**00 0000)
$DF (**1101 1111**)	$F7C0 (11**11 0111 11**00 0000)
$FE (**1111 1110**)	$FF80 (11**11 1111 10**00 0000) FLBPR, internal oscillator trim values, and vectors are protected
$FF	The entire FLASH memory is not protected.

LVIPWRD — LVI Power Disable Bit

LVIPWRD disables the LVI module.

    1 = LVI module power disabled

    0 = LVI module power enabled

LVI5OR3 — LVI 5-V or 3-V Operating Mode Bit

LVI5OR3 selects the voltage operating mode of the LVI module. The voltage mode selected for the LVI should match the operating V_{DD} for the LVI's voltage trip points for each of the modes.

    1 = LVI operates in 5-V mode

    0 = LVI operates in 3-V mode

NOTE

The LVI5OR3 bit is cleared by a power-on reset (POR) only. Other resets will leave this bit unaffected.

SSREC — Short Stop Recovery Bit

SSREC enables the CPU to exit stop mode with a delay of 32 BUSCLKX4 cycles instead of a 4096 BUSCLKX4 cycle delay.

    1 = Stop mode recovery after 32 BUSCLKX4 cycles

    0 = Stop mode recovery after 4096 BUSCLKX4 cycles

NOTE

Exiting stop mode by an LVI reset will result in the long stop recovery.

The system stabilization time for power-on reset and long stop recovery (both 4096 BUSCLKX4 cycles) gives a delay longer than the LVI enable time for these startup scenarios. There is no period where the MCU is not protected from a low-power condition. However, when using the short stop recovery configuration option, the 32 BUSCLKX4 delay must be greater than the LVI's turn on time to avoid a period in startup where the LVI is not protecting the MCU.

STOP — STOP Instruction Enable Bit

STOP enables the STOP instruction.

    1 = STOP instruction enabled

    0 = STOP instruction treated as illegal opcode

COPD — COP Disable Bit

COPD disables the COP module.

    1 = COP module disabled

    0 = COP module enabled

7.3.3 Stack Pointer

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

In the stack pointer 8-bit offset and 16-bit offset addressing modes, the stack pointer can function as an index register to access data on the stack. The CPU uses the contents of the stack pointer to determine the conditional address of the operand.
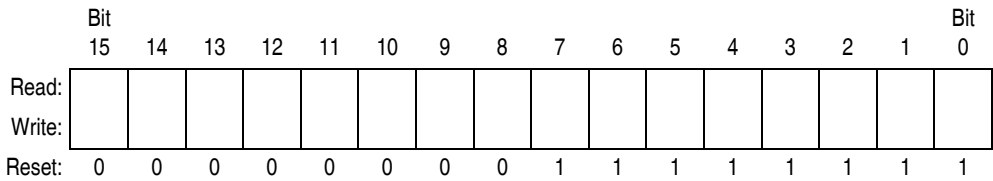
	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit 0
Read:																
Write:																
Reset:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

Figure 7-4. Stack Pointer (SP)

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

7.3.4 Program Counter

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

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

During reset, the program counter is loaded with the reset vector address located at $FFFE and $FFFF. The vector address is the address of the first instruction to be executed after exiting the reset state.
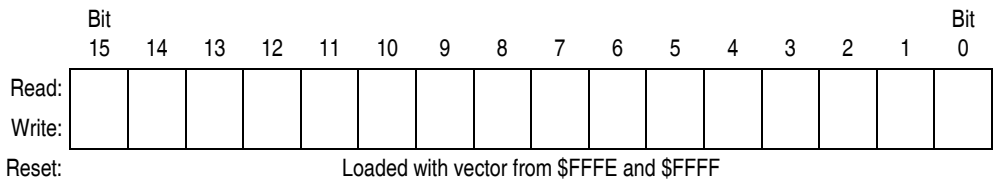
	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit 0
Read:																
Write:																
Reset:					Loaded with vector from $FFFE and $FFFF											

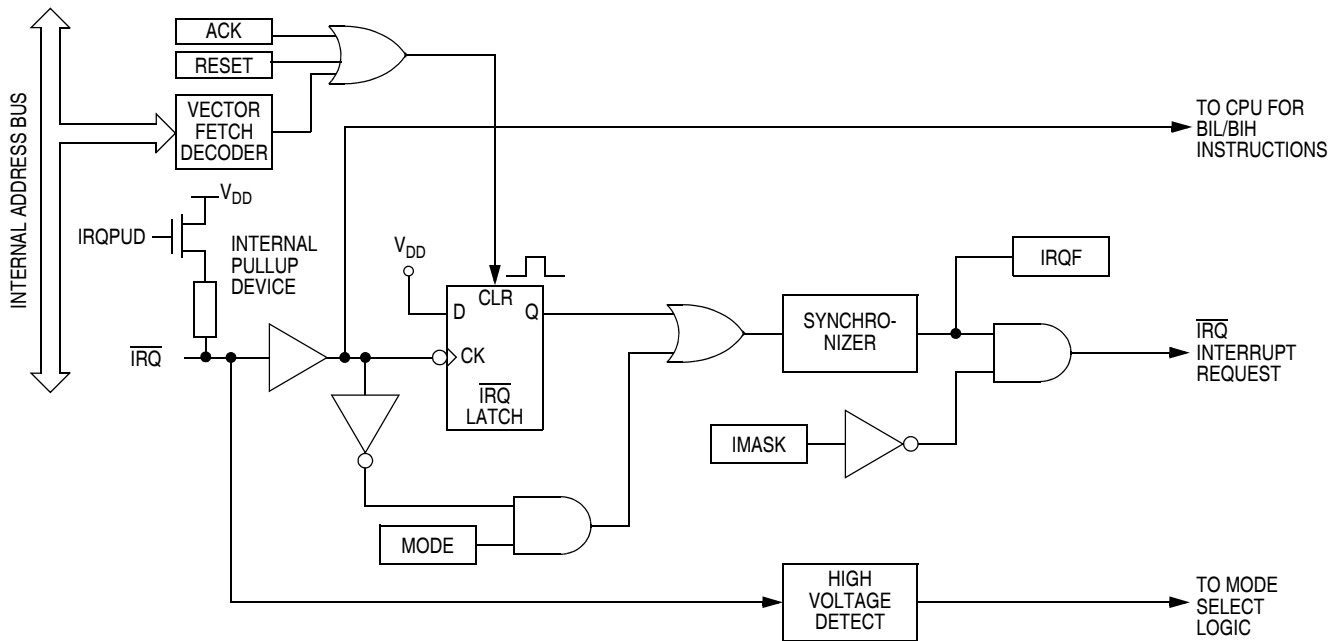**Figure 7-5. Program Counter (PC)**

Figure 8-2. IRQ Module Block Diagram

8.3.1 MODE = 1

If the MODE bit is set, the $\overline{\text{IRQ}}$ pin is both falling edge sensitive and low level sensitive. With MODE set, both of the following actions must occur to clear the $\overline{\text{IRQ}}$ interrupt request:

- Return of the $\overline{\text{IRQ}}$ pin to a high level. As long as the $\overline{\text{IRQ}}$ pin is low, the IRQ request remains active.
- IRQ vector fetch or software clear. An IRQ vector fetch generates an interrupt acknowledge signal to clear the IRQ latch. Software generates the interrupt acknowledge signal by writing a 1 to ACK in INTSCR. The ACK bit is useful in applications that poll the $\overline{\text{IRQ}}$ pin and require software to clear the IRQ latch. Writing to ACK prior to leaving an interrupt service routine can also prevent spurious interrupts due to noise. Setting ACK does not affect subsequent transitions on the $\overline{\text{IRQ}}$ pin. A falling edge that occurs after writing to ACK latches another interrupt request. If the IRQ mask bit, IMASK, is clear, the CPU loads the program counter with the IRQ vector address.

The IRQ vector fetch or software clear and the return of the $\overline{\text{IRQ}}$ pin to a high level may occur in any order. The interrupt request remains pending as long as the $\overline{\text{IRQ}}$ pin is low. A reset will clear the IRQ latch and the MODE control bit, thereby clearing the interrupt even if the pin stays low.

Use the BIH or BIL instruction to read the logic level on the $\overline{\text{IRQ}}$ pin.

8.3.2 MODE = 0

If the MODE bit is clear, the $\overline{\text{IRQ}}$ pin is falling edge sensitive only. With MODE clear, an IRQ vector fetch or software clear immediately clears the IRQ latch.

The IRQF bit in INTSCR can be read to check for pending interrupts. The IRQF bit is not affected by IMASK, which makes it useful in applications where polling is preferred.

> ***NOTE***
> *When using the level-sensitive interrupt trigger, avoid false IRQ interrupts*
> *by masking interrupt requests in the interrupt routine.*

MC68HC908QY/QT Family Data Sheet, Rev. 6

8.7.1 IRQ Input Pins (IRQ)

The IRQ pin provides a maskable external interrupt source. The IRQ pin contains an internal pullup device.

8.8 Registers

The IRQ status and control register (INTSCR) controls and monitors operation of the IRQ module. See Chapter 5 Configuration Register (CONFIG).

The INTSCR has the following functions:
- Shows the state of the IRQ flag
- Clears the IRQ latch
- Masks the IRQ interrupt request
- Controls triggering sensitivity of the IRQ interrupt pin

Address: $001D

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	IRQF	0	IMASK	MODE
Write:						ACK	IMASK	MODE
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 8-3. IRQ Status and Control Register (INTSCR)

IRQF — IRQ Flag
This read-only status bit is set when the IRQ interrupt is pending.
1 = IRQ interrupt pending
0 = IRQ interrupt not pending

ACK — IRQ Interrupt Request Acknowledge Bit
Writing a 1 to this write-only bit clears the IRQ latch. ACK always reads as 0.

IMASK — IRQ Interrupt Mask Bit
Writing a 1 to this read/write bit disables the IRQ interrupt request.
1 = IRQ interrupt request disabled
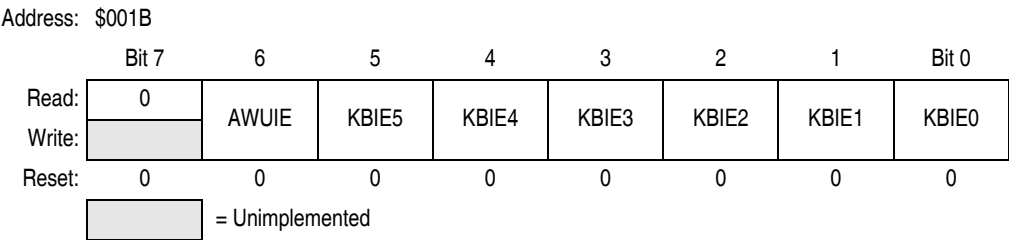0 = IRQ interrupt request enabled

MODE — IRQ Edge/Level Select Bit
This read/write bit controls the triggering sensitivity of the IRQ pin.
1 = IRQ interrupt request on falling edges and low levels
0 = IRQ interrupt request on falling edges only

9.7.2 Keyboard Interrupt Enable Register

The port A keyboard interrupt enable register (KBIER) enables or disables each port A pin or auto wakeup to operate as a keyboard interrupt input.

Address: $001B

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	AWUIE	KBIE5	KBIE4	KBIE3	KBIE2	KBIE1	KBIE0
Write:		AWUIE	KBIE5	KBIE4	KBIE3	KBIE2	KBIE1	KBIE0
Reset:	0	0	0	0	0	0	0	0

	= Unimplemented

Figure 9-4. Keyboard Interrupt Enable Register (KBIER)

KBIE5–KBIE0 — Port A Keyboard Interrupt Enable Bits

Each of these read/write bits enables the corresponding keyboard interrupt pin on port A to latch interrupt requests. Reset clears the keyboard interrupt enable register.

1 = KBIx pin enabled as keyboard interrupt pin
0 = KBIx pin not enabled as keyboard interrupt pin

NOTE

AWUIE bit is not used in conjunction with the keyboard interrupt feature. To see a description of this bit, see Chapter 4 Auto Wakeup Module (AWU).

12.3.2 Data Direction Register B

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

Address: $0005

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
Reset:	0	0	0	0	0	0	0	0

Figure 12-6. Data Direction Register B (DDRB)

DDRB[7:0] — Data Direction Register B Bits

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

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

NOTE

Avoid glitches on port B pins by writing to the port B data register before changing data direction register B bits from 0 to 1. Figure 12-7 shows the port B I/O logic.
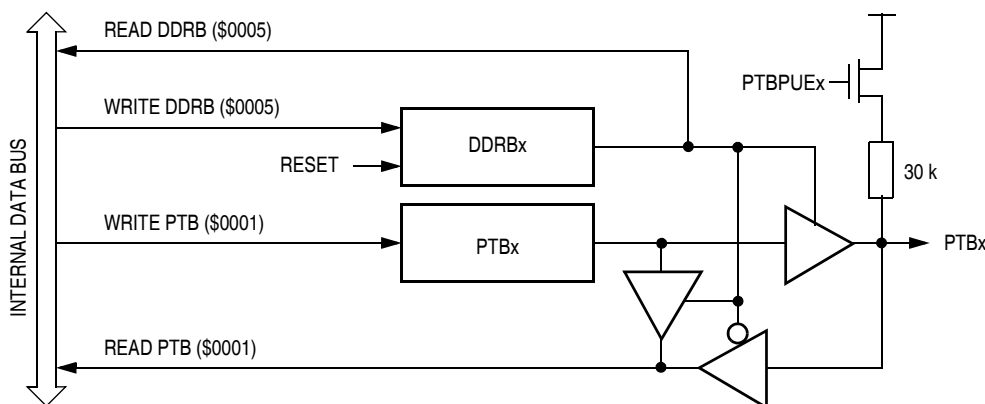

Figure 12-7. Port B I/O Circuit

When DDRBx is a 1, reading address $0001 reads the PTBx data latch. When DDRBx is a 0, reading address $0001 reads the voltage level on the pin. The data latch can always be written, regardless of the state of its data direction bit. Table 12-2 summarizes the operation of the port B pins.
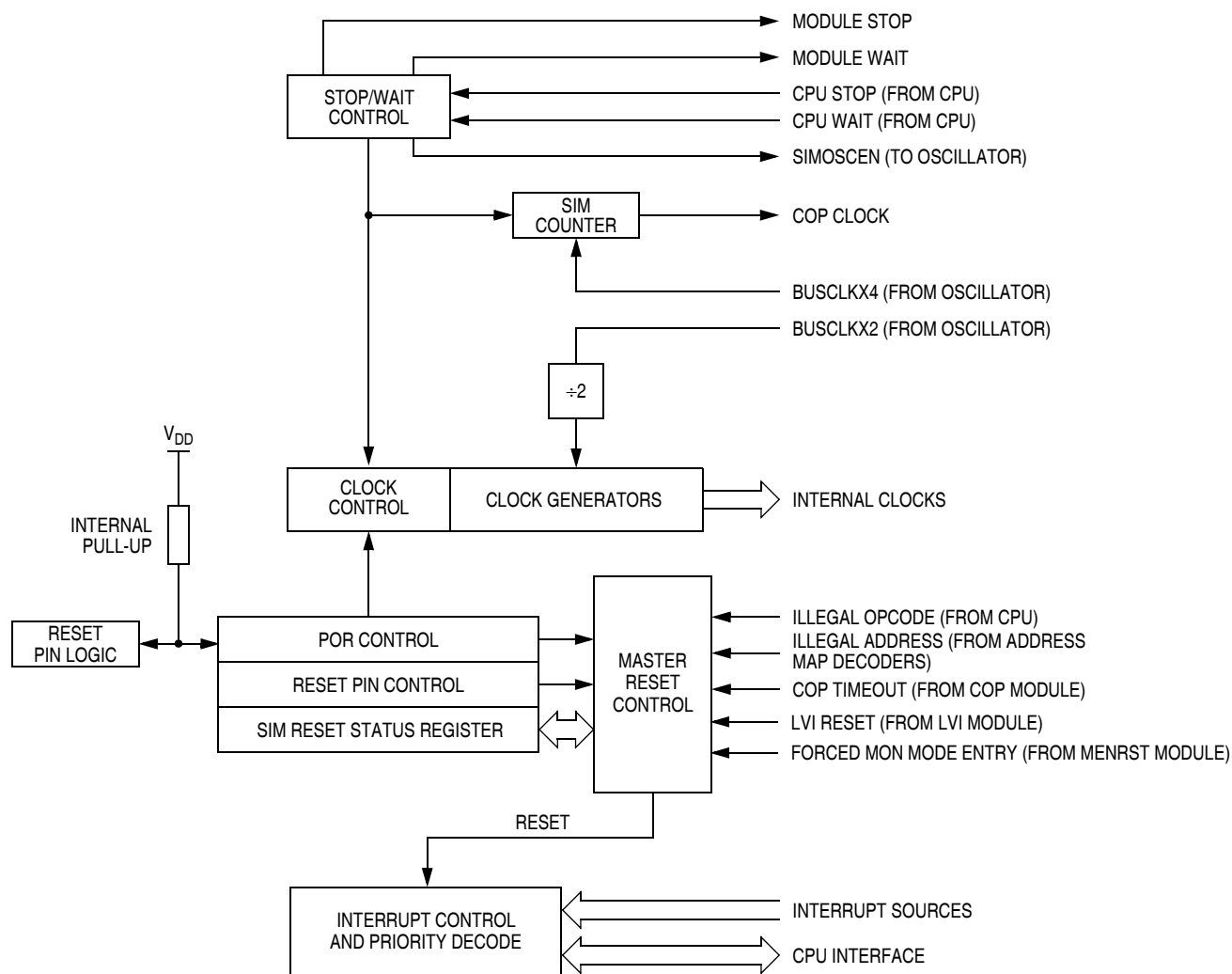
Table 12-2. Port B Pin Functions

DDRB Bit	PTB Bit	I/O Pin Mode	Accesses to DDRB	Accesses to PTB	
			Read/Write	Read	Write
0	X[1]	Input, Hi-Z[2]	DDRB7–DDRB0	Pin	PTB7–PTB0[3]
1	X	Output	DDRB7–DDRB0	Pin	PTB7–PTB0

1. X = don't care
2. Hi-Z = high impedance
3. Writing affects data register, but does not affect the input.

MC68HC908QY/QT Family Data Sheet, Rev. 6

Figure 13-1. SIM Block Diagram

13.2 $\overline{\text{RST}}$ and $\overline{\text{IRQ}}$ Pins Initialization

$\overline{\text{RST}}$ and $\overline{\text{IRQ}}$ pins come out of reset as PTA3 and PTA2 respectively. $\overline{\text{RST}}$ and $\overline{\text{IRQ}}$ functions can be activated by programing CONFIG2 accordingly. Refer to Chapter 5 Configuration Register (CONFIG).

13.3 SIM Bus Clock Control and Generation

The bus clock generator provides system clock signals for the CPU and peripherals on the MCU. The system clocks are generated from an incoming clock, BUSCLKX2, as shown in Figure 13-2.
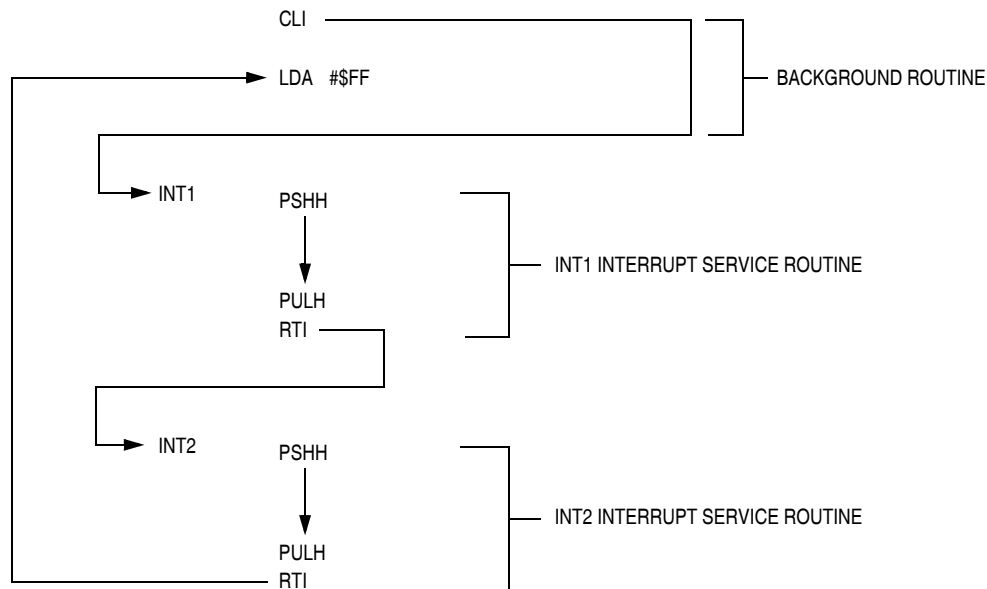
Figure 13-10. Interrupt Recognition Example

13.6.1.2 SWI Instruction

The SWI instruction is a non-maskable instruction that causes an interrupt regardless of the state of the interrupt mask (I bit) in the condition code register.

> **NOTE**
> *A software interrupt pushes PC onto the stack. A software interrupt does*
> **not** *push PC – 1, as a hardware interrupt does.*

13.6.2 Interrupt Status Registers

The flags in the interrupt status registers identify maskable interrupt sources. Table 13-3 summarizes the interrupt sources and the interrupt status register flags that they set. The interrupt status registers can be useful for debugging.

Table 13-3. Interrupt Sources

Priority	Source	Flag	Mask[1]	INT Register Flag	Vector Address
Highest	Reset	—	—	—	$FFFE–$FFFF
↑	SWI instruction	—	—	—	$FFFC–$FFFD
	IRQ pin	IRQF	IMASK	IF1	$FFFA–$FFFB
	Timer channel 0 interrupt	CH0F	CH0IE	IF3	$FFF6–$FFF7
	Timer channel 1 interrupt	CH1F	CH1IE	IF4	$FFF4–$FFF5
	Timer overflow interrupt	TOF	TOIE	IF5	$FFF2–$FFF3
	Keyboard interrupt	KEYF	IMASKK	IF14	$FFE0–$FFE1
Lowest	ADC conversion complete interrupt	COCO	AIEN	IF15	$FFDE–$FFDF

1. The I bit in the condition code register is a global mask for all interrupt sources except the SWI instruction.

16.9 3-V DC Electrical Characteristics

Characteristic[1]	Symbol	Min	Typ[2]	Max	Unit
Output high voltage I_{Load} = –0.6 mA, all I/O pins I_{Load} = –4.0 mA, all I/O pins I_{Load} = –10.0 mA, PTA0, PTA1, PTA3–PTA5 only	V_{OH}	V_{DD}–0.3 V_{DD}–1.0 V_{DD}–0.8	— — —	— — —	V
Maximum combined I_{OH} (all I/O pins)	I_{OHT}	—	—	50	mA
Output low voltage I_{Load} = 0.5 mA, all I/O pins I_{Load} = 6.0 mA, all I/O pins I_{Load} = 10.0 mA, PTA0, PTA1, PTA3–PTA5 only	V_{OL}	— — —	— — —	0.3 1.0 0.8	V
Maximum combined I_{OL} (all I/O pins)	I_{OLT}	—	—	50	mA
Input high voltage PTA0–PTA5, PTB0–PTB7	V_{IH}	0.7 x V_{DD}	—	V_{DD}	V
Input low voltage PTA0–PTA5, PTB0–PTB7	V_{IL}	V_{SS}	—	0.3 x V_{DD}	V
Input hysteresis	V_{HYS}	0.06 x V_{DD}	—	—	V
DC injection current, all ports	I_{INJ}	–2	—	+2	mA
Total dc current injection (sum of all I/O)	I_{INJTOT}	–25	—	+25	mA
Ports Hi-Z leakage current	I_{IL}	–1	±0.1	+1	µA
Capacitance Ports (as input) Ports (as input)	C_{IN} C_{OUT}	— —	— —	12 8	pF
POR rearm voltage[3]	V_{POR}	0	—	100	mV
POR rise time ramp rate[4]	R_{POR}	0.035	—	—	V/ms
Monitor mode entry voltage	V_{TST}	V_{DD} + 2.5	—	V_{DD} + 4.0	V
Pullup resistors[5] PTA0–PTA5, PTB0–PTB7	R_{PU}	16	26	36	kΩ
Low-voltage inhibit reset, trip falling voltage	V_{TRIPF}	2.40	2.55	2.70	V
Low-voltage inhibit reset, trip rising voltage	V_{TRIPR}	2.50	2.65	2.80	V
Low-voltage inhibit reset/recover hysteresis	V_{HYS}	—	60	—	mV

1. V_{DD} = 2.7 to 3.3 Vdc, V_{SS} = 0 Vdc, T_A = T_L to T_H, unless otherwise noted.
2. Typical values reflect average measurements at midpoint of voltage range, 25•C only.
3. Maximum is highest voltage that POR is guaranteed.
4. If minimum V_{DD} is not reached before the internal POR reset is released, the LVI will hold the part in reset until minimum V_{DD} is reached.
5. R_{PU} are measured at V_{DD} = 3.0 V

MC68HC908QY/QT Family Data Sheet, Rev. 6

16.11 3-V Control Timing

Characteristic[1]	Symbol	Min	Max	Unit
Internal operating frequency	f_{OP} (f_{Bus})	—	4	MHz
Internal clock period (1/f_{OP})	t_{cyc}	250	—	ns
$\overline{RST}$ input pulse width low	t_{RL}	200	—	ns
$\overline{IRQ}$ interrupt pulse width low (edge-triggered)	t_{ILIH}	200	—	ns
$\overline{IRQ}$ interrupt pulse period	t_{ILIL}	Note[2]	—	t_{cyc}

1. V_{DD} = 2.7 to 3.3 Vdc, V_{SS} = 0 Vdc, T_A = T_L to T_H; timing shown with respect to 20% V_{DD} and 70% V_{DD}, unless otherwise noted.
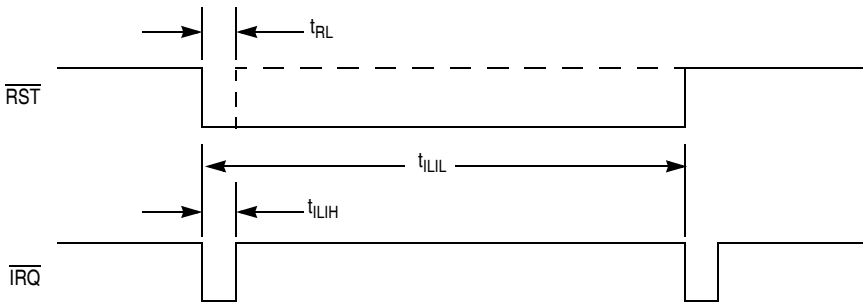2. The minimum period is the number of cycles it takes to execute the interrupt service routine plus 1 t_{cyc}.

Figure 16-7. $\overline{RST}$ and $\overline{IRQ}$ Timing

NOTES:

1. DIMENSIONING AND TOLERANCING PER ASME Y14.5M, 1994.

2. CONTROLLING DIMENSION: INCH.

△3. DIMENSION TO CENTER OF LEADS WHEN FORMED PARALLEL.

△4. DIMENSIONS DOES NOT INCLUDE MOLD FLASH.

5. ROUNDED CORNERS OPTIONAL.

6. 648-01 THRU -08 OBSOLETE, NEW STANDARD 648-09.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX	DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	18.80	19.55	0.740	0.770					
B	6.35	6.85	0.250	0.270					
C	3.69	4.44	0.145	0.175					
D	0.39	0.53	0.015	0.021					
F	1.02	1.77	0.040	0.070					
G	2.54 BSC		0.100 BSC						
H	1.27 BSC		0.050 BSC						
J	0.21	0.38	0.008	0.015					
K	2.80	3.30	0.110	0.130					
L	7.50	7.74	0.295	0.305					
M	0°	10°	0°	10°					
S	0.51	1.01	0.020	0.040					

TITLE: 16 LD PDIP	CASE NUMBER: 648-08
	STANDARD: NON-JEDEC
	PACKAGE CODE: 0006 SHEET: 2 OF 4

PIN'S NUMBER

PIN 1 INDEX

8X 10.55 / 10.05

⊕ 0.25 Ⓜ T

A

1

16

⚠4 10.45 / 10.15

A   A

8

9

7.6 / 7.4

B

⚠5

2.65 / 2.35

0.25 / 0.10

16X 0.49 / 0.35   ⚠6

⊕ 0.25 Ⓜ T A B

14X 1.27

0.635

T
SEATING PLANE

16X ⌒ 0.1 T

0.75 X45° / 0.25

SECTION A-A

0.32 / 0.23

1.0 / 0.4

7° / 0°