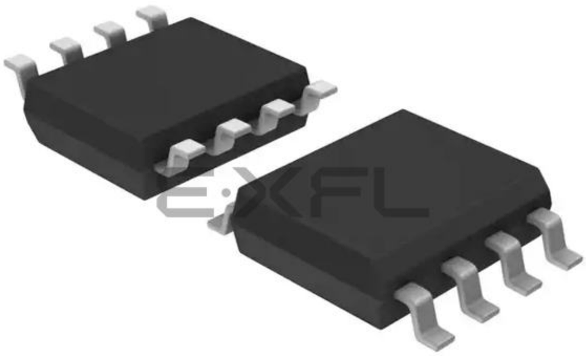


Welcome to E-XFL.COM

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	5
Program Memory Size	4KB (4K 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 6x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	8-SOIC (0.209", 5.30mm Width)
Supplier Device Package	8-SO
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc908qt4acdwer

Chapter 11
Oscillator (OSC) Module

Chapter 12
Input/Output Ports (PORTS)

Chapter 15
Development Support

Chapter 1
General Description

1.1 Introduction

The MC68HC908QY4A is a member of the low-cost, high-performance M68HC08 Family of 8-bit microcontroller units (MCUs). 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 1-1. Summary of Device Variations

Device	FLASH Memory Size	ADC	Pin Count
MC68HC908QT1A	1536 bytes	—	8 pins
MC68HC908QT2A	1536 bytes	6 channel, 10 bit	8 pins
MC68HC908QT4A	4096 bytes	6 channel, 10 bit	8 pins
MC68HC908QY1A	1536 bytes	—	16 pins
MC68HC908QY2A	1536 bytes	6 channel, 10 bit	16 pins
MC68HC908QY4A	4096 bytes	6 channel, 10 bit	16 pins

1.2 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
  - Software selectable 1 MHz, 2 MHz, or 3.2 MHz internal bus operation
  - 8-bit trim capability
  - ±25% untrimmed
  - Trimmable to approximately 0.4%(1)
- Software selectable crystal oscillator range, 32–100 kHz, 1–8 MHz and 8–32 MHz
- Software configurable input clock from either internal or external source
- Auto wakeup from STOP capability using dedicated internal 32-kHz RC or bus clock source
- On-chip in-application programmable FLASH memory
  - Internal program/erase voltage generation
  - Monitor ROM containing user callable program/erase routines
  - FLASH security(2)

1. See 16.11 Oscillator Characteristics for internal oscillator specifications
2. No security feature is absolutely secure. However, Freescale's strategy is to make reading or copying the FLASH difficult for unauthorized users.

2.6 FLASH Memory (FLASH)

The FLASH memory is intended primarily for program storage. In-circuit programming allows the operating program to be loaded into the FLASH memory after final assembly of the application product. It is possible to program the entire array through the single-wire monitor mode interface. Because no special voltages are needed for FLASH erase and programming operations, in-application programming is also possible through other software-controlled communication paths.

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

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.

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.

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

= Unimplemented

Figure 2-3. FLASH Control Register (FLCR)

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.

Table 7-1. Instruction Set Summary (Sheet 3 of 6)

Source Form	Operation	Description	V	H	I	N	Z	C	Address Mode	Opcode	Operand	Cycles
CLR *opr*	Clear	M ← $00	0	–	–	0	1	–	DIR	3F	dd	3
CLRA		A ← $00							INH	4F		1
CLRX		X ← $00							INH	5F		1
CLRH		H ← $00							INH	8C		1
CLR *opr*,X		M ← $00							IX1	6F	ff	3
CLR ,X		M ← $00							IX	7F		2
CLR *opr*,SP		M ← $00							SP1	9E6F	ff	4
CMP #*opr*	Compare A with M	(A) – (M)	↕	–	–	↕	↕	↕	IMM	A1	ii	2
CMP *opr*									DIR	B1	dd	3
CMP *opr*									EXT	C1	hh ll	4
CMP *opr*,X									IX2	D1	ee ff	4
CMP *opr*,X									IX1	E1	ff	3
CMP ,X									IX	F1		2
CMP *opr*,SP									SP1	9EE1	ff	4
CMP *opr*,SP									SP2	9ED1	ee ff	5
COM *opr*	Complement (One's Complement)	M ← $(\overline{M})$ = $FF – (M)	0	–	–	↕	↕	1	DIR	33	dd	4
COMA		A ← $(\overline{A})$ = $FF – (M)							INH	43		1
COMX		X ← $(\overline{X})$ = $FF – (M)							INH	53		1
COM *opr*,X		M ← $(\overline{M})$ = $FF – (M)							IX1	63	ff	4
COM ,X		M ← $(\overline{M})$ = $FF – (M)							IX	73		3
COM *opr*,SP		M ← $(\overline{M})$ = $FF – (M)							SP1	9E63	ff	5
CPHX #*opr*	Compare H:X with M	(H:X) – (M:M + 1)	↕	–	–	↕	↕	↕	IMM	65	ii ii+1	3
CPHX *opr*									DIR	75	dd	4
CPX #*opr*	Compare X with M	(X) – (M)	↕	–	–	↕	↕	↕	IMM	A3	ii	2
CPX *opr*									DIR	B3	dd	3
CPX *opr*									EXT	C3	hh ll	4
CPX ,X									IX2	D3	ee ff	4
CPX *opr*,X									IX1	E3	ff	3
CPX *opr*,X									IX	F3		2
CPX *opr*,SP									SP1	9EE3	ff	4
CPX *opr*,SP									SP2	9ED3	ee ff	5
DAA	Decimal Adjust A	$(A)_{10}$	U	–	–	↕	↕	↕	INH	72		2
DBNZ *opr,rel*	Decrement and Branch if Not Zero	A ← (A) – 1 or M ← (M) – 1 or X ← (X) – 1 PC ← (PC) + 3 + *rel* ? (result) ≠ 0	–	–	–	–	–	–	DIR	3B	dd rr	5
DBNZA *rel*		PC ← (PC) + 2 + *rel* ? (result) ≠ 0							INH	4B	rr	3
DBNZX *rel*		PC ← (PC) + 2 + *rel* ? (result) ≠ 0							INH	5B	rr	3
DBNZ *opr*,X,*rel*		PC ← (PC) + 3 + *rel* ? (result) ≠ 0							IX1	6B	ff rr	5
DBNZ X,*rel*		PC ← (PC) + 2 + *rel* ? (result) ≠ 0							IX	7B	rr	4
DBNZ *opr*,SP,*rel*		PC ← (PC) + 4 + *rel* ? (result) ≠ 0							SP1	9E6B	ff rr	6
DEC *opr*	Decrement	M ← (M) – 1	↕	–	–	↕	↕	–	DIR	3A	dd	4
DECA		A ← (A) – 1							INH	4A		1
DECX		X ← (X) – 1							INH	5A		1
DEC *opr*,X		M ← (M) – 1							IX1	6A	ff	4
DEC ,X		M ← (M) – 1							IX	7A		3
DEC *opr*,SP		M ← (M) – 1							SP1	9E6A	ff	5
DIV	Divide	A ← (H:A)/(X) H ← Remainder	–	–	–	–	↕	↕	INH	52		7
EOR #*opr*	Exclusive OR M with A	A ← (A ⊕ M)	0	–	–	↕	↕	–	IMM	A8	ii	2
EOR *opr*									DIR	B8	dd	3
EOR *opr*									EXT	C8	hh ll	4
EOR *opr*,X									IX2	D8	ee ff	4
EOR *opr*,X									IX1	E8	ff	3
EOR ,X									IX	F8		2
EOR *opr*,SP									SP1	9EE8	ff	4
EOR *opr*,SP									SP2	9ED8	ee ff	5
INC *opr*	Increment	M ← (M) + 1	↕	–	–	↕	↕	–	DIR	3C	dd	4
INCA		A ← (A) + 1							INH	4C		1
INCX		X ← (X) + 1							INH	5C		1
INC *opr*,X		M ← (M) + 1							IX1	6C	ff	4
INC ,X		M ← (M) + 1							IX	7C		3
INC *opr*,SP		M ← (M) + 1							SP1	9E6C	ff	5

8.8 Registers

The IRQ status and control register (INTSCR) controls and monitors operation of the IRQ module. The INTSCR:

- Shows the state of the IRQ flag
- Clears the IRQ latch
- Masks the IRQ interrupt request
- Controls triggering sensitivity of the $\overline{\text{IRQ}}$ interrupt pin

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

= Unimplemented

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

IRQF — IRQ Flag Bit

This read-only status bit is set when the IRQ interrupt is pending.

1 = $\overline{\text{IRQ}}$ interrupt pending
0 = $\overline{\text{IRQ}}$ interrupt not pending

ACK — IRQ Interrupt Request Acknowledge Bit

Writing a 1 to this write-only bit clears the IRQ latch. ACK always reads 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 $\overline{\text{IRQ}}$ pin.

1 = $\overline{\text{IRQ}}$ interrupt request on falling edges and low levels
0 = $\overline{\text{IRQ}}$ interrupt request on falling edges only

9.8.2 Keyboard Interrupt Enable Register (KBIER)

KBIER enables or disables each keyboard interrupt pin.

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

= Unimplemented

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

KBIE5–KBIE0 — Keyboard Interrupt Enable Bits

Each of these read/write bits enables the corresponding keyboard interrupt pin to latch KBI interrupt requests.

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).

9.8.3 Keyboard Interrupt Polarity Register (KBIPR)

KBIPR determines the polarity of the enabled keyboard interrupt pin and enables the appropriate pullup or pulldown device.

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	KBIP5	KBIP4	KBIP3	KBIP2	KBIP1	KBIP0
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 9-5. Keyboard Interrupt Polarity Register (KBIPR)

KBIP5–KBIP0 — Keyboard Interrupt Polarity Bits

Each of these read/write bits enables the polarity of the keyboard interrupt detection.

1 = Keyboard polarity is high level and/or rising edge
0 = Keyboard polarity is low level and/or falling edge

10.4 LVI Interrupts

The LVI module does not generate interrupt requests.

10.5 Low-Power Modes

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

10.5.1 Wait Mode

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

10.5.2 Stop Mode

If the LVIPWRD bit in the configuration register is cleared and the LVISTOP bit in the configuration register is set, the LVI module remains active. If enabled to generate resets, the LVI module can generate a reset and bring the MCU out of stop mode.

10.6 Registers

The LVI status register (LVISR) contains a status bit that is useful when the LVI is enabled and LVI reset is disabled.

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

(shaded) = Unimplemented   R = Reserved

Figure 10-2. 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}. (See Table 10-1).

Table 10-1. LVIOUT Bit Indication

V_{DD}	LVIOUT
$V_{DD} > V_{TRIPR}$	0
$V_{DD} < V_{TRIPF}$	1
$V_{TRIPF} < V_{DD} < V_{TRIPR}$	Previous value

11.3.5 RC Oscillator

The RC oscillator circuit is designed for use with an external resistor (R_{EXT}) to provide a clock source with a tolerance within 25% of the expected frequency. See Figure 11-3.

The capacitor (C) for the RC oscillator is internal to the MCU. The R_{EXT} value must have a tolerance of 1% or less to minimize its effect on the frequency.

In this configuration, the OSC2 pin can be used as general-purpose input/output (I/O) port pins or other alternative pin function. The OSC2EN bit can be set to enable the OSC2 output function on the pin. Enabling the OSC2 output can affect the external RC oscillator frequency, f_{RCCLK}.

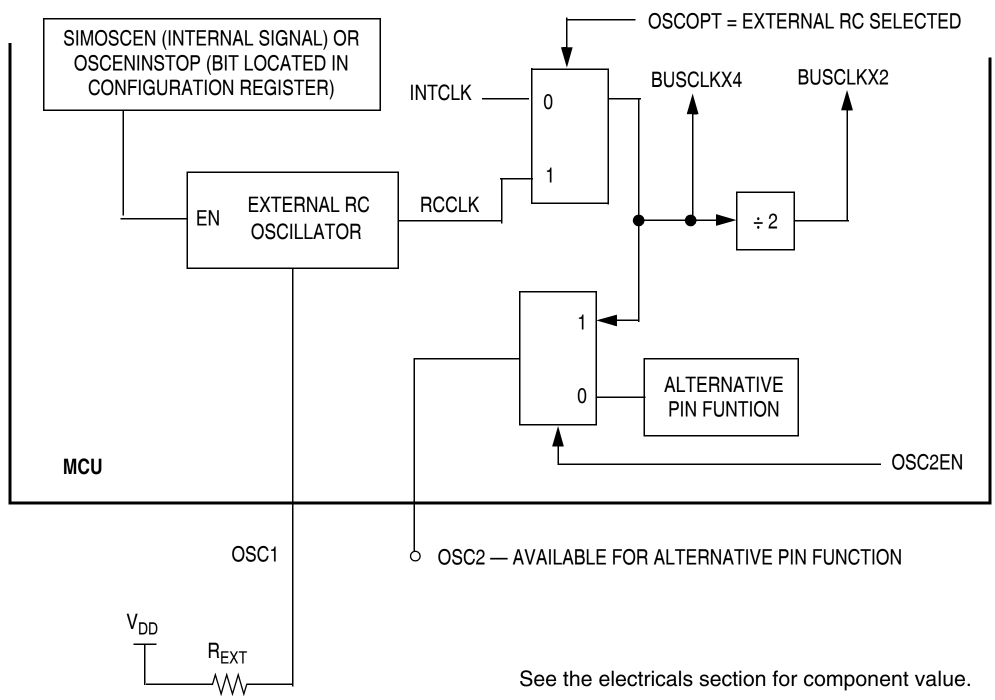


Figure 11-3. RC Oscillator External Connections

11.4 Interrupts

There are no interrupts associated with the OSC module.

11.5 Low-Power Modes

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

11.5.1 Wait Mode

The OSC module remains active in wait mode.

11.5.2 Stop Mode

The OSC module can be configured to remain active in stop mode by setting OSCENINSTOP located in a configuration register.

ECFS1:ECFS0 — External Crystal Frequency Select Bits

These read/write bits enable the specific amplifier for the crystal frequency range. Refer to oscillator characteristics table in the Electricals section for information on maximum external clock frequency versus supply voltage.

ECFS1	ECFS0	External Crystal Frequency
0	0	8 MHz – 32 MHz
0	1	1 MHz – 8 MHz
1	0	32 kHz – 100 kHz
1	1	Reserved

ECGON — External Clock Generator On Bit

This read/write bit enables the OSC1 pin as the clock input to the MCU, so that the switching process can be initiated. This bit is cleared by reset. This bit is ignored in monitor mode with the internal oscillator bypassed.

1 = External clock enabled
0 = External clock disabled

ECGST — External Clock Status Bit

This read-only bit indicates whether an external clock source is engaged to drive the system clock.

1 = An external clock source engaged
0 = An external clock source disengaged

11.8.2 Oscillator Trim Register (OSCTRIM)

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	TRIM7	TRIM6	TRIM5	TRIM4	TRIM3	TRIM2	TRIM1	TRIM0
Reset:	1	0	0	0	0	0	0	0

Figure 11-5. Oscillator Trim Register (OSCTRIM)

TRIM7–TRIM0 — Internal Oscillator Trim Factor Bits

These read/write bits change the internal capacitance used by the internal oscillator. By measuring the period 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 oscillator period. The oscillator period is based on the oscillator frequency selected by the ICFS bits in OSCSC.

Applications using the internal oscillator should copy the internal oscillator trim value at location $FFC0 or $FFC1 into this register to trim the clock source.

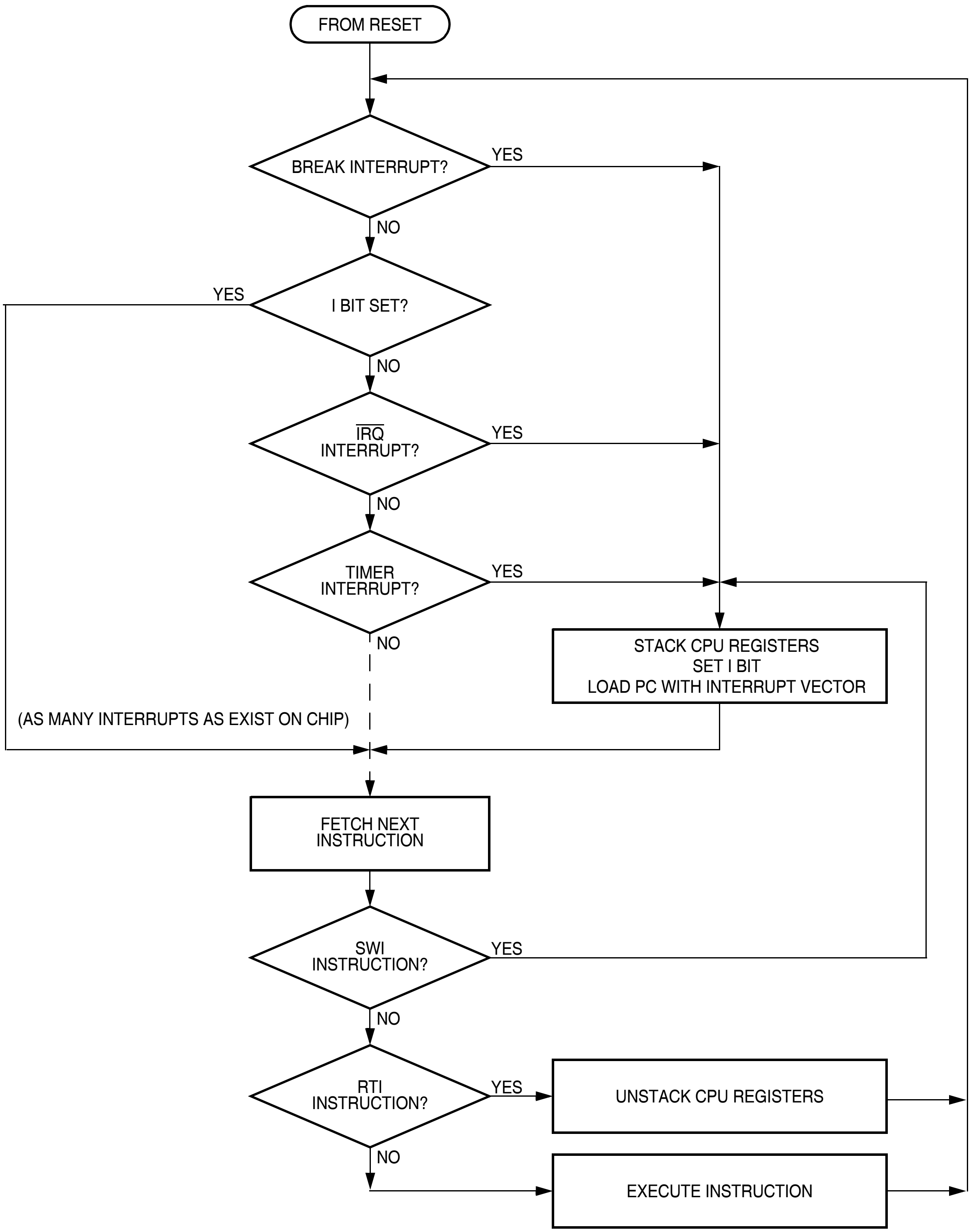


Figure 13-7. Interrupt Processing

14.7 I/O Signals

The TIM module can share its pins with the general-purpose I/O pins. See Figure 14-1 for the port pins that are shared.

14.7.1 TIM Channel I/O Pins (TCH1:TCH0)

Each channel I/O pin is programmable independently as an input capture pin or an output compare pin. TCH0 can be configured as buffered output compare or buffered PWM pin.

14.7.2 TIM Clock Pin (TCLK)

TCLK is an external clock input that can be the clock source for the counter instead of the prescaled internal bus clock. Select the TCLK input by writing 1s to the three prescaler select bits, PS[2:0]. 14.8.1 TIM Status and Control Register The minimum TCLK pulse width is specified in the Timer Interface Module Characteristics table in the Electricals section. The maximum TCLK frequency is the least of 4 MHz or bus frequency ÷ 2.

14.8 Registers

The following registers control and monitor operation of the TIM:

- TIM status and control register (TSC)
- TIM control registers (TCNTH:TCNTL)
- TIM counter modulo registers (TMODH:TMODL)
- TIM channel status and control registers (TSC0 and TSC1)
- TIM channel registers (TCH0H:TCH0L and TCH1H:TCH1L)

14.8.1 TIM Status and Control Register

The TIM status and control register (TSC) does the following:

- Enables TIM overflow interrupts
- Flags TIM overflows
- Stops the counter
- Resets the counter
- Prescales the counter clock

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

= Unimplemented

Figure 14-4. TIM Status and Control Register (TSC)

TOF — TIM Overflow Flag Bit

This read/write flag is set when the counter reaches the modulo value programmed in the TIM counter modulo registers. Clear TOF by reading the TSC register when TOF is set and then writing a 0 to TOF.

The break interrupt timing is:

- When a break address is placed at the address of the instruction opcode, the instruction is not executed until after completion of the break interrupt routine.
- When a break address is placed at an address of an instruction operand, the instruction is executed before the break interrupt.
- When software writes a 1 to the BRKA bit, the break interrupt occurs just before the next instruction is executed.

By updating a break address and clearing the BRKA bit in a break interrupt routine, a break interrupt can be generated continuously.

CAUTION

A break address should be placed at the address of the instruction opcode. When software does not change the break address and clears the BRKA bit in the first break interrupt routine, the next break interrupt will not be generated after exiting the interrupt routine even when the internal address bus matches the value written in the break address registers.

15.2.1.1 Flag Protection During Break Interrupts

The system integration module (SIM) controls whether or not module status bits can be cleared during the break state. The BCFE bit in the break flag control register (BFCR) enables software to clear status bits during the break state. See 13.8.2 Break Flag Control Register and the **Break Interrupts** subsection for each module.

15.2.1.2 TIM During Break Interrupts

A break interrupt stops the timer counter.

15.2.1.3 COP During Break Interrupts

The COP is disabled during a break interrupt with monitor mode when BDCOP bit is set in break auxiliary register (BRKAR).

15.2.2 Break Module Registers

These registers control and monitor operation of the break module:

- Break status and control register (BRKSCR)
- Break address register high (BRKH)
- Break address register low (BRKL)
- Break status register (BSR)
- Break flag control register (BFCR)

15.2.2.1 Break Status and Control Register

The break status and control register (BRKSCR) contains break module enable and status bits.

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

= Unimplemented

Figure 15-3. Break Status and Control Register (BRKSCR)

BRKE — Break Enable Bit

This read/write bit enables breaks on break address register matches. Clear BRKE by writing a 0 to bit 7. Reset clears the BRKE bit.

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 1 to BRKA generates a break interrupt. Clear BRKA by writing a 0 to it before exiting the break routine. Reset clears the BRKA bit.

1 = Break address match
0 = No break address match

15.2.2.2 Break Address Registers

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

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

Figure 15-4. Break Address Register High (BRKH)

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

Figure 15-5. Break Address Register Low (BRKL)

15.2.2.3 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.

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

= Unimplemented

Figure 15-6. Break Auxiliary Register (BRKAR)

BDCOP — Break Disable COP Bit

This read/write bit disables the COP during a break interrupt. Reset clears the BDCOP bit.

1 = COP disabled during break interrupt
0 = COP enabled during break interrupt

15.2.2.4 Break Status Register

The break status register (BSR) contains a flag to indicate that a break caused an exit from wait mode. This register is only used in emulation mode.

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	R	R	R	R	R	R	SBSW	R
Write:							Note(1)	
Reset:							0	

R = Reserved

1. Writing a 0 clears SBSW.

Figure 15-7. Break Status Register (BSR)

SBSW — SIM Break Stop/Wait

SBSW can be read within the break state SWI routine. The user can modify the return address on the stack by subtracting one from it.

1 = Wait mode was exited by break interrupt
0 = Wait mode was not exited by break interrupt

15.2.2.5 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.

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

R = Reserved

Figure 15-8. Break Flag Control Register (BFCR)

Table 15-7. READSP (Read Stack Pointer) Command

Description	Reads stack pointer
Operand	None
Data Returned	Returns incremented stack pointer value (SP + 1) in high-byte:low-byte order
Opcode	$0C

Command Sequence

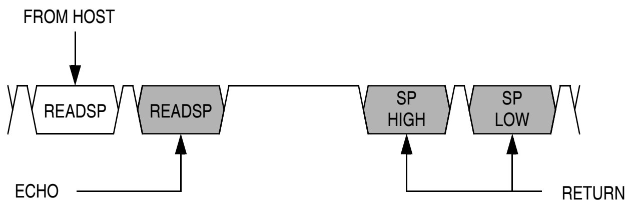
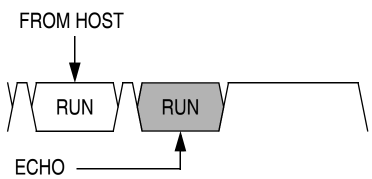


Table 15-8. RUN (Run User Program) Command

Description	Executes PULH and RTI instructions
Operand	None
Data Returned	None
Opcode	$28

Command Sequence



The MCU executes the SWI and PSHH instructions when it enters monitor mode. The RUN command tells the MCU to execute the PULH and RTI instructions. Before sending the RUN command, the host can modify the stacked CPU registers to prepare to run the host program. The READSP command returns the incremented stack pointer value, SP + 1. The high and low bytes of the program counter are at addresses SP + 5 and SP + 6.

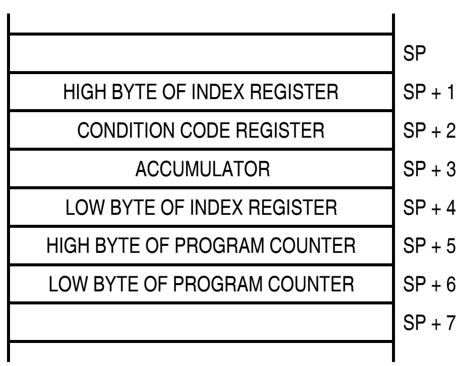


Figure 15-17. Stack Pointer at Monitor Mode Entry

16.5 5-V DC Electrical Characteristics

Characteristic[1]	Symbol	Min	Typ[2]	Max	Unit
Output high voltage I_{Load} = –2.0 mA, all I/O pins I_{Load} = –10.0 mA, all I/O pins I_{Load} = –15.0 mA, PTA0, PTA1, PTA3–PTA5 only	V_{OH}	V_{DD}–0.4 V_{DD}–1.5 V_{DD}–0.8	— — —	— — —	V
Maximum combined I_{OH} (all I/O pins)	I_{OHT}	—	—	50	mA
Output low voltage I_{Load} = 1.6 mA, all I/O pins I_{Load} = 10.0 mA, all I/O pins I_{Load} = 15.0 mA, PTA0, PTA1, PTA3–PTA5 only	V_{OL}	— — —	— — —	0.4 1.5 0.8	V
Maximum combined I_{OL} (all I/O pins)	I_{OHL}	—	—	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[3]	V_{HYS}	0.06 x V_{DD}	—	—	V
DC injection current, all ports[4]	I_{INJ}	–2	—	+2	mA
Total dc current injection (sum of all I/O)[4]	I_{INJTOT}	–25	—	+25	mA
Ports Hi-Z leakage current	I_{IL}	–1	±0.1	+1	µA
Capacitance Ports (as input)[3]	C_{IN}	—	—	8	pF
POR rearm voltage	V_{POR}	750	—	—	mV
POR rise time ramp rate[3][5]	R_{POR}	0.035	—	—	V/ms
Monitor mode entry voltage [3]	V_{TST}	V_{DD} + 2.5	—	9.1	V
Pullup resistors[6] PTA0–PTA5, PTB0–PTB7	R_{PU}	16	26	36	kΩ
Pulldown resistors[7] PTA0–PTA5	R_{PD}	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. Values are based on characterization results, not tested in production.
4. Guaranteed by design, not tested in production.
5. 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.
6. R_{PU} is measured at V_{DD} = 5.0 V.
7. R_{PD} is measured at V_{DD} = 5.0 V, Pulldown resistors only available when KBIx is enabled with KBIxPOL =1.

16.10 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[2]	t_{RL}	200	—	ns
$\overline{IRQ}$ interrupt pulse width low (edge-triggered)[2]	t_{ILIH}	200	—	ns
$\overline{IRQ}$ interrupt pulse period[2]	t_{ILIL}	Note[3]	—	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. Values are based on characterization results, not tested in production.
3. The minimum period is the number of cycles it takes to execute the interrupt service routine plus 1 t_{cyc}.

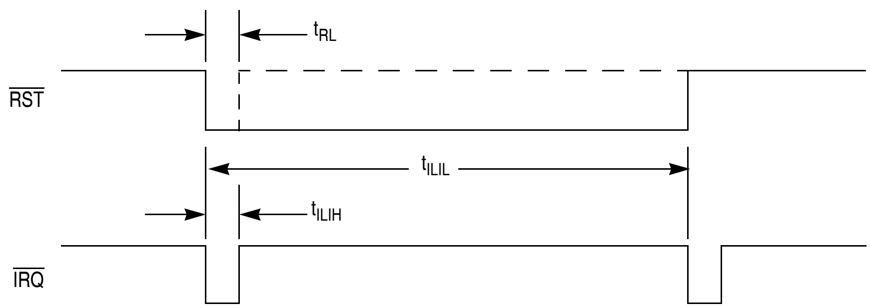


Figure 16-6. $\overline{RST}$ and $\overline{IRQ}$ Timing

16.14 Timer Interface Module Characteristics

Characteristic	Symbol	Min	Max	Unit
Timer input capture pulse width(1)	t_{TH}, t_{TL}	2	—	t_{cyc}
Timer input capture period	t_{TLTL}	Note(2)	—	t_{cyc}
Timer input clock pulse width(1)	t_{TCL}, t_{TCH}	t_{cyc} + 5	—	ns

1. Values are based on characterization results, not tested in production.
2. The minimum period is the number of cycles it takes to execute the interrupt service routine plus 1 t_{cyc}.

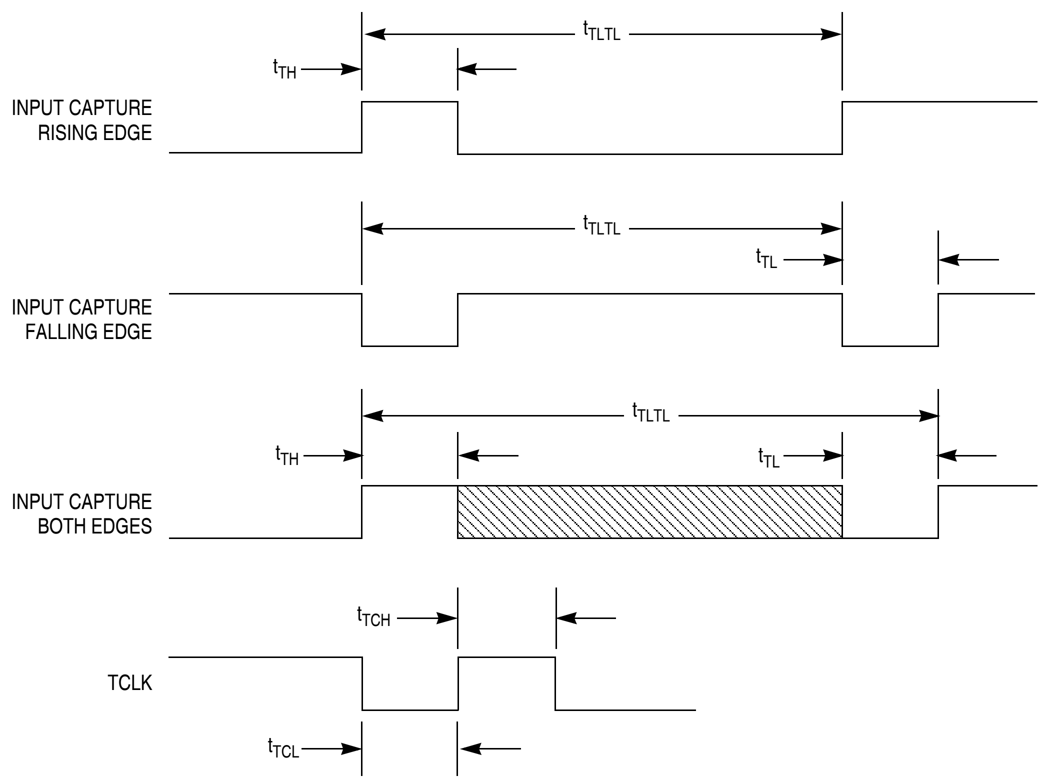


Figure 16-11. Timer Input Timing

Case 1452 page 2 of 4