Data Sheet Summary

MC68HC908QY4SM/D
9/2002

MC68HC908QY4,
MC68HC908QT4,
MC68HC908QY2,
MC68HC908QT2,
MC68HC908QY1,
MC68HC908QT1

MOTOROLA
intelligence everywhere™

digital dna™

Introduction

This document provides an overview of the MC68HC908QY4, MC68HC908QT4, MC68HC908QY2, MC68HC908QT2, MC68HC908QY1, and MC68HC908QT1 devices. For complete details refer to the *MC68HC908QY4 Data Sheet* (Motorola document order number MC68HC908QY4/D).

General Description

The MC68HC908QY4 is a member of the low-cost, high-performance M68HC08 Family of 8-bit microcontroller units (MCUs). The M68HC08 Family is a Complex Instruction Set Computer (CISC) with a Von Neumann architecture. All MCUs in the family use the enhanced M68HC08 central processor unit (CPU08) and are available with a variety of modules, memory sizes and types, and package types.

Table 1. MC Order Numbers

MC Order Number	ADC	FLASH Memory	Package
MC68HC908QY1	—	1536 bytes	16-pins PDIP, SOIC, and TSSOP
MC68HC908QY2	Yes	1536 bytes	
MC68HC908QY4	Yes	4096 bytes	
MC68HC908QT1	—	1536 bytes	8-pins PDIP or SOIC
MC68HC908QT2	Yes	1536 bytes	
MC68HC908QT4	Yes	4096 bytes	

Temperature and package designators:
C = –40°C to +85°C
V = –40°C to +105°C (available for V_{DD} = 5 V only)
M = –40°C to +125°C (available for V_{DD} = 5 V only)
P = Plastic dual in-line package (PDIP)
DW = Small outline integrated circuit package (SOIC)
DT = Thin shrink small outline package (TSSOP)

Features

Features include:

- High-performance M68HC08 CPU core
- Fully upward-compatible object code with M68HC05 Family
- 5-V and 3-V operating voltages (V_{DD})
- 8-MHz internal bus operation at 5 V, 4-MHz at 3 V
- Trimmable internal oscillator
  - 3.2 MHz internal bus operation
  - 8-bit trim capability, ± 5% trimmed
- Auto wake-up from STOP capability
- Configuration (CONFIG) register for MCU configuration options, including low-voltage inhibit (LVI) trip point
- In-system FLASH programming
- FLASH security(1)
- 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:
  - High current sink/source capability on all port pins
  - Selectable pullups on all ports, selectable on an individual bit basis
- 6-bit keyboard interrupt with wakeup feature (KBI)
- Low-voltage inhibit (LVI) module features software selectable trip point in CONFIG register

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

- 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 ($\overline{IRQ}$) shared with general-purpose input pin
- Master asynchronous reset pin ($\overline{RST}$) shared with general-purpose I/O pin
- Power-on reset
- Internal pullups on $\overline{IRQ}$ and $\overline{RST}$ 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

MCU Block Diagram

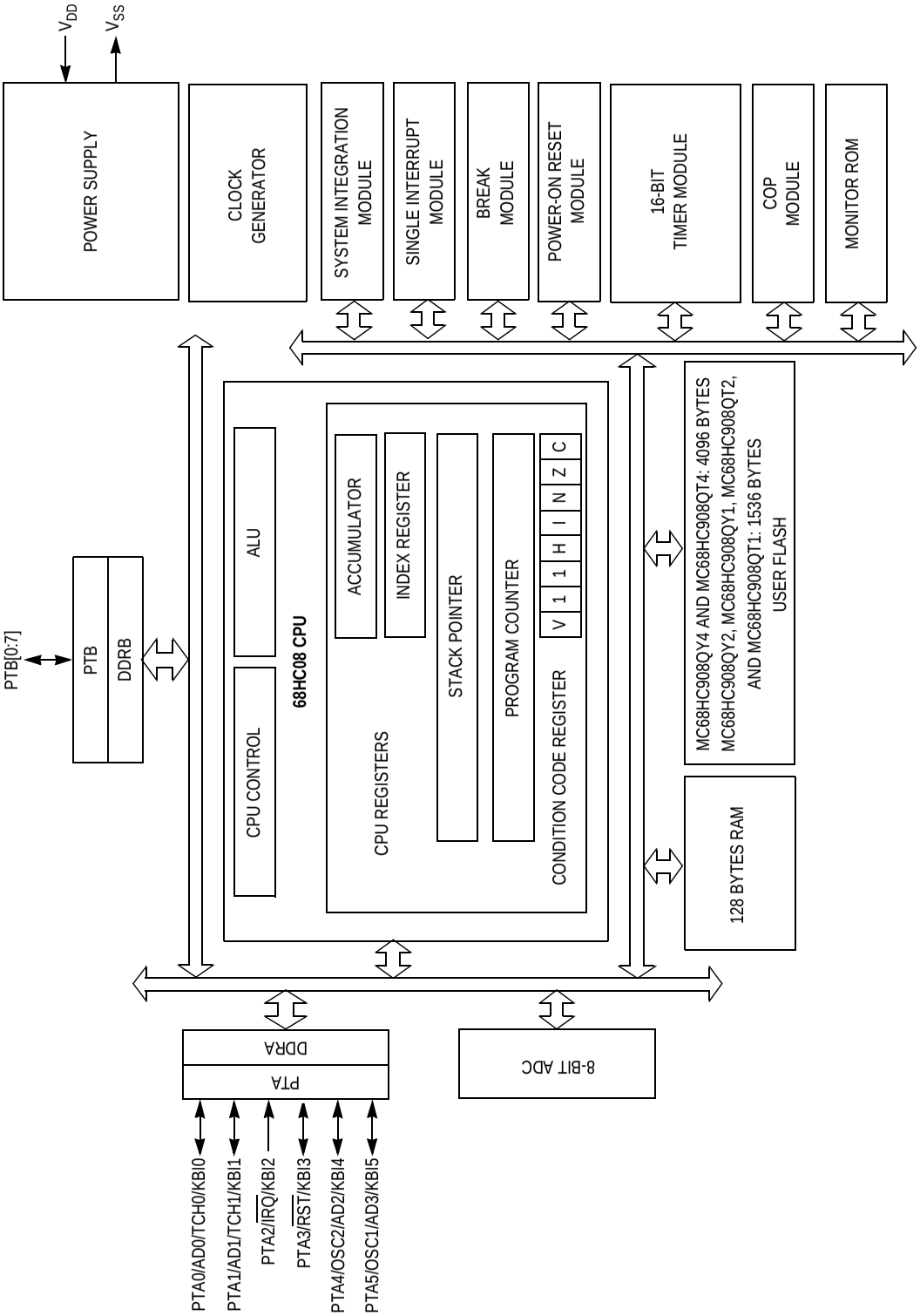
See **Figure 1**.

Memory

The central processor unit (CPU08) can address 64 Kbytes of memory space. The memory map is shown in **Figure 3**.

Addresses $0000–$003F, shown in **Figure 4**, contain most of the control, status, and data registers.

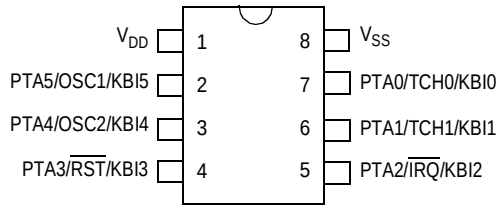
The vector addresses are shown in **Table 3**.



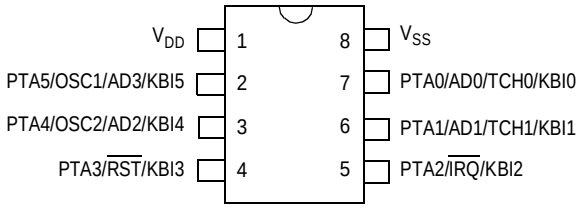
RST, IRQ: Pins have internal (about 30K Ohms) pull up
PTA[0:5]: High current sink and source capability
PTA[0:5]: Pins have programmable keyboard interrupt and pull up
PTB[0:7]: Not available on 8-pin devices – MC68HC908QT1, MC68HC908QT2, and MC68HC908QT4

Figure 1. Block Diagram

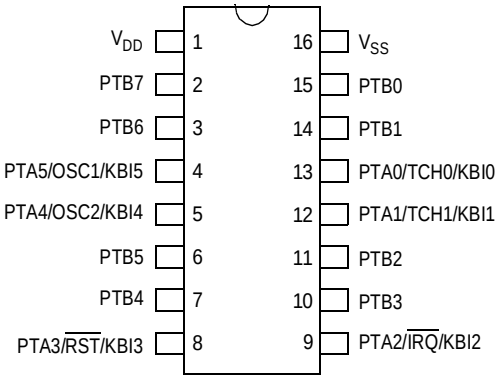
Pin Assignments



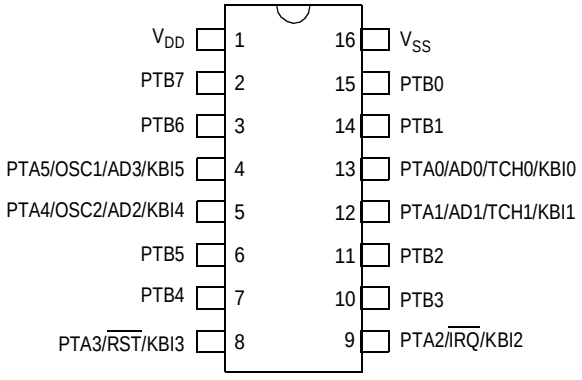
8-PIN ASSIGNMENT
MC68HC908QT1 PDIP/SOIC



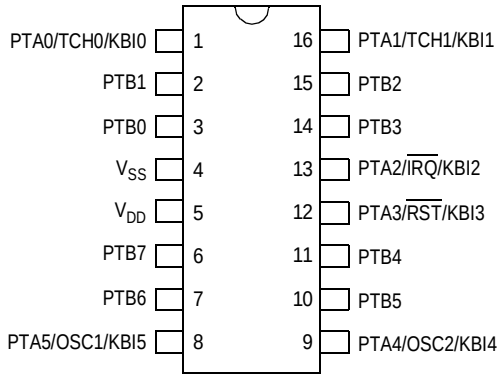
8-PIN ASSIGNMENT
MC68HC908QT2 AND MC68HC908QT4 PDIP/SOIC



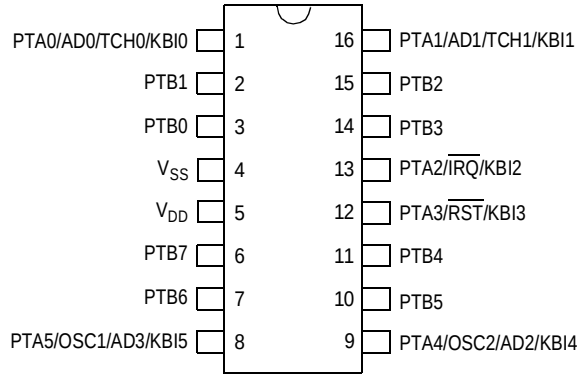
16-PIN ASSIGNMENT
MC68HC908QY1 PDIP/SOIC



16-PIN ASSIGNMENT
MC68HC908QY2 AND MC68HC908QY4 PDIP/SOIC



16-PIN ASSIGNMENT
MC68HC908QY1 TSSOP



16-PIN ASSIGNMENT
MC68HC908QY2 AND MC68HC908QY4 TSSOP

Figure 2. MCU Pin Assignments

Pin Functions

Table 2 provides a description of the pin functions.

Table 2. Pin Functions

Pin Name	Description	Input/Output
V_{DD}	Power supply	Power
V_{SS}	Power supply ground	Power
PTA0	PTA0 — General purpose I/O port	Input/Output
	AD0 — A/D channel 0 input	Input
	TCH0 — Timer Channel 0 I/O	Input/Output
	KBI0 — Keyboard interrupt input 0	Input
PTA1	PTA1 — General purpose I/O port	Input/Output
	AD1 — A/D channel 1 input	Input
	TCH1 — Timer Channel 1 I/O	Input/Output
	KBI1 — Keyboard interrupt input 1	Input
PTA2	PTA2 — General purpose input-only port	Input
	$\overline{IRQ}$ — External interrupt with programmable pullup and Schmitt trigger input	Input
	KBI2 — Keyboard interrupt input 2	Input
PTA3	PTA3 — General purpose I/O port	Input/Output
	$\overline{RST}$ — Reset input, active low with internal pullup and Schmitt trigger	Input
	KBI3 — Keyboard interrupt input 3	Input
PTA4	PTA4 — General purpose I/O port	Input/Output
	OSC2 — XTAL oscillator output (XTAL option only) RC or internal oscillator output (OSC2EN = 1 in PTAPUE register)	Output Output
	AD2 — A/D channel 2 input	Input
	KBI4 — Keyboard interrupt input 4	Input
PTA5	PTA5 — General purpose I/O port	Input/Output
	OSC1 —XTAL, RC, or external oscillator input	Input
	AD3 — A/D channel 3 input	Input
	KBI5 — Keyboard interrupt input 5	Input
PTB[0:7](1)	8 general-purpose I/O ports.	Input/Output

1. The PTB pins are not available on the 8-pin packages.

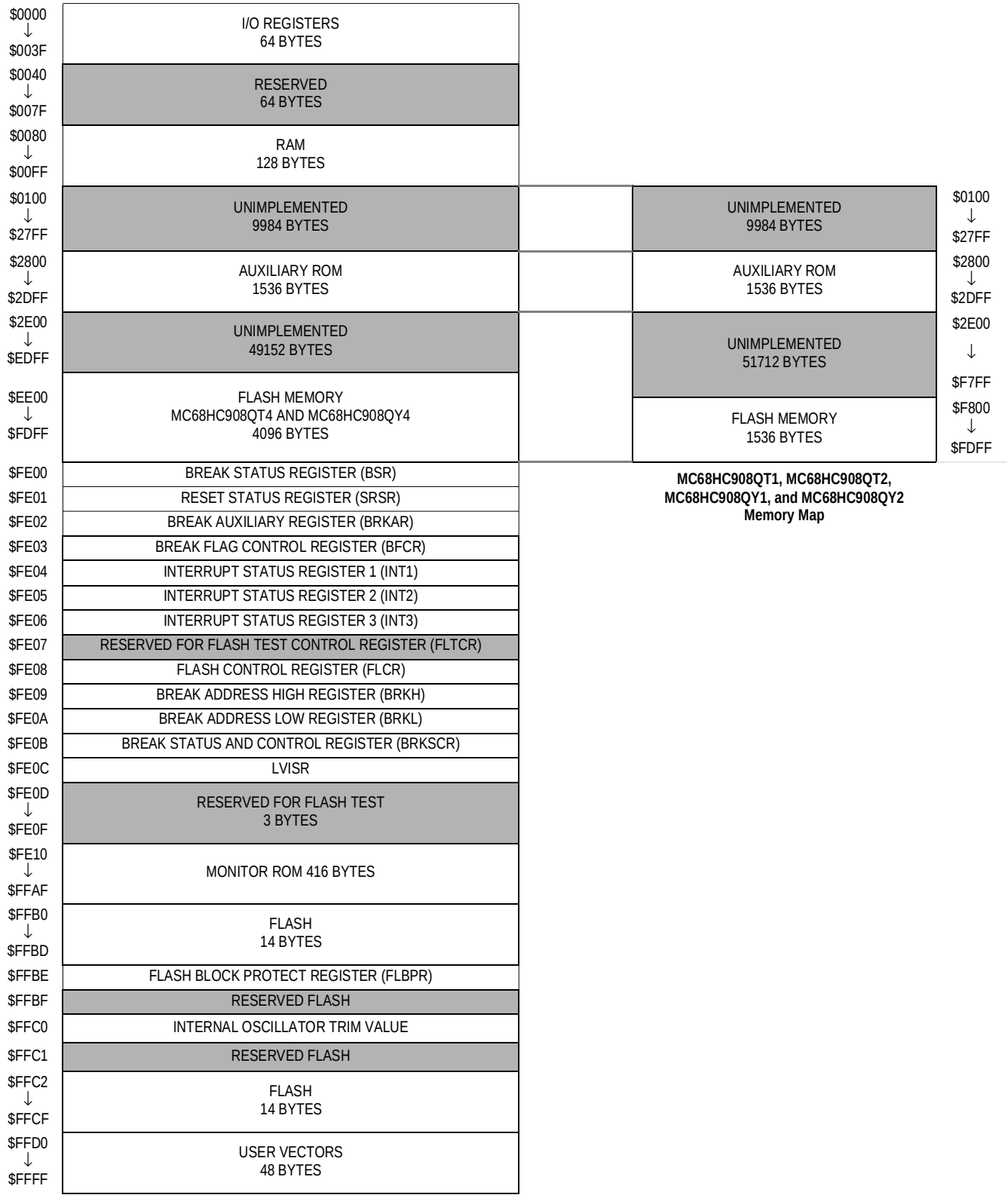


Figure 3. Memory Map

Addr.	Register	Bit 7	6	5	4	3	2	1	Bit 0
$0000	PTA	0	AWUL	PTA5	PTA4	PTA3	PTA2	PTA1	PTA0
$0001	PTB	PTB7	PTB6	PTB5	PTB4	PTB3	PTB2	PTB1	PTB0
$0002	Unimplemented								
$0003	Unimplemented								
$0004	DDRA	0	0	DDRA5	DDRA4	DDRA3	0	DDRA1	DDRA0
$0006	DDRB	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
$0007–$000A	Unimplemented								
	Unimplemented								
$000B	PTAPUE	OSC2EN	0	PTAPUE5	PTAPUE4	PTAPUE3	PTAPUE2	PTAPUE1	PTAPUE0
$000C	PTBPUE	PTBPUE7	PTBPUE6	PTBPUE5	PTBPUE4	PTBPUE3	PTBPUE2	PTBPUE1	PTBPUE0
$000D–$0019	Unimplemented								
	Unimplemented								
$001A	KBSCR	0	0	0	0	KEYF	ACKK	IMASKK	MODEK
$001B	KBIER	0	AWUIE	KBIE5	KBIE4	KBIE3	KBIE2	KBIE1	KBIE0
$001C	Unimplemented								
$001D	INTSCR	0	0	0	0	IRQF1	ACK1	IMASK1	MODE1
$001E	CONFIG2	IRQPUD	IRQEN		OSCOPT1	OSCOPT0			RSTEN
$001F	CONFIG1	COPRS	LVISTOP	LVIRSTD	LVIPWRD	LVI5OR3	SSREC	STOP	COPD
$0020	TSC	TOF	TOIE	TSTOP	TRST	0	PS2	PS1	PS0
$0021	TCNTH	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
$0022	TCNTL	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
$0023	TMODH	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
$0024	TMODL	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
$0025	TSC0	CH0F	CH0IE	MS0B	MS0A	ELS0B	ELS0A	TOV0	CH0MAX
$0026	TCH0H	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
$0027	TCH0L	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
$0028	TSC1	CH1F	CH1IE	0	MS1A	ELS1B	ELS1A	TOV1	CH1MAX
$0029	TCH1H	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
$002A	TCH1L	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
$002B–$0035	Unimplemented								
	Unimplemented								
$0036	OSCSTAT							ECGON	ECGST
$0037	Unimplemented								
$0038	OSCTRIM	TRIM7	TRIM6	TRIM5	TRIM4	TRIM3	TRIM2	TRIM1	TRIM0
$0039–$003B	Unimplemented								
	Unimplemented								
$003C	ADSCR	COCO	AIEN	ADCO	CH4	CH3	CH2	CH1	CH0
$003D	Unimplemented								
$003E	ADR	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
$003F	ADICLK	ADIV2	ADIV1	ADIV0	0	0	0	0	0

= Unimplemented or Reserved

Figure 4. Control, Status, and Data Registers (Sheet 1 of 2)

Addr.	Register	Bit 7	6	5	4	3	2	1	Bit 0
$FE00	BSR							SBSW	
$FE01	SRSR	POR	PIN	COP	ILOP	ILAD	MODRST	LVI	0
$FE02	BRKAR	0	0	0	0	0	0	0	BDCOP
$FE03	BFCR	BCFE							
$FE04	INT1	0	IF5	IF4	IF3	0	IF1	0	0
$FE05	INT2	IF14	0	0	0	0	0	0	0
$FE06	INT3	0	0	0	0	0	0	0	IF15
$FE07	Reserved								
$FE08	FLCR	0	0	0	0	HVEN	MASS	ERASE	PGM
$FE09	BRKH	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
$FE0A	BRKL	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
$FE0B	BRKSCR	BRKE	BRKA	0	0	0	0	0	0
$FE0C	LVISR	LVIOUT	0	0	0	0	0	0	
$FE0D–$FE0F	Reserved for FLASH Test								
	Reserved for FLASH Test								
$FFBE	FLBPR	BPR7	BPR6	BPR5	BPR4	BPR3	BPR2	BPR1	BPR0
$FFBF	Reserved								
$FFC0	TRIMLOC	NON-VOLATILE TRIM ADJUSTMENT VALUE							
$FFC1	Reserved								
$FFFF	COPCTL	WRITE ANY VALUE TO RESET COP WATCHDOG							

= Unimplemented or Reserved

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

Table 3. Vector Addresses

Vector Priority	Vector	Address	Vector
Lowest	IF15	$FFDE	ADC conversion complete vector (high)
		$FFDF	ADC conversion complete vector (low)
	IF14	$FFE0	Keyboard vector (high)
		$FFE1	Keyboard vector (low)
	IF13 through IF6	—	Not used
	IF5	$FFF2	TIM overflow vector (high)
		$FFF3	TIM overflow vector (low)
	IF4	$FFF4	TIM Channel 1 vector (high)
		$FFF5	TIM Channel 1 vector (low)
	IF3	$FFF6	TIM Channel 0 vector (high)
		$FFF7	TIM Channel 0 vector (low)
	IF2	—	Not used
	IF1	$FFFA	$\overline{\text{IRQ}}$ vector (high)
		$FFFB	$\overline{\text{IRQ}}$ vector (low)
	—	$FFFC	SWI vector (high)
		$FFFD	SWI vector (low)
	—	$FFFE	Reset vector (high)
Highest		$FFFF	Reset vector (low)

FLASH Module

The FLASH memory consists of an array of 4096 or 1536 bytes with an additional 80 bytes for user vectors and miscellaneous. 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
- $FFB0 – $FFFF; user interrupt vectors etc., 80 bytes.

NOTE: *An erased bit reads as logic 1 and a programmed bit reads as logic 0. A security feature prevents unauthorized viewing of the FLASH contents.*

FLASH Control Register

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

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

Figure 5. FLASH Control Register (FLCR)

HVEN — High Voltage Enable Bit
1 = High voltage enabled to array and charge pump on

MASS — Mass Erase Control Bit
1 = Mass Erase operation selected

ERASE — Erase Control Bit
1 = Erase operation selected

PGM — Program Control Bit
1 = Program operation selected

FLASH Page Erase Operation

Use the following procedure to erase a page of FLASH memory. A page consists of 64 consecutive bytes starting from addresses $XX00, $XX40, $XX80, or $XXC0. The 80-byte user interrupt vectors area includes two pages ($FFB0–$FFBF and $FFC0–$FFFF). Any FLASH memory page can be erased alone.

1. Set the ERASE bit and clear the MASS bit in the FLASH control register.
2. Read the FLASH block protect register ($FFBE).

3. Write any data to any FLASH location within the address range of the block to be erased.
4. Wait for a time, t_{nvs} (minimum 10 μs).
5. Set the HVEN bit.
6. Wait for a time, t_{Erase} (minimum 1 ms or 4 ms).
7. Clear the ERASE and MASS bits.
8. Wait for a time, t_{nvh} (minimum 5 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. These operations must be performed in the order as shown, but other unrelated operations may occur between the steps.*

In applications that need up to 10,000 program/erase cycles, use the 4 ms page erase specification to get improved long-term reliability. Any application can use this 4 ms page erase specification. However, in applications where a FLASH location will be erased and reprogrammed less than 1000 times, and speed is important, use the 1 ms page erase specification to get a lower minimum erase time.

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.

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 from the FLASH block protect register ($FFBE).
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 6 and 7 until desired bytes within the row are programmed.
10. Clear the PGM bit(1).

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.

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: *Programming and erasing of FLASH locations cannot be performed by code being executed from the FLASH memory. These operations must be performed in the order shown, other unrelated operations may occur between the steps. Do not exceed t_{PROG} maximum.*

FLASH Block Protect Register

The FLASH block protect register is implemented as a byte within the FLASH memory, and therefore it is programmed using a FLASH memory byte-programming operation. The value in this register determines the starting address of the protected range within the FLASH memory. The FLASH is protected from this address to the end of FLASH memory at $FFFF.

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

Figure 6. FLASH Block Protect Register (FLBPR)

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

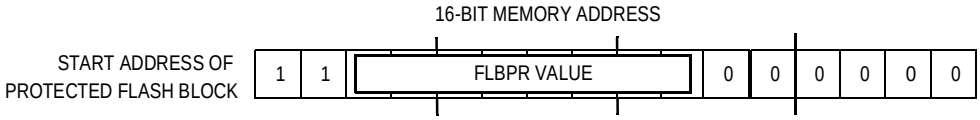


Figure 7. FLASH Block Protect Start Address

Table 4. 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, OSCTRIM, and vectors are protected
$FF	The entire FLASH memory is not protected.

Configuration Registers (CONFIG1, CONFIG2)

The configuration registers are used to initialize various options. The configuration registers can each be written once after each reset. Most of the configuration register bits are cleared during reset. Since the various options affect the operation of the microcontroller unit (MCU) it is recommended that these registers be written immediately after reset. The configuration registers are located at $001E and $001F, and may be read at anytime.

$001E	Bit 7	6	5	4	3	2	1	Bit 0
	IRQPUD	IRQEN	R	OSCOPT1	OSCOPT0	R	R	RSTEN
Reset:	0	0	0	0	0	0	0	U
POR:	0	0	0	0	0	0	0	0

R = Reserved U = Unaffected

Figure 8 Configuration Register 2 (CONFIG2)

IRQPUD — $\overline{IRQ}$ Pin Pullup Disable Control Bit
0 = Internal pullup is connected between $\overline{IRQ}$ pin and V_{DD} (if IRQEN = 1)

IRQEN — $\overline{IRQ}$ Pin Function Selection Bit
1 = PTA2/$\overline{IRQ}$/KBI2 pin configured for $\overline{IRQ}$ function
0 = Pin configured for PTA2 or KBI2 function

OSCOPT1:OSCOPT0 — Selection Bits for Oscillator Option
(0:0) Internal oscillator
(0:1) External oscillator
(1:0) External RC oscillator
(1:1) External XTAL oscillator

RSTEN — $\overline{RST}$ Pin Function Selection
1 = PTA2/$\overline{RST}$/KBI3 pin configured for $\overline{RESET}$ function
0 = Pin configured for PTA3 or KBI3 function

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

$001F	Bit 7	6	5	4	3	2	1	Bit 0
	COPRS	LVISTOP	LVIRSTD	LVIPWRD	LVI5OR3	SSREC	STOP	COPD
Reset:	0	0	0	0	U	0	0	0
POR:	0	0	0	0	0	0	0	0

U = Unaffected

Figure 9 Configuration Register 1 (CONFIG1)

COPRS (Out of STOP Mode) — COP Reset Period Selection Bit
1 = COP reset short cycle = ($2^{13} - 2^4$) x BUSCLKX4
0 = COP reset long cycle = ($2^{18} - 2^4$) x BUSCLKX4
To prevent a reset due to a COP watchdog timeout, write any value to COPCTL ($FFFF) before the COP timer reaches the selected timeout.

COPRS (In STOP Mode) — Auto Wake-up Period Selection Bit
1 = Auto wake-up short cycle = approximately 16 ms
0 = Auto wake-up long cycle = approximately 512 ms

LVISTOP — LVI Enable in Stop Mode Bit
1 = LVI enabled during stop mode
0 = LVI disabled during stop mode

LVIRSTD — LVI Reset Disable Bit
1 = LVI module resets disabled
0 = LVI module resets enabled

LVIPWRD — LVI Power Disable Bit
1 = LVI module power disabled

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

STOP — STOP Instruction Enable Bit
1 = STOP instruction enabled
0 = STOP instruction treated as illegal opcode

COPD — COP Disable Bit
1 = COP module disabled (does not force resets)

LVI Status Register

The LVI status register (LVISR) indicates if the V_{DD} voltage was detected below the V_{TRIPF} level while LVI resets have been disabled.

$FE0C	Bit 7	6	5	4	3	2	1	Bit 0
	LVIOUT	0	0	0	0	0	0	R
Reset:	0	0	0	0	0	0	0	0
	R = Reserved							

Figure 10. LVI Status Register (LVISR)

LVIOUT — LVI Output Bit

This read-only flag becomes set when the V_{DD} voltage falls below the V_{TRIPF} trip voltage and is cleared when V_{DD} voltage rises above V_{TRIPR}.

IRQ Status and Control Register

$001D	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	0	0	IRQF1	ACK1	IMASK1	MODE1
Reset:	0	0	0	0	0	0	0	0

Figure 11. IRQ Status and Control Register (INTSCR)

IRQF1 — IRQ Flag

This read-only status bit is high when the IRQ interrupt is pending.
1 = $\overline{\text{IRQ}}$ interrupt pending

ACK1 — IRQ Interrupt Request Acknowledge Bit

Writing a logic 1 to this write-only bit clears the IRQ latch. ACK1 always reads as logic 0.

IMASK1 — IRQ Interrupt Mask Bit
1 = IRQ interrupt requests disabled

MODE1 — IRQ Edge/Level Select Bit

This read/write bit controls the triggering sensitivity of the $\overline{\text{IRQ}}$ pin.
1 = $\overline{\text{IRQ}}$ interrupt requests on falling edges and low levels
0 = $\overline{\text{IRQ}}$ interrupt requests on falling edges only

SIM Reset Status Register

This register contains seven flags that show the source of the last reset. Clear the SIM reset status register by reading it. A power-on reset sets the POR bit and clears all other bits in the register.

$FE01	Bit 7	6	5	4	3	2	1	Bit 0
	POR	PIN	COP	ILOP	ILAD	MODRST	LVI	0
POR:	1	0	0	0	0	0	0	0

Figure 12. SIM Reset Status Register (SRSR)

POR — Power-On Reset Bit
1 = Last reset caused by POR circuit

PIN — External Reset Bit
1 = Last reset caused by external reset pin ($\overline{\text{RST}}$)

COP — Computer Operating Properly Reset Bit
1 = Last reset caused by COP timeout

ILOP — Illegal Opcode Reset Bit
1 = Last reset caused by an illegal opcode

ILAD — Illegal Address Reset Bit (opcode fetches only)
1 = Last reset caused by an opcode fetch from an illegal address

MODRST — Monitor Mode Entry Module Reset Bit
1 = Last reset caused by monitor mode entry when vector locations $FFFE and $FFFF are $FF after POR while PTA2/$\overline{IRQ}$ = V_{DD}

LVI — Low Voltage Inhibit Reset Bit
1 = Last reset caused by LVI circuit

Interrupt Status Registers (INT1, INT2, INT3)

These three registers include status flags which indicate which interrupt sources currently have pending requests. See **Table 3**.

$FE04	Bit 7	6	5	4	3	2	1	Bit 0
	0	IF5	IF4	IF3	0	IF1	0	0
Reset:	0	0	0	0	0	0	0	0
Source:		TOF	TCH1	TCH0		$\overline{IRQ}$		

Figure 13. Interrupt Status Register 1 (INT1)

$FE05	Bit 7	6	5	4	3	2	1	Bit 0
	IF14	0	0	0	0	0	0	0
Reset:	0	0	0	0	0	0	0	0
Source:	KBI							

Figure 14. Interrupt Status Register 2 (INT2)

$FE06	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	0	0	0	0	0	IF15
Reset:	0	0	0	0	0	0	0	0
Source:								ADC

Figure 15. Interrupt Status Register 3 (INT3)

IFxx — Interrupt Flags

These flags indicate the presence of interrupt requests from the sources shown below the corresponding IFxx bit.
1 = Interrupt request pending
0 = No interrupt request present

Central Processor Unit (CPU)

Figure 16 shows the five CPU registers. CPU registers are not part of the memory map.

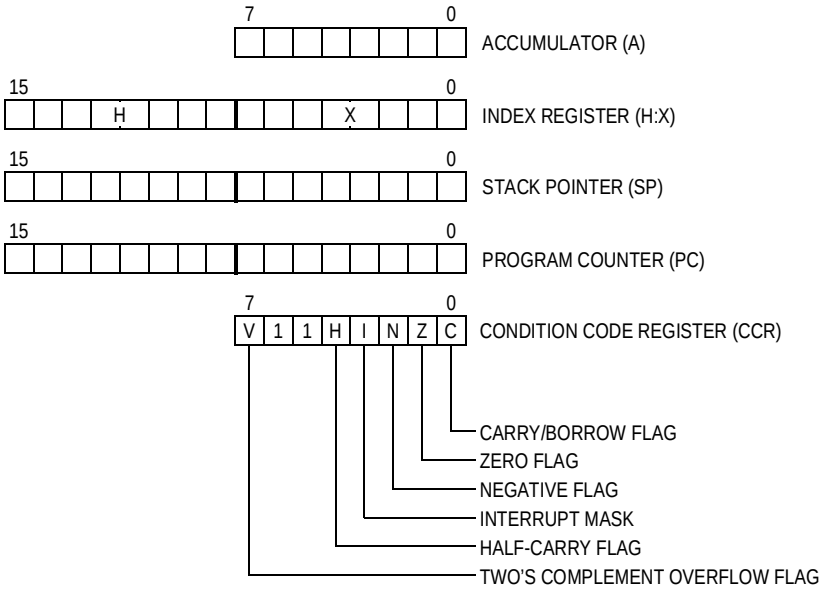


Figure 16. CPU Registers

Instruction Set Summary

Table 5 provides a summary of the M68HC08 instruction set.

Table 5. Instruction Set Summary (Sheet 1 of 7)

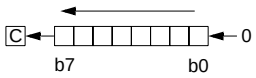
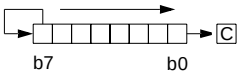
Source Form	Operation	Description	Effect on CCR						Address Mode	Opcode	Operand	Cycles
			V	H	I	N	Z	C				
ADC #*opr* ADC *opr* ADC *opr* ADC *opr*,X ADC *opr*,X ADC ,X ADC *opr*,SP ADC *opr*,SP	Add with Carry	A ← (A) + (M) + (C)	↕	↕	–	↕	↕	↕	IMM DIR EXT IX2 IX1 IX SP1 SP2	A9 B9 C9 D9 E9 F9 9EE9 9ED9	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5
ADD #*opr* ADD *opr* ADD *opr* ADD *opr*,X ADD *opr*,X ADD ,X ADD *opr*,SP ADD *opr*,SP	Add without Carry	A ← (A) + (M)	↕	↕	–	↕	↕	↕	IMM DIR EXT IX2 IX1 IX SP1 SP2	AB BB CB DB EB FB 9EEB 9EDB	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5
AIS #*opr*	Add Immediate Value (Signed) to SP	SP ← (SP) + (16 « M)	–	–	–	–	–	–	IMM	A7	ii	2
AIX #*opr*	Add Immediate Value (Signed) to H:X	H:X ← (H:X) + (16 « M)	–	–	–	–	–	–	IMM	AF	ii	2
AND #*opr* AND *opr* AND *opr* AND *opr*,X AND *opr*,X AND ,X AND *opr*,SP AND *opr*,SP	Logical AND	A ← (A) & (M)	0	–	–	↕	↕	–	IMM DIR EXT IX2 IX1 IX SP1 SP2	A4 B4 C4 D4 E4 F4 9EE4 9ED4	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5
ASL *opr* ASLA ASLX ASL *opr*,X ASL ,X ASL *opr*,SP	Arithmetic Shift Left (Same as LSL)		↕	–	–	↕	↕	↕	DIR INH INH IX1 IX SP1	38 48 58 68 78 9E68	dd ff ff	4 1 1 4 3 5
ASR *opr* ASRA ASRX ASR *opr*,X ASR *opr*,X ASR *opr*,SP	Arithmetic Shift Right		↕	–	–	↕	↕	↕	DIR INH INH IX1 IX SP1	37 47 57 67 77 9E67	dd ff ff	4 1 1 4 3 5
BCC *rel*	Branch if Carry Bit Clear	PC ← (PC) + 2 + rel ? (C) = 0	–	–	–	–	–	–	REL	24	rr	3
BCLR *n*, *opr*	Clear Bit n in M	Mn ← 0	–	–	–	–	–	–	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	11 13 15 17 19 1B 1D 1F	dd dd dd dd dd dd dd dd	4 4 4 4 4 4 4 4
BCS *rel*	Branch if Carry Bit Set (Same as BLO)	PC ← (PC) + 2 + *rel* ? (C) = 1	–	–	–	–	–	–	REL	25	rr	3

Table 5. Instruction Set Summary (Sheet 2 of 7)

Source Form	Operation	Description	Effect on CCR						Address Mode	Opcode	Operand	Cycles
			V	H	I	N	Z	C				
BEQ *rel*	Branch if Equal	PC ← (PC) + 2 + *rel* ? (Z) = 1	–	–	–	–	–	–	REL	27	rr	3
BGE *opr*	Branch if Greater Than or Equal To (Signed Operands)	PC ← (PC) + 2 + *rel* ? (N ⊕ V) = 0	–	–	–	–	–	–	REL	90	rr	3
BGT *opr*	Branch if Greater Than (Signed Operands)	PC ← (PC) + 2 + *rel* ? (Z) | (N ⊕ V) = 0	–	–	–	–	–	–	REL	92	rr	3
BHCC *rel*	Branch if Half Carry Bit Clear	PC ← (PC) + 2 + *rel* ? (H) = 0	–	–	–	–	–	–	REL	28	rr	3
BHCS *rel*	Branch if Half Carry Bit Set	PC ← (PC) + 2 + *rel* ? (H) = 1	–	–	–	–	–	–	REL	29	rr	3
BHI *rel*	Branch if Higher	PC ← (PC) + 2 + *rel* ? (C) | (Z) = 0	–	–	–	–	–	–	REL	22	rr	3
BHS *rel*	Branch if Higher or Same (Same as BCC)	PC ← (PC) + 2 + *rel* ? (C) = 0	–	–	–	–	–	–	REL	24	rr	3
BIH *rel*	Branch if $\overline{\text{IRQ}}$ Pin High	PC ← (PC) + 2 + *rel* ? $\overline{\text{IRQ}}$ = 1	–	–	–	–	–	–	REL	2F	rr	3
BIL *rel*	Branch if $\overline{\text{IRQ}}$ Pin Low	PC ← (PC) + 2 + *rel* ? $\overline{\text{IRQ}}$ = 0	–	–	–	–	–	–	REL	2E	rr	3
BIT #*opr* BIT *opr* BIT *opr* BIT *opr*,X BIT *opr*,X BIT ,X BIT *opr*,SP BIT *opr*,SP	Bit Test	(A) & (M)	0	–	–	↕	↕	–	IMM DIR EXT IX2 IX1 IX SP1 SP2	A5 B5 C5 D5 E5 F5 9EE5 9ED5	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5
BLE *opr*	Branch if Less Than or Equal To (Signed Operands)	PC ← (PC) + 2 + *rel* ? (Z) | (N ⊕ V) = 1	–	–	–	–	–	–	REL	93	rr	3
BLO *rel*	Branch if Lower (Same as BCS)	PC ← (PC) + 2 + *rel* ? (C) = 1	–	–	–	–	–	–	REL	25	rr	3
BLS *rel*	Branch if Lower or Same	PC ← (PC) + 2 + *rel* ? (C) | (Z) = 1	–	–	–	–	–	–	REL	23	rr	3
BLT *opr*	Branch if Less Than (Signed Operands)	PC ← (PC) + 2 + *rel* ? (N ⊕ V) =1	–	–	–	–	–	–	REL	91	rr	3
BMC *rel*	Branch if Interrupt Mask Clear	PC ← (PC) + 2 + *rel* ? (I) = 0	–	–	–	–	–	–	REL	2C	rr	3
BMI *rel*	Branch if Minus	PC ← (PC) + 2 + *rel* ? (N) = 1	–	–	–	–	–	–	REL	2B	rr	3
BMS *rel*	Branch if Interrupt Mask Set	PC ← (PC) + 2 + *rel* ? (I) = 1	–	–	–	–	–	–	REL	2D	rr	3
BNE *rel*	Branch if Not Equal	PC ← (PC) + 2 + *rel* ? (Z) = 0	–	–	–	–	–	–	REL	26	rr	3
BPL *rel*	Branch if Plus	PC ← (PC) + 2 + *rel* ? (N) = 0	–	–	–	–	–	–	REL	2A	rr	3
BRA *rel*	Branch Always	PC ← (PC) + 2 + *rel*	–	–	–	–	–	–	REL	20	rr	3
BRCLR *n*,*opr*,*rel*	Branch if Bit *n* in M Clear	PC ← (PC) + 3 + *rel* ? (Mn) = 0	–	–	–	–	–	↕	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	01 03 05 07 09 0B 0D 0F	dd rr dd rr dd rr dd rr dd rr dd rr dd rr dd rr	5 5 5 5 5 5 5 5
BRN *rel*	Branch Never	PC ← (PC) + 2	–	–	–	–	–	–	REL	21	rr	3

Table 5. Instruction Set Summary (Sheet 3 of 7)

Source Form	Operation	Description	Effect on CCR						Address Mode	Opcode	Operand	Cycles
			V	H	I	N	Z	C				
BRSET *n*,*opr*,*rel*	Branch if Bit *n* in M Set	PC ← (PC) + 3 + *rel* ? (Mn) = 1	–	–	–	–	–	↕	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	00 02 04 06 08 0A 0C 0E	dd rr dd rr dd rr dd rr dd rr dd rr dd rr dd rr	5 5 5 5 5 5 5 5
BSET *n*,*opr*	Set Bit *n* in M	Mn ← 1	–	–	–	–	–	–	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	10 12 14 16 18 1A 1C 1E	dd dd dd dd dd dd dd dd	4 4 4 4 4 4 4 4
BSR *rel*	Branch to Subroutine	PC ← (PC) + 2; push (PCL) SP ← (SP) – 1; push (PCH) SP ← (SP) – 1 PC ← (PC) + *rel*	–	–	–	–	–	–	REL	AD	rr	4
CBEQ *opr*,*rel* CBEQA #*opr*,*rel* CBEQX #*opr*,*rel* CBEQ *opr*,X+,*rel* CBEQ X+,*rel* CBEQ *opr*,SP,*rel*	Compare and Branch if Equal	PC ← (PC) + 3 + rel ? (A) – (M) = $00 PC ← (PC) + 3 + rel ? (A) – (M) = $00 PC ← (PC) + 3 + rel ? (X) – (M) = $00 PC ← (PC) + 3 + rel ? (A) – (M) = $00 PC ← (PC) + 2 + rel ? (A) – (M) = $00 PC ← (PC) + 4 + rel ? (A) – (M) = $00	–	–	–	–	–	–	DIR IMM IMM IX1+ IX+ SP1	31 41 51 61 71 9E61	dd rr ii rr ii rr ff rr rr ff rr	5 4 4 5 4 6
CLC	Clear Carry Bit	C ← 0	–	–	–	–	–	0	INH	98		1
CLI	Clear Interrupt Mask	I ← 0	–	–	0	–	–	–	INH	9A		2
CLR *opr* CLRA CLRX CLRH CLR *opr*,X CLR ,X CLR *opr*,SP	Clear	M ← $00 A ← $00 X ← $00 H ← $00 M ← $00 M ← $00 M ← $00	0	–	–	0	1	–	DIR INH INH INH IX1 IX SP1	3F 4F 5F 8C 6F 7F 9E6F	dd ff ff	3 1 1 1 3 2 4
CMP #*opr* CMP *opr* CMP *opr* CMP *opr*,X CMP *opr*,X CMP ,X CMP *opr*,SP CMP *opr*,SP	Compare A with M	(A) – (M)	↕	–	–	↕	↕	↕	IMM DIR EXT IX2 IX1 IX SP1 SP2	A1 B1 C1 D1 E1 F1 9EE1 9ED1	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5
COM *opr* COMA COMX COM *opr*,X COM ,X COM *opr*,SP	Complement (One's Complement)	M ← ($\overline{M}$) = $FF – (M) A ← ($\overline{A}$) = $FF – (M) X ← ($\overline{X}$) = $FF – (M) M ← ($\overline{M}$) = $FF – (M) M ← ($\overline{M}$) = $FF – (M) M ← (M) = $FF – (M)	0	–	–	↕	↕	1	DIR INH INH IX1 IX SP1	33 43 53 63 73 9E63	dd ff ff	4 1 1 4 3 5
CPHX #*opr* CPHX *opr*	Compare H:X with M	(H:X) – (M:M + 1)	↕	–	–	↕	↕	↕	IMM DIR	65 75	ii ii+1 dd	3 4

Table 5. Instruction Set Summary (Sheet 4 of 7)

Source Form	Operation	Description	Effect on CCR						Address Mode	Opcode	Operand	Cycles
			V	H	I	N	Z	C				
CPX #*opr* CPX *opr* CPX *opr* CPX ,X CPX *opr*,X CPX *opr*,X CPX *opr*,SP CPX *opr*,SP	Compare X with M	(X) – (M)	↕	–	–	↕	↕	↕	IMM DIR EXT IX2 IX1 IX SP1 SP2	A3 B3 C3 D3 E3 F3 9EE3 9ED3	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5
DAA	Decimal Adjust A	$(A)_{10}$	U	–	–	↕	↕	↕	INH	72		2
DBNZ *opr,rel* DBNZA *rel* DBNZX *rel* DBNZ *opr*,X,*rel* DBNZ X,*rel* DBNZ *opr*,SP,*rel*	Decrement and Branch if Not Zero	A ← (A) – 1 or M ← (M) – 1 or X ← (X) – 1 PC ← (PC) + 3 + *rel* ? (result) ≠ 0 PC ← (PC) + 2 + *rel* ? (result) ≠ 0 PC ← (PC) + 2 + *rel* ? (result) ≠ 0 PC ← (PC) + 3 + *rel* ? (result) ≠ 0 PC ← (PC) + 2 + *rel* ? (result) ≠ 0 PC ← (PC) + 4 + *rel* ? (result) ≠ 0	–	–	–	–	–	–	DIR INH INH IX1 IX SP1	3B 4B 5B 6B 7B 9E6B	dd rr rr rr ff rr rr ff rr	5 3 3 5 4 6
DEC *opr* DECA DECX DEC *opr*,X DEC ,X DEC *opr*,SP	Decrement	M ← (M) – 1 A ← (A) – 1 X ← (X) – 1 M ← (M) – 1 M ← (M) – 1 M ← (M) – 1	↕	–	–	↕	↕	–	DIR INH INH IX1 IX SP1	3A 4A 5A 6A 7A 9E6A	dd ff ff	4 1 1 4 3 5
DIV	Divide	A ← (H:A)/(X) H ← Remainder	–	–	–	–	↕	↕	INH	52		7
EOR #*opr* EOR *opr* EOR *opr* EOR *opr*,X EOR *opr*,X EOR ,X EOR *opr*,SP EOR *opr*,SP	Exclusive OR M with A	A ← (A ⊕ M)	0	–	–	↕	↕	–	IMM DIR EXT IX2 IX1 IX SP1 SP2	A8 B8 C8 D8 E8 F8 9EE8 9ED8	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5
INC *opr* INCA INCX INC *opr*,X INC ,X INC *opr*,SP	Increment	M ← (M) + 1 A ← (A) + 1 X ← (X) + 1 M ← (M) + 1 M ← (M) + 1 M ← (M) + 1	↕	–	–	↕	↕	–	DIR INH INH IX1 IX SP1	3C 4C 5C 6C 7C 9E6C	dd ff ff	4 1 1 4 3 5
JMP *opr* JMP *opr* JMP *opr*,X JMP *opr*,X JMP ,X	Jump	PC ← Jump Address	–	–	–	–	–	–	DIR EXT IX2 IX1 IX	BC CC DC EC FC	dd hh ll ee ff ff	2 3 4 3 2
JSR *opr* JSR *opr* JSR *opr*,X JSR *opr*,X JSR ,X	Jump to Subroutine	PC ← (PC) + *n* (*n* = 1, 2, or 3) Push (PCL); SP ← (SP) – 1 Push (PCH); SP ← (SP) – 1 PC ← Unconditional Address	–	–	–	–	–	–	DIR EXT IX2 IX1 IX	BD CD DD ED FD	dd hh ll ee ff ff	4 5 6 5 4
LDA #*opr* LDA *opr* LDA *opr* LDA *opr*,X LDA *opr*,X LDA ,X LDA *opr*,SP LDA *opr*,SP	Load A from M	A ← (M)	0	–	–	↕	↕	–	IMM DIR EXT IX2 IX1 IX SP1 SP2	A6 B6 C6 D6 E6 F6 9EE6 9ED6	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5

Table 5. Instruction Set Summary (Sheet 5 of 7)

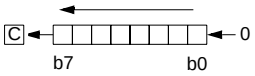
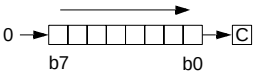
Source Form	Operation	Description	Effect on CCR						Address Mode	Opcode	Operand	Cycles
			V	H	I	N	Z	C				
LDHX #*opr* LDHX *opr*	Load H:X from M	H:X ← (M:M + 1)	0	–	–	↕	↕	–	IMM DIR	45 55	ii jj dd	3 4
LDX #*opr* LDX *opr* LDX *opr* LDX *opr*,X LDX *opr*,X LDX ,X LDX *opr*,SP LDX *opr*,SP	Load X from M	X ← (M)	0	–	–	↕	↕	–	IMM DIR EXT IX2 IX1 IX SP1 SP2	AE BE CE DE EE FE 9EEE 9EDE	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5
LSL *opr* LSLA LSLX LSL *opr*,X LSL ,X LSL *opr*,SP	Logical Shift Left (Same as ASL)		↕	–	–	↕	↕	↕	DIR INH INH IX1 IX SP1	38 48 58 68 78 9E68	dd ff ff	4 1 1 4 3 5
LSR *opr* LSRA LSR*X* LSR *opr*,X LSR ,X LSR *opr*,SP	Logical Shift Right		↕	–	–	0	↕	↕	DIR INH INH IX1 IX SP1	34 44 54 64 74 9E64	dd ff ff	4 1 1 4 3 5
MOV *opr*,*opr* MOV *opr*,X+ MOV #*opr*,*opr* MOV X+,*opr*	Move	$(M)_{Destination} \leftarrow (M)_{Source}$ H:X ← (H:X) + 1 (IX+D, DIX+)	0	–	–	↕	↕	–	DD DIX+ IMD IX+D	4E 5E 6E 7E	dd dd dd ii dd dd	5 4 4 4
MUL	Unsigned multiply	X:A ← (X) × (A)	–	0	–	–	–	0	INH	42		5
NEG *opr* NEGA NEGX NEG *opr*,X NEG ,X NEG *opr*,SP	Negate (Two's Complement)	M ← –(M) = $00 – (M) A ← –(A) = $00 – (A) X ← –(X) = $00 – (X) M ← –(M) = $00 – (M) M ← –(M) = $00 – (M)	↕	–	–	↕	↕	↕	DIR INH INH IX1 IX SP1	30 40 50 60 70 9E60	dd ff ff	4 1 1 4 3 5
NOP	No Operation	None	–	–	–	–	–	–	INH	9D		1
NSA	Nibble Swap A	A ← (A[3:0]:A[7:4])	–	–	–	–	–	–	INH	62		3
ORA #*opr* ORA *opr* ORA *opr* ORA *opr*,X ORA *opr*,X ORA ,X ORA *opr*,SP ORA *opr*,SP	Inclusive OR A and M	A ← (A) | (M)	0	–	–	↕	↕	–	IMM DIR EXT IX2 IX1 IX SP1 SP2	AA BA CA DA EA FA 9EEA 9EDA	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5
PSHA	Push A onto Stack	Push (A); SP ← (SP) – 1	–	–	–	–	–	–	INH	87		2
PSHH	Push H onto Stack	Push (H); SP ← (SP) – 1	–	–	–	–	–	–	INH	8B		2
PSHX	Push X onto Stack	Push (X); SP ← (SP) – 1	–	–	–	–	–	–	INH	89		2
PULA	Pull A from Stack	SP ← (SP + 1); Pull (A)	–	–	–	–	–	–	INH	86		2
PULH	Pull H from Stack	SP ← (SP + 1); Pull (H)	–	–	–	–	–	–	INH	8A		2
PULX	Pull X from Stack	SP ← (SP + 1); Pull (X)	–	–	–	–	–	–	INH	88		2

Table 5. Instruction Set Summary (Sheet 6 of 7)

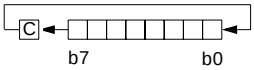
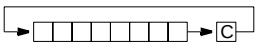
Source Form	Operation	Description	Effect on CCR						Address Mode	Opcode	Operand	Cycles
			V	H	I	N	Z	C				
ROL *opr* ROLA ROLX ROL *opr*,X ROL ,X ROL *opr*,SP	Rotate Left through Carry		↕	–	–	↕	↕	↕	DIR INH INH IX1 IX SP1	39 49 59 69 79 9E69	dd ff ff	4 1 1 4 3 5
ROR *opr* RORA RORX ROR *opr*,X ROR ,X ROR *opr*,SP	Rotate Right through Carry		↕	–	–	↕	↕	↕	DIR INH INH IX1 IX SP1	36 46 56 66 76 9E66	dd ff ff	4 1 1 4 3 5
RSP	Reset Stack Pointer	SP ← $FF	–	–	–	–	–	–	INH	9C		1
RTI	Return from Interrupt	SP ← (SP) + 1; Pull (CCR) SP ← (SP) + 1; Pull (A) SP ← (SP) + 1; Pull (X) SP ← (SP) + 1; Pull (PCH) SP ← (SP) + 1; Pull (PCL)	↕	↕	↕	↕	↕	↕	INH	80		7
RTS	Return from Subroutine	SP ← SP + 1; Pull (PCH) SP ← SP + 1; Pull (PCL)	–	–	–	–	–	–	INH	81		4
SBC #*opr* SBC *opr* SBC *opr* SBC *opr*,X SBC *opr*,X SBC ,X SBC *opr*,SP SBC *opr*,SP	Subtract with Carry	A ← (A) – (M) – (C)	↕	–	–	↕	↕	↕	IMM DIR EXT IX2 IX1 IX SP1 SP2	A2 B2 C2 D2 E2 F2 9EE2 9ED2	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5
SEC	Set Carry Bit	C ← 1	–	–	–	–	–	1	INH	99		1
SEI	Set Interrupt Mask	I ← 1	–	–	1	–	–	–	INH	9B		2
STA *opr* STA *opr* STA *opr*,X STA *opr*,X STA ,X STA *opr*,SP STA *opr*,SP	Store A in M	M ← (A)	0	–	–	↕	↕	–	DIR EXT IX2 IX1 IX SP1 SP2	B7 C7 D7 E7 F7 9EE7 9ED7	dd hh ll ee ff ff ff ee ff	3 4 4 3 2 4 5
STHX *opr*	Store H:X in M	(M:M + 1) ← (H:X)	0	–	–	↕	↕	–	DIR	35	dd	4
STOP	Enable $\overline{\text{IRQ}}$ Pin; Stop Oscillator	I ← 0; Stop Oscillator	–	–	0	–	–	–	INH	8E		1
STX *opr* STX *opr* STX *opr*,X STX *opr*,X STX ,X STX *opr*,SP STX *opr*,SP	Store X in M	M ← (X)	0	–	–	↕	↕	–	DIR EXT IX2 IX1 IX SP1 SP2	BF CF DF EF FF 9EEF 9EDF	dd hh ll ee ff ff ff ee ff	3 4 4 3 2 4 5
SUB #*opr* SUB *opr* SUB *opr* SUB *opr*,X SUB *opr*,X SUB ,X SUB *opr*,SP SUB *opr*,SP	Subtract	A ← (A) – (M)	↕	–	–	↕	↕	↕	IMM DIR EXT IX2 IX1 IX SP1 SP2	A0 B0 C0 D0 E0 F0 9EE0 9ED0	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5

Table 5. Instruction Set Summary (Sheet 7 of 7)

Source Form	Operation	Description	Effect on CCR						Address Mode	Opcode	Operand	Cycles
			V	H	I	N	Z	C				
SWI	Software Interrupt	PC ← (PC) + 1; Push (PCL) SP ← (SP) – 1; Push (PCH) SP ← (SP) – 1; Push (X) SP ← (SP) – 1; Push (A) SP ← (SP) – 1; Push (CCR) SP ← (SP) – 1; I ← 1 PCH ← Interrupt Vector High Byte PCL ← Interrupt Vector Low Byte	–	–	1	–	–	–	INH	83		9
TAP	Transfer A to CCR	CCR ← (A)	↕	↕	↕	↕	↕	↕	INH	84		2
TAX	Transfer A to X	X ← (A)	–	–	–	–	–	–	INH	97		1
TPA	Transfer CCR to A	A ← (CCR)	–	–	–	–	–	–	INH	85		1
TST *opr* TSTA TSTX TST *opr*,X TST ,X TST *opr*,SP	Test for Negative or Zero	(A) – $00 or (X) – $00 or (M) – $00	0	–	–	↕	↕	–	DIR INH INH IX1 IX SP1	3D 4D 5D 6D 7D 9E6D	dd ff ff	3 1 1 3 2 4
TSX	Transfer SP to H:X	H:X ← (SP) + 1	–	–	–	–	–	–	INH	95		2
TXA	Transfer X to A	A ← (X)	–	–	–	–	–	–	INH	9F		1
TXS	Transfer H:X to SP	(SP) ← (H:X) – 1	–	–	–	–	–	–	INH	94		2

A	Accumulator
C	Carry/borrow bit
CCR	Condition code register
dd	Direct address of operand
dd rr	Direct address of operand and relative offset of branch instruction
DD	Direct to direct addressing mode
DIR	Direct addressing mode
DIX+	Direct to indexed with post increment addressing mode
ee ff	High and low bytes of offset in indexed, 16-bit offset addressing
EXT	Extended addressing mode
ff	Offset byte in indexed, 8-bit offset addressing
H	Half-carry bit
H	Index register high byte
hh ll	High and low bytes of operand address in extended addressing
I	Interrupt mask
ii	Immediate operand byte
IMD	Immediate source to direct destination addressing mode
IMM	Immediate addressing mode
INH	Inherent addressing mode
IX	Indexed, no offset addressing mode
IX+	Indexed, no offset, post increment addressing mode
IX+D	Indexed with post increment to direct addressing mode
IX1	Indexed, 8-bit offset addressing mode
IX1+	Indexed, 8-bit offset, post increment addressing mode
IX2	Indexed, 16-bit offset addressing mode
M	Memory location
N	Negative bit
n	Any bit
opr	Operand (one or two bytes)
PC	Program counter
PCH	Program counter high byte
PCL	Program counter low byte
REL	Relative addressing mode
rel	Relative program counter offset byte
rr	Relative program counter offset byte
SP1	Stack pointer, 8-bit offset addressing mode
SP2	Stack pointer 16-bit offset addressing mode
SP	Stack pointer
U	Undefined
V	Overflow bit
X	Index register low byte
Z	Zero bit
&	Logical AND
|	Logical OR
⊕	Logical EXCLUSIVE OR
( )	Contents of
–( )	Negation (two's complement)
#	Immediate value
«	Sign extend
←	Loaded with
?	If
:	Concatenated with
↕	Set or cleared
—	Not affected

Oscillator Module (OSC)

The oscillator has these four clock source options available:

1. Internal oscillator: An internally generated, fixed frequency clock, trimmable to ± 5% in steps of approximately 0.2%. This is the default option out of reset.
2. External oscillator: An external clock that can be driven directly into OSC1.
3. External RC: A built-in oscillator module (RC oscillator) that requires an external R connection only on one pin. The capacitor will be internal to the chip.
4. External crystal: A built-in oscillator module (XTAL oscillator) that requires an external crystal or ceramic-resonator on two pins.

Internal to External Clock Switching

When external clock source (external OSC, RC, or XTAL) is desired, the user must perform the following steps:

1. For External crystal circuits only, OSCOPT[1:0] = 1:1: To help precharge an external crystal oscillator, set PTA4 (OSC2) as an output and drive high for several cycles. Before writing OSCOPT[1:0], the crystal will see a sharp falling edge at startup.
2. Set CONFIG2 bits OSCOPT[1:0] according to **Table 7**. The oscillator module control logic will then set OSC1 as an external clock input and, if the external crystal option is selected, OSC2 will also be set as the clock output.
3. Create a software delay to wait the stabilization time needed for the selected clock source (crystal, resonator, RC) as recommended by the component manufacturer. A good rule of thumb for crystal oscillators is to wait 4096 cycles of the crystal frequency, i.e., for a 4-MHz crystal, wait approximately 1 msec.
4. After this delay has elapsed, the ECGON bit in the OSC status register (OSCSTAT) should be set by the user software.
5. After ECGON set is detected, the OSC module checks for oscillator activity by waiting two external clock rising edges.
6. The OSC module than switches to the external clock. Logic provides a glitch free transition.
7. The OSC module sets the ECGST bit in the OSCSTAT register and then stops the internal oscillator.

NOTE: *Once transition to the external clock is done, the internal oscillator will only be reactivated with reset. Clock does not switch back to internal if external clock stops.*

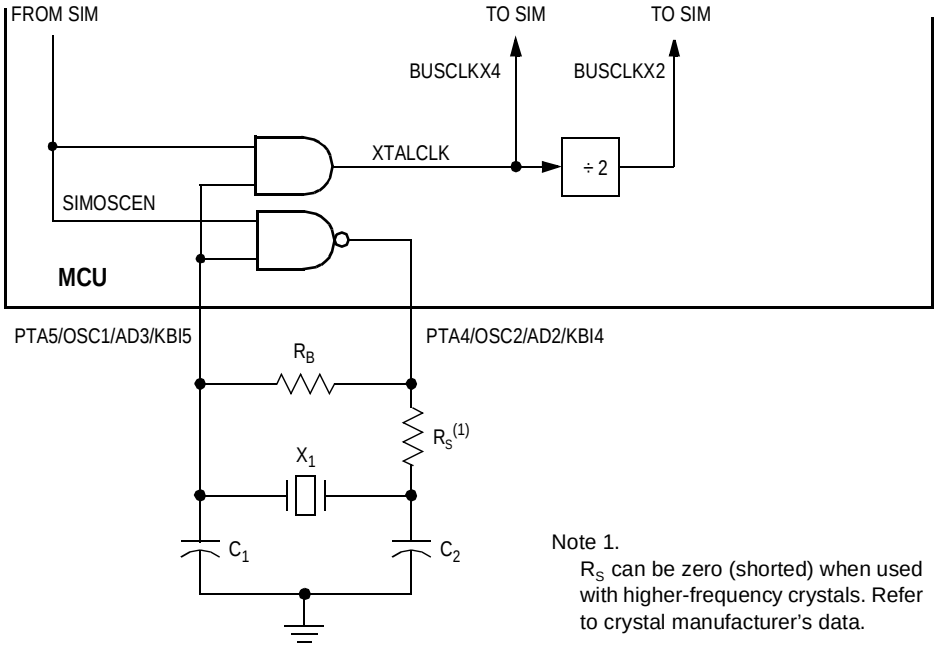


Figure 17. XTAL Oscillator External Connections

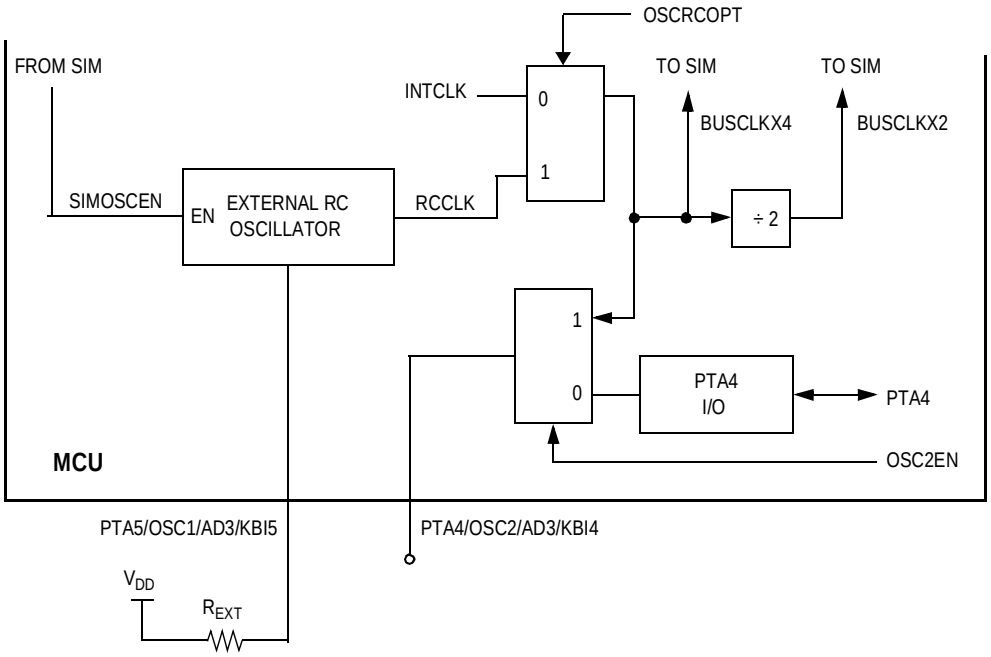


Figure 18. RC Oscillator External Connections

Oscillator Status Register

The oscillator status register (OSCSTAT) contains the bits for switching from internal to external clock sources

$0036	Bit 7	6	5	4	3	2	1	Bit 0
	R	R	R	R	R	R	ECGON	ECGST
Reset:	0	0	0	0	0	0	0	0

R = Reserved

Figure 19. Oscillator Status Register (OSCSTAT)

ECGON — External Clock Generator On Bit
1 = External clock generator enabled

ECGST — External Clock Status Bit
1 = An external clock source engaged

Oscillator Trim Register (OSCTRIM)

$0038	Bit 7	6	5	4	3	2	1	Bit 0
	TRIM7	TRIM6	TRIM5	TRIM4	TRIM3	TRIM2	TRIM71	TRIM0
Reset:	1	0	0	0	0	0	0	0

Figure 20. Oscillator Trim Register (OSCTRIM)

TRIM7–TRIM0 — Internal Oscillator Trim Factor Bits

These read/write bits change the size of the internal capacitor used by the internal oscillator. By testing the frequency of the internal clock and adjusting this factor accordingly, the frequency of the internal clock can be fine tuned. Increasing (decreasing) this factor by one increases (decreases) the period by approximately 0.2% of the untrimmed period (the period for trim = $80). The trimmed frequency is guaranteed not to vary by more than ±5% over the full specified range of temperature and voltage. The reset value is $80 which sets the frequency to 3.2 MHz ± 25% (bus rate).

A trim adjustment factor can be programmed into FLASH memory at TRIMLOC ($FFC0). During the application initialization routine, this value can be read from TRIMLOC and be stored to OSCTRIM ($0038) to fine tune the internal oscillator frequency.

Timer Interface Module (TIM)

Features of the TIM include the following:

- Two input capture/output compare channels
  - Rising-edge, falling-edge, or any-edge input capture trigger
  - Set, clear, or toggle output compare action
- Buffered and unbuffered pulse width modulation (PWM) signal generation
- Programmable TIM clock input with 7-frequency internal bus clock prescaler selection
- Free-running or modulo up-count operation
- Optional toggle of any channel pin on overflow
- TIM counter stop and reset bits

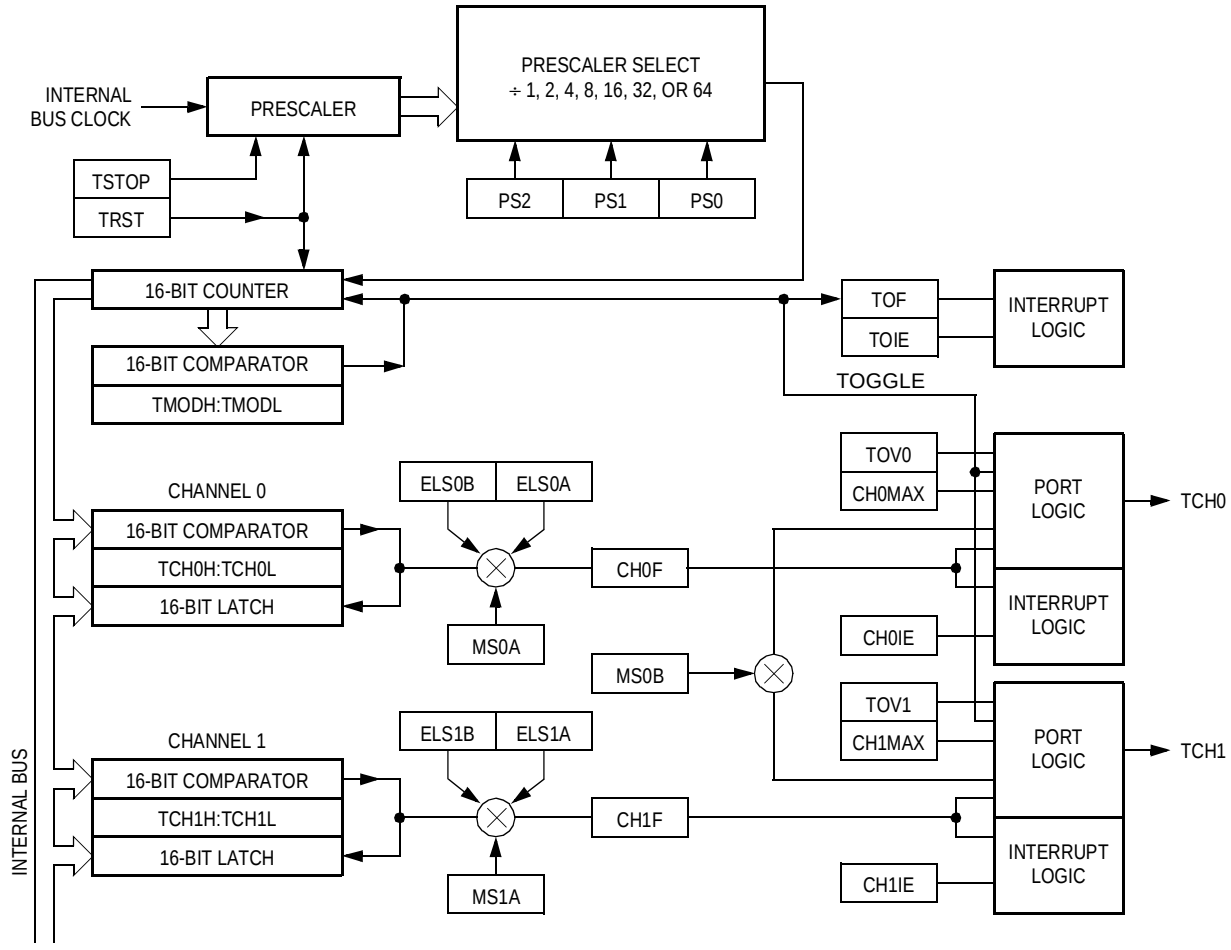


Figure 21. TIM Block Diagram

PWM Initialization

Recommended initialization procedure for unbuffered or buffered PWM signals.

1. In TSC:
   a. Stop the TIM counter by setting TSTOP.
   b. Reset the TIM counter and prescaler by setting TRST.
2. Write TMODH:TMODL to set the required PWM period.
3. Write TCHxH:TCHxL to set the required pulse width.
4. Write TIM channel x status and control register (TSCx) to select the desired function:
   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 7**.
   b. Write 1 to the toggle-on-overflow bit, TOVx.
   c. Write 1:0 (to clear output on compare) or 1:1 (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 7**.
5. Clear TSTOP in the TIM status control register (TSC).

TIM Status and Control Register

$0020	Bit 7	6	5	4	3	2	1	Bit 0
	TOF	TOIE	TSTOP	TRST	0	PS2	PS1	PS0
Reset:	0	0	1	0	0	0	0	0

Figure 22. TIM Status and Control Register (TSC)

TOF — TIM Overflow Flag Bit

TOF is set when the TIM counter reaches the modulo value programmed in the TIM counter modulo registers. Clear TOF by reading the TIM status and control register when TOF is set and then writing a logic 0 to TOF.

1 = TIM counter has reached modulo value

TOIE — TIM Overflow Interrupt Enable Bit

1 = TIM overflow interrupts enabled

TSTOP — TIM Stop Bit

1 = TIM counter stopped

TRST — TIM Reset Bit

Setting this write-only bit resets the TIM counter and the TIM prescaler. TRST is cleared automatically after the TIM counter is reset and always reads as logic 0.

1 = Prescaler and TIM counter cleared

NOTE: *Setting the TSTOP and TRST bits simultaneously stops the TIM counter at a value of $0000.*

PS[2:0] — Prescaler Select Bits

Table 6. Prescaler Selection

PS2	PS1	PS0	TIM Clock Source
0	0	0	Internal bus clock ÷ 1
0	0	1	Internal bus clock ÷ 2
0	1	0	Internal bus clock ÷ 4
0	1	1	Internal bus clock ÷ 8
1	0	0	Internal bus clock ÷ 16
1	0	1	Internal bus clock ÷ 32
1	1	0	Internal bus clock ÷ 64
1	1	1	Reserved

TIM Counter Registers

The two read-only TIM counter registers contain the high and low bytes of the value in the TIM counter. Reading the high byte (TCNTH) latches the contents of the low byte (TCNTL) into a buffer. Subsequent reads of TCNTH do not affect the latched TCNTL value until TCNTL is read.

TCNTH $0021	Bit 7	6	5	4	3	2	1	Bit 0
	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
TCNTL $0022	Bit 7	6	5	4	3	2	1	Bit 0
	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

Figure 23. TIM Counter Registers (TCNTH:TCNTL)

TIM Counter Modulo Registers

When the TIM counter reaches the modulo value, the overflow flag (TOF) becomes set, and the TIM counter resumes counting from $0000 at the next timer clock. Writing to the high byte (TMODH) inhibits the TOF bit and overflow interrupts until the low byte (TMODL) is written.

TMODH $0023	Bit 7	6	5	4	3	2	1	Bit 0
	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
Reset:	1	1	1	1	1	1	1	1
TMODL $0024	Bit 7	6	5	4	3	2	1	Bit 0
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Reset:	1	1	1	1	1	1	1	1

Figure 24. TIM Counter Modulo Registers (TMODH:TMODL)

TIM Channel Status and Control Registers

		Bit 7	6	5	4	3	2	1	Bit 0
TSC0	$0025	CH0F	CH0IE	MS0B	MS0A	ELS0B	ELS0A	TOV0	CH0MAX
	Reset:	0	0	0	0	0	0	0	0
TSC1	$0028	Bit 7	6	5	4	3	2	1	Bit 0
		CH1F	CH1IE	0	MS1A	ELS1B	ELS1A	TOV1	CH1MAX
	Reset:	0	0	0	0	0	0	0	0

Figure 25. TIM Channel Status and Control Registers (TSC0, TSC1)

CHxF — Channel x Flag Bit

When channel x is an input capture channel, CHxF is set when an active edge occurs on the channel x pin. When channel x is an output compare channel, CHxF is set when the value in the TIM counter registers matches the value in the TIM channel x registers.

Clear CHxF by reading the TIM channel x status and control register with CHxF set and then writing a logic 0 to CHxF.

1 = Input capture or output compare on channel x

CHxIE — Channel x Interrupt Enable Bit

1 = Channel x CPU interrupt requests enabled

MSxB, MSxA, ELSxB, and ELSxA

Table 7. Mode, Edge, and Level Selection

MSxB	MSxA	ELSxB	ELSxA	Mode	Configuration
X	0	0	0	Output preset	Pin under port control; initial output level high
X	1	0	0		Pin under port control; initial output level low
0	0	0	1	Input capture	Capture on rising edge only
0	0	1	0		Capture on falling edge only
0	0	1	1		Capture on rising or falling edge
0	1	0	1	Output compare or PWM	Toggle output on compare
0	1	1	0		Clear output on compare
0	1	1	1		Set output on compare
1	X	0	1	Buffered output compare or buffered PWM	Toggle output on compare
1	X	1	0		Clear output on compare
1	X	1	1		Set output on compare

TOVx — Toggle-On-Overflow Bit

1 = Channel x pin toggles on TIM counter overflow.

NOTE: *When TOVx is set, a TIM counter overflow takes precedence over a channel x output compare if both occur at the same time.*

CHxMAX — Channel x Maximum Duty Cycle Bit

When the TOVx bit is at logic 1, setting the CHxMAX bit forces the duty cycle of buffered and unbuffered PWM signals to 100%. The CHxMAX bit takes effect in the cycle after it is set or cleared. The output stays at the 100% duty cycle level until the cycle after CHxMAX is cleared.

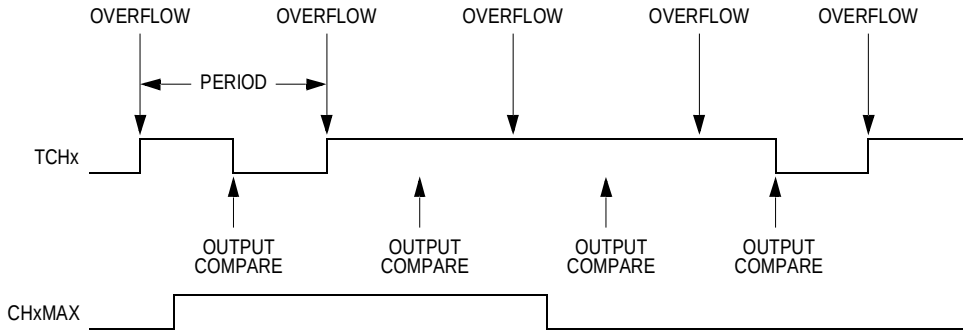


Figure 26. CHxMAX Latency

TIM Channel Registers

In input capture mode (MSxB:MSxA = 0:0), reading the high byte of the TIM channel x registers (TCHxH) inhibits input captures until the low byte (TCHxL) is read.

In output compare mode (MSxB:MSxA ≠ 0:0), writing to the high byte of the TIM channel x registers (TCHxH) inhibits output compares until the low byte (TCHxL) is written.

TCH0H	$0026	Bit 7	6	5	4	3	2	1	Bit 0
		Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
	Reset:	Indeterminate after reset							
TCH0L	$0027	Bit 7	6	5	4	3	2	1	Bit 0
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Reset:	Indeterminate after reset							
TCH1H	$0029	Bit 7	6	5	4	3	2	1	Bit 0
		Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
	Reset:	Indeterminate after reset							
TCH1L	$002A	Bit 7	6	5	4	3	2	1	Bit 0
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Reset:	Indeterminate after reset							

Figure 27. TIM Channel Registers (TCH0H:L, TCH1H:L)

Analog-to-Digital Converter (ADC)

The ADC is an 8-bit, 4-channel analog-to-digital converter. The ADC module is only available on the MC68HC908QY2, MC68HC908QT2, MC68HC908QY4, and MC68HC908QT4.

Features of the ADC module include:

- 4 channels with multiplexed input
- Linear successive approximation with monotonicity
- 8-bit resolution
- Single or continuous conversion
- Conversion complete flag or conversion complete interrupt
- Selectable ADC clock

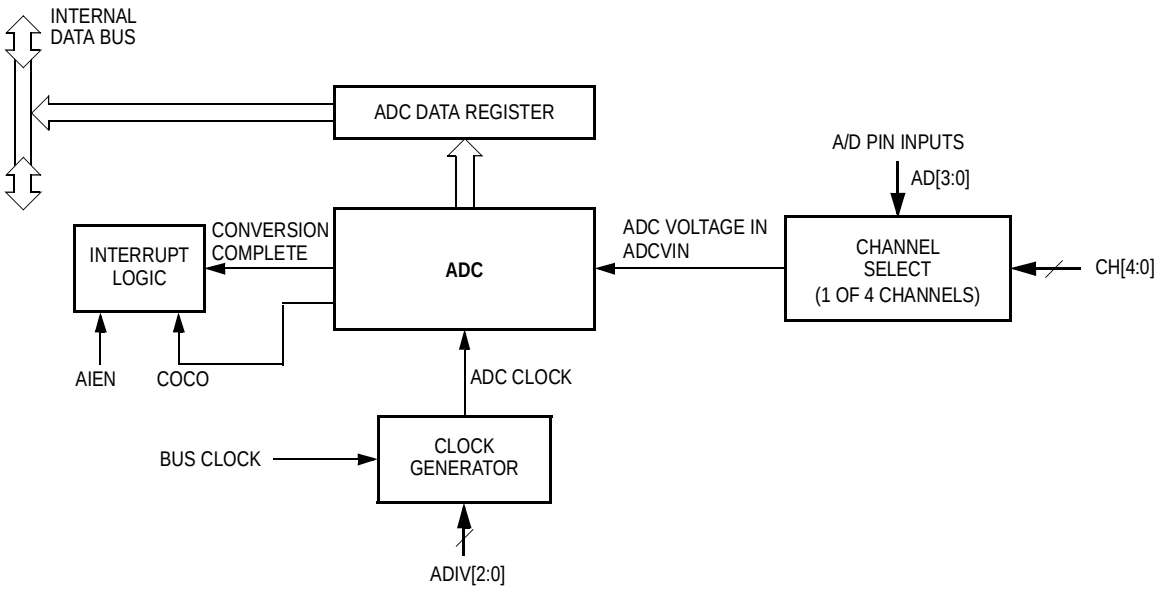


Figure 28. ADC Block Diagram

Conversion Time

$$\text{Conversion Time} = \frac{16 \text{ ADC Clock Cycles}}{\text{ADC Clock Frequency}}$$

$$\text{Number of Bus Cycles} = \text{Conversion Time} \times \text{Bus Frequency}$$

ADC Status and Control Register

$003C	Bit 7	6	5	4	3	2	1	Bit 0
	COCO	AIEN	ADCO	CH4	CH3	CH2	CH1	CH0
Reset:	0	0	0	1	1	1	1	1

Figure 29. ADC Status and Control Register (ADSCR)

COCO — Conversions Complete Bit

When the AIEN bit is a logic 0, the COCO is a read-only bit which is set each time a conversion is completed. This bit is cleared whenever ADSCR is written or whenever the ADR is read.

When the AIEN bit is a logic 1 (CPU interrupt enabled), COCO will always be logic 0 when read.

1 = Conversion completed (AIEN = 0)

AIEN — ADC Interrupt Enable Bit

1 = ADC interrupt enabled

ADCO — ADC Continuous Conversion Bit

1 = Continuous ADC conversion

0 = Single ADC conversion

CH[4:0] — ADC Channel Select Bits

NOTE: *Startup from the ADC power off state requires one conversion cycle to stabilize.*

Table 8. MUX Channel Select

CH4	CH3	CH2	CH1	CH0	ADC Channel	Input Select
0	0	0	0	0	AD0	PTA0
0	0	0	0	1	AD1	PTA1
0	0	0	1	0	AD2	PTA4
0	0	0	1	1	AD3	PTA5
0 ↓ 1	0 ↓ 1	1 ↓ 0	0 ↓ 1	0 ↓ 0	— — —	Unused[1]
1	1	0	1	1	—	Reserved
1	1	1	0	0	—	Unused
1	1	1	0	1	—	V_{DDA}[2]
1	1	1	1	0	—	V_{SSA}[2]
1	1	1	1	1	—	ADC power off

1. If any unused channels are selected, the resulting ADC conversion will be unknown.
2. The voltage levels supplied from internal reference nodes, as specified in the table, are used to verify the operation of the ADC converter both in production test and for user applications.

ADC Data Register

This register is updated each time an ADC conversion completes.

$003E	Bit 7	6	5	4	3	2	1	Bit 0
	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
Reset:	Indeterminate after reset							

Figure 30. ADC Data Register (ADR)

ADC Input Clock Register

$03F	Bit 7	6	5	4	3	2	1	Bit 0
	ADIV2	ADIV1	ADIV0	0	0	0	0	0
Reset:	0	0	0	0	0	0	0	0

Figure 31. ADC Input Clock Register (ADICLK)

ADIV2–ADIV0 — ADC Clock Prescaler Bits

Table 9. ADC Clock Divide Ratio

ADIV2	ADIV1	ADIV0	ADC Clock Rate
0	0	0	Bus clock ÷ 1
0	0	1	Bus clock ÷ 2
0	1	0	Bus clock ÷ 4
0	1	1	Bus clock ÷ 8
1	X	X	Bus clock ÷ 16

X = don't care

Input/Output (I/O) Ports

Port A

Port A is an 6-bit special function port that shares all six of its pins with the keyboard interrupt (KBI) module. Each port A pin also has a software configurable pullup device if the corresponding port pin is configured as a general-purpose input port, a KBI input, or the IRQ input. PTA3 has a fixed pullup device when configured as RST.

NOTE: *PTA2 is input only.*

Port A Data Register

$0000	Bit 7	6	5	4	3	2	1	Bit 0
	0	AWUL	PTA5	PTA4	PTA3	PTA2	PTA1	PTA0
Reset:	Unaffected by reset							
Additional Functions:			KBI5 AD3 OSC1	KBI4 AD2 OSC2	KBI3 RST	KBI2 IRQ	KBI1 AD1 TCH1	KBI0 AD0 TCH0

Figure 32. Port A Data Register (PTA)

PTA[5:0] — Port A Data Bits

These read/write bits are software programmable. Data direction of each port A pin is under the control of the corresponding bit in data direction register A (PTA2 is input only). Reset has no effect on port A data.

AWUL — Auto Wake-up Latch Data Bit

This is a read-only bit which has the value of the auto wake-up interrupt request latch. The wake-up request signal is generated internally.

Data Direction Register A

$0004	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	DDRA5	DDRA4	DDRA3	0	DDRA1	DDRA0
Reset:	0	0	0	0	0	0	0	0

Figure 33. Data Direction Register A (DDRA)

DDRA[5:0] — Data Direction Register A Bits

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

Port A Input Pullup Enable Register

$000B	Bit 7	6	5	4	3	2	1	Bit 0
	OSC2EN		PTAPUE5	PTAPUE4	PTAPUE3	PTAPUE2	PTAPUE2	PTAPUE0
Reset:	0	0	0	0	0	0	0	0

Figure 34. Port A Input Pull-up Enable Register (PTAPUE)

OSC2EN — Enable Clock Output on OSC2 Pin

This read/write bit configures the OSC2 pin function as a reference frequency output when internal oscillator or RC oscillator option is selected. This bit has no effect for the XTAL oscillator or external oscillator options.

1 = OSC2 pin outputs the internal or RC oscillator clock (BUSCLKX4)

PTAPUE[5:0] — Port A Input Pullup Enable Bits

1 = Corresponding port A pin configured to have internal pull if its DDRA bit is set to 0 and no alternate function such as KBI, IRQ, or timer controls the pin.

Port B

Port B is an 8-bit general purpose I/O port. Port B is only available on the MC68HC908QY1, MC68HC908QY2, and MC68HC908QY4.

Port B Data Register

$0001	Bit 7	6	5	4	3	2	1	Bit 0
	PTB7	PTB6	PTB5	PTB4	PTB3	PTB2	PTB1	PTB0
Reset:	Unaffected by reset							

Figure 35. Port B Data Register (PTB)

PTB[7:0] — Port B Data Bits

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

Data Direction Register B

$0005	Bit 7	6	5	4	3	2	1	Bit 0
	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
Reset:	0	0	0	0	0	0	0	0

Figure 36. Data Direction Register B (DDRB)

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

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

Port B Input Pullup Enable Register

$000C	Bit 7	6	5	4	3	2	1	Bit 0
	PTBPUE7	PTBPUE6	PTBPUE5	PTBPUE4	PTBPUE3	PTBPUE2	PTBPUE2	PTBPUE0
Reset:	0	0	0	0	0	0	0	0

Figure 37. Port B Input Pullup Enable Register (PTBPUE)

PTBPUE[7:0] — Port B Input Pullup Enable Bits

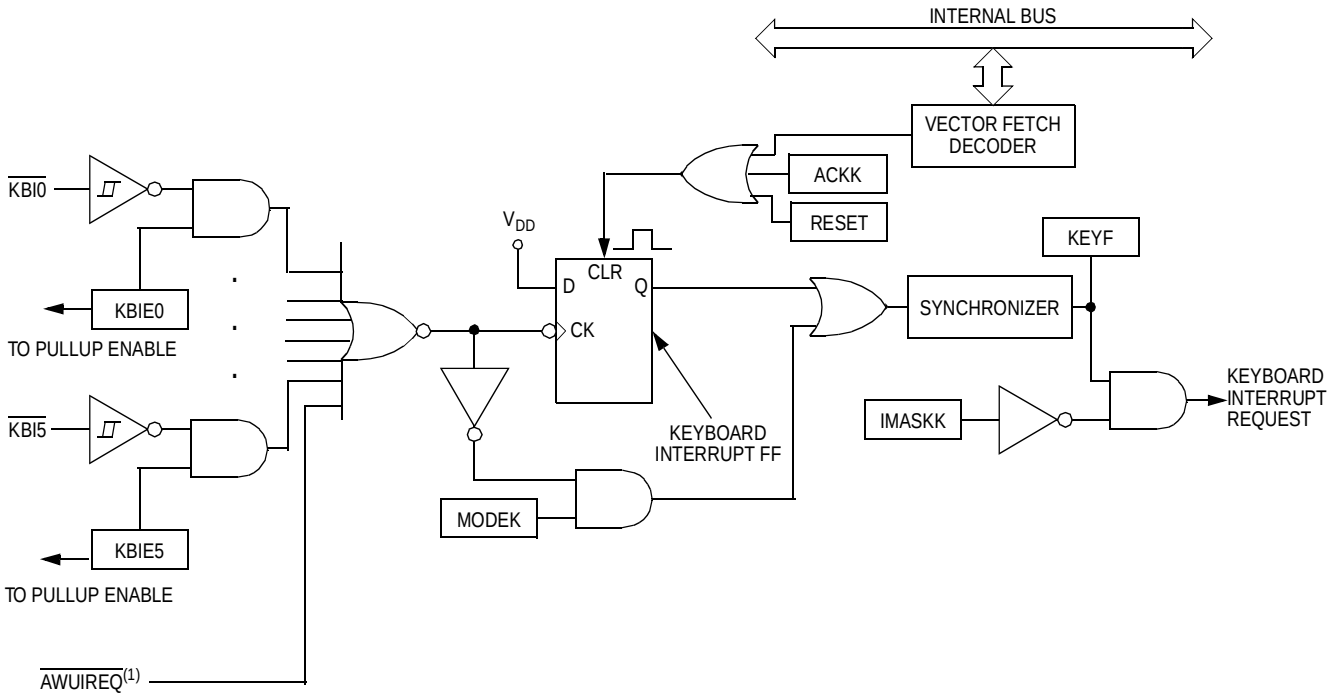
These read/write bits are software programmable to enable pullup devices on port B pins

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

Keyboard Interrupt Module (KBI)

Features of the keyboard interrupt module include:

- Six keyboard interrupt pins with separate keyboard interrupt enable bits and one keyboard interrupt mask
- One internal interrupt controlled by the auto wake-up logic, with separate interrupt enable bit, sharing the same keyboard interrupt mask
- Pullup device if input pin is configured as a keyboard interrupt input
- Programmable edge-only or edge and level interrupt sensitivity
- Exit from low-power modes



1. For AWUGEN logic refer to **Figure 39**.

Figure 38. Keyboard Interrupt Block Diagram

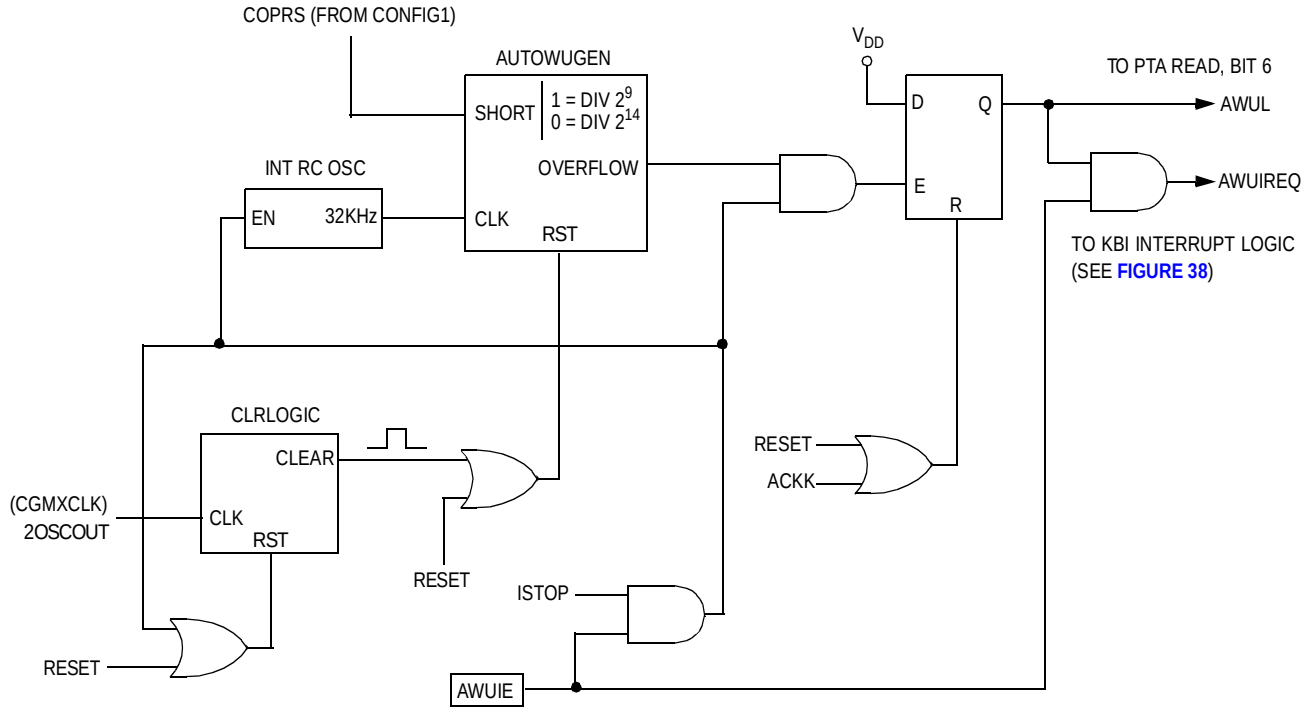


Figure 39. Auto Wake-up Interrupt Request Generation Logic

Keyboard Status and Control Register

$001A	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	0	0	KEYF	ACKK	IMASKK	MODEK
Reset:	0	0	0	0	0	0	0	0

Figure 40. Keyboard Status and Control Register (KBSCR)

KEYF — Keyboard Flag Bit

1 = Keyboard interrupt pending

ACKK — Keyboard Acknowledge Bit

Writing a logic 1 to this write-only bit clears the keyboard interrupt request on port A and auto wake-up logic. ACKK always reads as logic 0.

IMASKK— Keyboard Interrupt Mask Bit

1 = Keyboard interrupt requests masked (disabled)

MODEK — Keyboard Triggering Sensitivity Bit

1 = Keyboard interrupt requests on falling edges and low levels
0 = Keyboard interrupt requests on falling edges only

Keyboard Interrupt Enable Register

$001B	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	AWUIE	KBIE5	KBIE4	KBIE3	KBIE2	KBIE1	KBIE0
Reset:	0	0	0	0	0	0	0	0

Figure 41. Keyboard Interrupt Enable Register (KBIER)

KBIE5–KBIE0 — Port A Keyboard Interrupt Enable Bits
1 = KBIx pin enabled as keyboard interrupt pin

AWUIE — Auto Wake-up Interrupt Enable Bit
1 = Auto wake-up enabled as interrupt input

NOTE: *Auto wakeup timeout period is determined by COPRS bit in CONFIG1:*
COPRS = 0 — 512 ms (approximately)
COPRS = 1 — 16 ms (approximately)

Break Module

This section describes the breakpoint module which works in conjunction with third-party development software to allow development of debugging of application systems.

Break Status and Control Register

$FE0B	Bit 7	6	5	4	3	2	1	Bit 0
	BRKE	BRKA	0	0	0	0	0	0
Reset:	0	0	0	0	0	0	0	0

Figure 42. Break Status and Control Register (BRKSCR)

BRKE — Break Enable Bit

This read/write bit enables breaks on break address register matches.
1 = Breaks enabled on 16-bit address match
0 = Breaks disabled

BRKA — Break Active Bit

This read/write status and control bit is set when a break address match occurs. Writing a logic 1 to BRKA generates a break interrupt. Clear BRKA by writing a logic 0 to it before exiting the break routine.
1 = Break address match

Break Address Registers

The break address registers (BRKH and BRKL) contain the high and low bytes of the desired breakpoint address.

$FE09	Bit 7	6	5	4	3	2	1	Bit 0
	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

Figure 43. Break Address Register High (BRKH)

$FE0A	Bit 7	6	5	4	3	2	1	Bit 0
	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

Figure 44. Break Address Register Low (BRKL)

Break Auxiliary Register

The break auxiliary register (BRKAR) contains a bit that enables software to disable the COP while the MCU is in a state of break interrupt with monitor mode.

$FE02	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	0	0	0	0	0	BDCOP
Reset:	0	0	0	0	0	0	0	0

Figure 45. Break Auxiliary Register (BRKAR)

BDCOP — Break Disable COP Bit

1 = COP disabled during break interrupt

Break Flag Control Register

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

$FE03	Bit 7	6	5	4	3	2	1	Bit 0
	BCFE	R	R	R	R	R	R	R
Reset:	0							

R = Reserved

Figure 46. Break Flag Control Register (BFCR)

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

Break Status Register

The break status register (BSR) is reserved for use in supporting third party emulation systems.

$FE00	Bit 7	6	5	4	3	2	1	Bit 0
	R	R	R	R	R	R	SBSW	R
Reset:							0	

R = Reserved

Figure 47. Break Status Register (BSR)

Condensed Electrical Characteristics

For more detailed information refer to the *MC68HC908QY4 Data Sheet* (Motorola document order number MC68HC908QY4/D).

5-Volt DC Electrical Characteristics

Characteristic[1]	Symbol	Min	Typ[2]	Max	Unit
V_{DD} supply current Run, f_{OP} = 4 MHz[3] Wait[4] Stop[5], –40°C to 85°C	I_{DD}	 — — —	 7 5 1	 10 5.5 5	 mA mA µA
POR rearm voltage[6]	V_{POR}	0	—	100	mV
POR rise time ramp rate[7]	R_{POR}	0.035	—	—	V/ms
Monitor mode entry voltage	$V_{DD}+V_{HI}$	V_{DD} + 2.5	—	9.1	V
Pullup resistors[8] $\overline{RST}$, $\overline{IRQ}$, PTA0–PTA5, PTB0–PTB7	R_{PU}	16	26	36	kΩ
Low-voltage inhibit reset, trip falling voltage	V_{TRIPF}	3.90	4.20	4.50	V
Low-voltage inhibit reset, trip rising voltage	V_{TRIPR}	4.00	4.30	4.60	V
Low-voltage inhibit reset/recover hysteresis	V_{HYS}	—	100	—	mV

1. V_{DD} = 4.5 to 5.5 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. Run (operating) I_{DD} measured using external square wave clock source. All inputs 0.2 V from rail. No dc loads. Less than 100 pF on all outputs. All ports configured as inputs. Measured with all modules enabled.
4. Wait I_{DD} measured using external square wave clock source (f_{OP} = 4MHz); all inputs 0.2 V from rail; no dc loads; less than 100 pF on all outputs. All ports configured as inputs.
5. All ports configured as inputs. All ports driven 0.2 V or less from rail. No dc loads. On the 8-pin versions, port B is configured as inputs with pullups enabled.
6. Maximum is highest voltage that POR is guaranteed.
7. If minimum V_{DD} is not reached before the internal POR reset is released, $\overline{RST}$ must be driven low externally until minimum V_{DD} is reached.
8. R_{PU1} and R_{PU2} are measured at V_{DD} = 5.0 V.

5-Volt Control Timing

Characteristic[1]	Symbol	Min	Max	Unit
Internal operating frequency[2]	f_{OP}	—	8	MHz
$\overline{RST}$ input pulse width low[3]	t_{IRL}	750	—	ns

1. V_{DD} = 4.5 to 5.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H; timing shown with respect to 20% V_{DD} and 70% V_{SS}, unless otherwise noted.
2. Some modules may require a minimum frequency greater than dc for proper operation; see appropriate table for this information.
3. Minimum pulse width reset is guaranteed to be recognized. It is possible for a smaller pulse width to cause a reset.

5-Volt Oscillator Characteristics

Characteristic	Symbol	Min	Typ	Max	Unit
Internal oscillator frequency	f_{INTCLK}	—	12.8	—	MHz
Crystal frequency, XTALCLK	$f_{OSCXCLK}$	1	—	32	MHz
RC oscillator frequency, RCCLK	f_{RCCLK}	2	—	12	MHz
External clock reference frequency[1]	$f_{OSCXCLK}$	dc	—	32	MHz

1. No more than 10% duty cycle deviation from 50%.

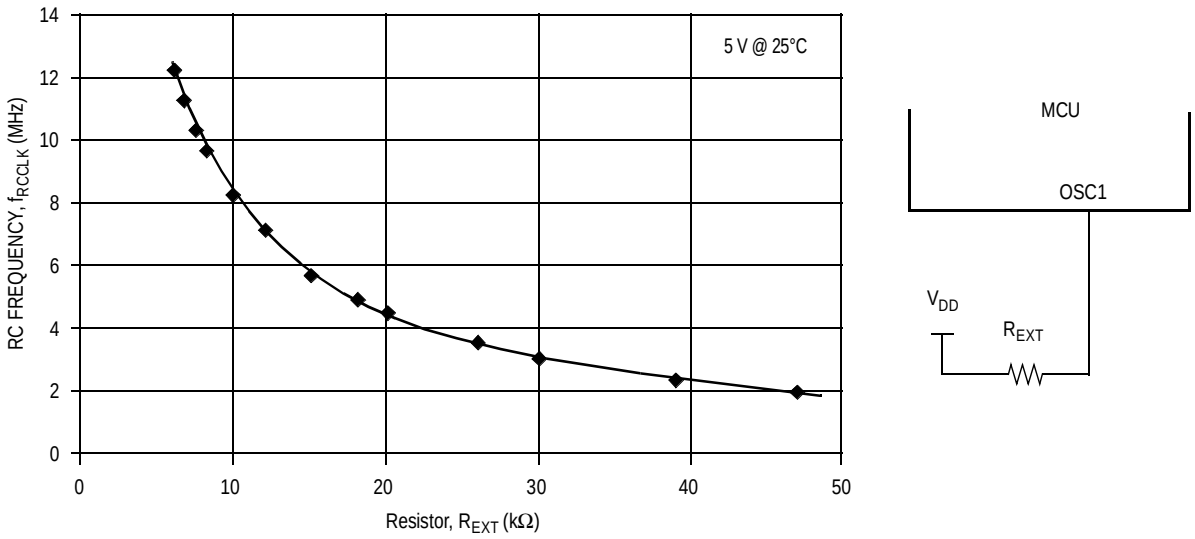


Figure 48. RC versus Frequency (5 Volts @ 25°C)

3-Volt DC Electrical Characteristics

Characteristic[1]	Symbol	Min	Typ[2]	Max	Unit
V_{DD} supply current Run, f_{OP} = 2 MHz[3] Wait, f_{OP} = 2 MHz[4] Stop[5],–40°C to 85°C	I_{DD}	 — — —	 5 1 1	 8 2.5 5	 mA mA µA
POR rearm voltage[6]	V_{POR}	0	—	100	mV
POR rise time ramp rate[7]	R_{POR}	0.035	—	—	V/ms
Monitor mode entry voltage	$V_{DD}+V_{HI}$	V_{DD} + 2.5	—	V_{DD} + 4.0	V
Pullup resistors[8] $\overline{RST}$, $\overline{IRQ}$, 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. Run (operating) I_{DD} measured using external square wave clock source. All inputs 0.2 V from rail. No dc loads. Less than 100 pF on all outputs. C_L = 20 pF on OSC2. All ports configured as inputs. OSC2 capacitance linearly affects run I_{DD}. Measured with all modules enabled.
4. Wait I_{DD} measured using external square wave clock source (f_{OP} = 4 MHz); all inputs 0.2 V from rail; no dc loads; less than 100 pF on all outputs. C_L = 20 pF on OSC2; all ports configured as inputs; OSC2 capacitance linearly affects wait I_{DD}.
5. All ports configured as inputs. All ports driven 0.2 V or less from rail. No dc loads. On the 8-pin versions, port B is configured as inputs with pullups enabled.
6. Maximum is highest voltage that POR is guaranteed.
7. If minimum V_{DD} is not reached before the internal POR reset is released, $\overline{RST}$ must be driven low externally until minimum V_{DD} is reached.
8. R_{PU1} and R_{PU2} are measured at V_{DD} = 5.0 V

3-Volt Control Timing

Characteristic[1]	Symbol	Min	Max	Unit
Internal operating frequency[2]	f_{OP}	—	4	MHz
$\overline{RST}$ input pulse width low[3]	t_{IRL}	1.5	—	µs

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. Some modules may require a minimum frequency greater than dc for proper operation; see appropriate table for this information.
3. Minimum pulse width reset is guaranteed to be recognized. It is possible for a smaller pulse width to cause a reset.

3-Volt Oscillator Characteristics

Characteristic	Symbol	Min	Typ	Max	Unit
Internal oscillator frequency	f_{INTCLK}	—	12.8	—	MHz
Crystal frequency, XTALCLK	$f_{OSCXCLK}$	1	—	16	MHz
RC oscillator frequency, RCCLK	f_{RCCLK}	2	—	12	MHz
External clock reference frequency(1)	$f_{OSCXCLK}$	dc	—	16	MHz

1. No more than 10% duty cycle deviation from 50%

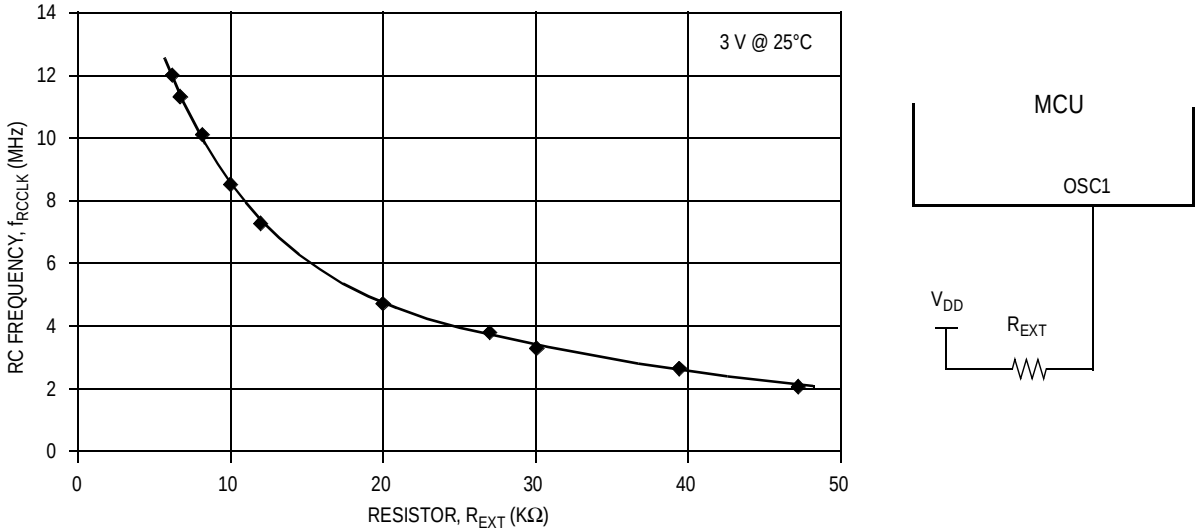


Figure 49. RC versus Frequency (3 Volts @ 25°C)

Typical Supply Currents

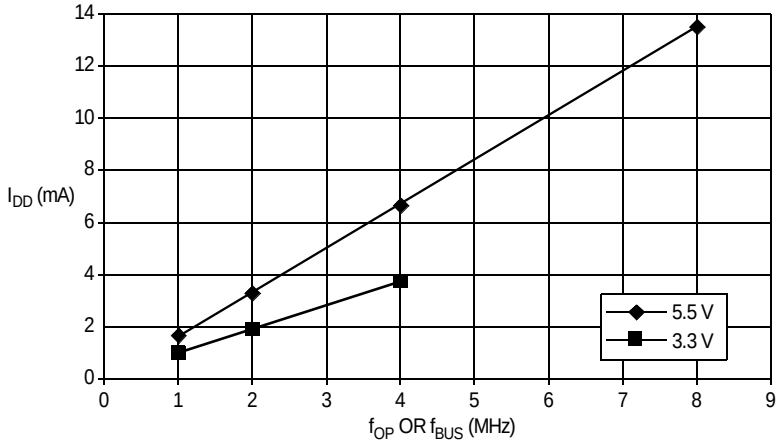


Figure 50. Typical Operating I_{DD}, with All Modules Turned On (25°C)

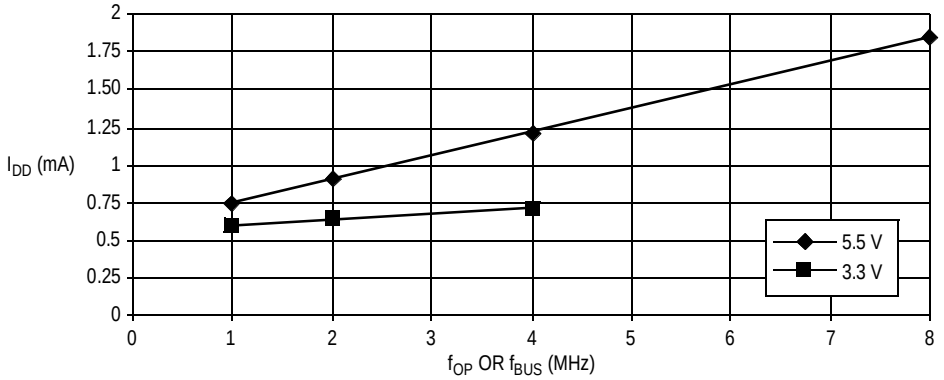


Figure 51. Typical Wait Mode I_{DD}, with ADC Turned On (25°C)

Analog-to-Digital Converter Characteristics

Characteristic	Symbol	Min	Max	Unit	Comments
Input voltages	V_{ADIN}	V_{SS}	V_{DD}	V	—
Resolution	B_{AD}	8	8	Bits	—
Absolute accuracy	A_{AD}	± 0.5	± 1.5	LSB	Includes quantization
ADC internal clock	f_{ADIC}	0.5	1.048	MHz	$t_{ADIC} = 1/f_{ADIC}$, tested only at 1 MHz
Conversion range	R_{AD}	V_{SS}	V_{DD}	V	—
Power-up time	t_{ADPU}	16		t_{ADIC} cycles	$t_{ADIC} = 1/f_{ADIC}$
Conversion time	t_{ADC}	16	17	t_{ADIC} cycles	$t_{ADIC} = 1/f_{ADIC}$
Sample time(1)	t_{ADS}	5	—	t_{ADIC} cycles	$t_{ADIC} = 1/f_{ADIC}$
Zero input reading(2)	Z_{ADI}	00	01	Hex	$V_{IN} = V_{SS}$
Full-scale reading(3)	F_{ADI}	FE	FF	Hex	$V_{IN} = V_{DD}$
Input capacitance	C_{ADI}	—	8	pF	Not tested
Input leakage(3)	—	—	± 1	μA	—

1. Source impedances greater than 10 kΩ may adversely affect internal RC charging time during input sampling.
2. Zero-input/full-scale reading requires sufficient decoupling measures for accurate conversions.
3. The external system error caused by input leakage current is approximately equal to the product of R source and input current.

Memory Characteristics

Characteristic	Symbol	Min	Max	Unit
RAM data retention voltage	V_{RDR}	1.3	—	V
FLASH program bus clock frequency	—	1	—	MHz
FLASH read bus clock frequency	f_{Read}(1)	32 k	8M	Hz
FLASH page erase time <1 K cycles <10 K cycles	t_{Erase}(2)	 1 4	 — —	ms
FLASH mass erase time	t_{MErase}(3)	4	—	ms
FLASH PGM/ERASE to HVEN set up time	t_{nvs}	10	—	us
FLASH high-voltage hold time	t_{nvh}	5	—	us
FLASH high-voltage hold time (mass erase)	t_{nvhl}	100	—	us
FLASH program hold time	t_{pgs}	5	—	us
FLASH program time	t_{PROG}	30	40	us
FLASH return to read time	t_{rcv}(4)	1	—	us
FLASH cumulative program hv period	t_{HV}(5)	—	4	ms
FLASH row erase endurance(6)	—	10 k	—	Cycles
FLASH row program endurance(7)	—	10 k	—	Cycles
FLASH data retention time(8)	—	10	—	Years

1. f_{Read} is defined as the frequency range for which the FLASH memory can be read.
2. If the page erase time is longer than t_{Erase} (Min), there is no erase disturb, but it reduces the endurance of the FLASH memory.
3. If the mass erase time is longer than t_{MErase} (Min), there is no erase disturb, but it reduces the endurance of the FLASH memory.
4. t_{rcv} is defined as the time it needs before the FLASH can be read after turning off the high voltage charge pump, by clearing HVEN to logic 0.
5. t_{HV} is defined as the cumulative high voltage programming time to the same row before next erase.
   t_{HV} must satisfy this condition: $t_{nvs} + t_{nvh} + t_{pgs} + (t_{PROG} \times 32) \leq t_{HV}$ max.
6. The minimum row endurance value specifies each row of the FLASH memory is guaranteed to work for at least this many erase/program cycles.
7. The minimum row endurance value specifies each row of the FLASH memory is guaranteed to work for at least this many erase/program cycles.
8. The FLASH is guaranteed to retain data over the entire operating temperature range for at least the minimum time specified.

HOW TO REACH US:

USA/EUROPE/LOCATIONS NOT LISTED:

Motorola Literature Distribution;
P.O. Box 5405, Denver, Colorado 80217
1-303-675-2140 or 1-800-441-2447

JAPAN:

Motorola Japan Ltd.; SPS, Technical Information Center,
3-20-1, Minami-Azabu Minato-ku, Tokyo 106-8573 Japan
81-3-3440-3569

ASIA/PACIFIC:

Motorola Semiconductors H.K. Ltd.;
Silicon Harbour Centre, 2 Dai King Street,
Tai Po Industrial Estate, Tai Po, N.T., Hong Kong
852-26668334

TECHNICAL INFORMATION CENTER:

1-800-521-6274

HOME PAGE:

http://www.motorola.com/semiconductors

MC68HC908QY4SM/D