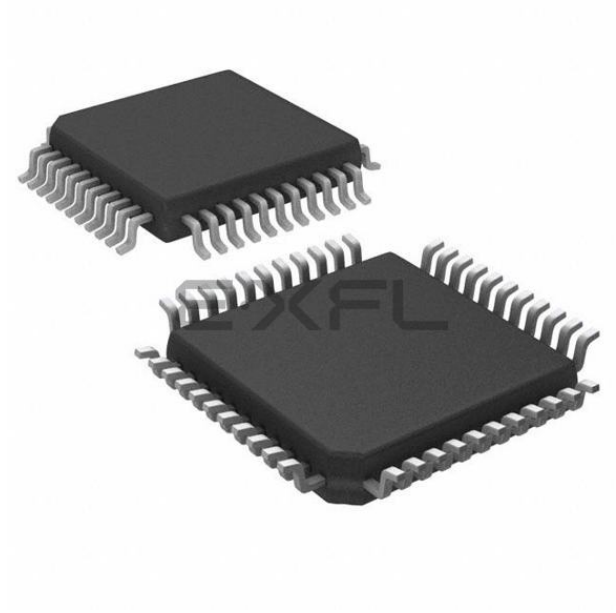


Welcome to E-XFL.COM

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	3MHz
Connectivity	USB
Peripherals	LVD, POR, PWM
Number of I/O	37
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	44-QFP
Supplier Device Package	44-QFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc908jb8fbe

8.8.1 Break Status Register 113
8.8.2 Reset Status Register 114
8.8.3 Break Flag Control Register 116

Section 9. Universal Serial Bus Module (USB)

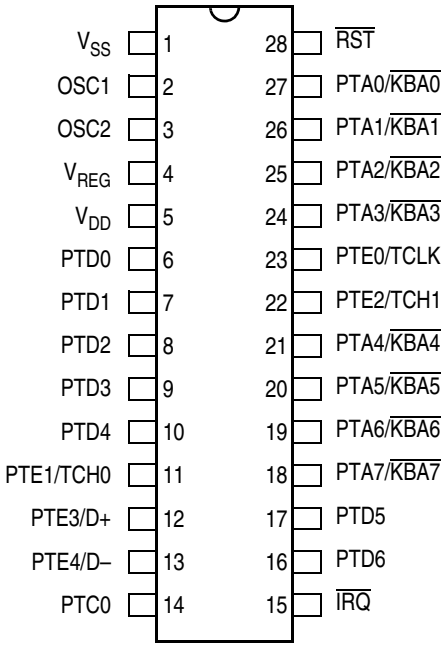
9.1 Contents 117
9.2 Introduction 118
9.3 Features 119
9.4 Pin Name Conventions 120
9.5 Functional Description 124
9.5.1 USB Protocol 125
9.5.1.1 Sync Pattern 126
9.5.1.2 Packet Identifier Field 127
9.5.1.3 Address Field (ADDR) 128
9.5.1.4 Endpoint Field (ENDP) 128
9.5.1.5 Cyclic Redundancy Check (CRC) 128
9.5.1.6 End-of-Packet (EOP) 128
9.5.2 Reset Signaling 129
9.5.3 Suspend 130
9.5.4 Resume After Suspend 131
9.5.4.1 Host Initiated Resume 131
9.5.4.2 USB Reset Signalling 131
9.5.4.3 Remote Wakeup 131
9.5.5 Low-Speed Device 132
9.6 Clock Requirements 132
9.7 Hardware Description 133
9.7.1 Voltage Regulator 133
9.7.2 USB Transceiver 133
9.7.2.1 Output Driver Characteristics 134
9.7.2.2 Low Speed (1.5 Mbps) Driver Characteristics 134
9.7.2.3 Receiver Data Jitter 135
9.7.2.4 Data Source Jitter 135
9.7.2.5 Data Signal Rise and Fall Time 136
9.7.3 USB Control Logic 137

1.3 Features

Features of the MC68HC908JB8 include:

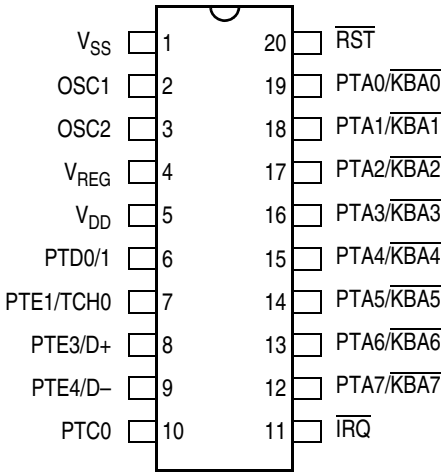
- High-performance M68HC08 architecture
- Fully upward-compatible object code with M6805, M146805, and M68HC05 Families
- 3-MHz internal bus frequency
- 8,192 bytes of on-chip FLASH memory
- 256 bytes of on-chip random-access memory (RAM)
- FLASH program memory security[1]
- On-chip programming firmware for use with host PC computer
- Up to 37 general-purpose 3.3V input/output (I/O) pins, including:
  - 13 or 10 shared-function I/O pins, depending on package
  - 24, 8, or 2 dedicated I/O pins, depending on package
  - 8 keyboard interrupts on port A, on all packages
  - 10mA sink capability for normal LED on 4 pins
  - 25mA sink capability for infrared LED on 2 pins
  - 10mA sink capability for PS/2 connection on 2 pins (with USB module disabled)
- 16-bit, 2-channel timer interface module (TIM) with selectable input capture, output compare, PWM capability on each channel, and external clock input option (TCLK)
- Full Universal Serial Bus Specification 1.1 low-speed functions:
  - 1.5 Mbps data rate
  - On-chip 3.3V regulator
  - Endpoint 0 with 8-byte transmit buffer and 8-byte receive buffer
  - Endpoint 1 with 8-byte transmit buffer
  - Endpoint 2 with 8-byte transmit buffer and 8-byte receive buffer

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



Pins not available on 28-pin package:		
PTB0		
PTB1	PTC1	
PTB2	PTC2	
PTB3	PTC3	
PTB4	PTC4	
PTB5	PTC5	
PTB6	PTC6	
PTB7	PTC7	PTD7
Internal pads are unconnected.		

Figure 1-3. 28-Pin SOIC Pin Assignments



PTD0/1 pin: PTD0 and PTD1 internal pads are bonded together to PTD0/1 pin.

Pins not available on 20-pin package:			
PTB0			PTE0/TCLK
PTB1	PTC1		
PTB2	PTC2	PTD2	PTE2/TCH1
PTB3	PTC3	PTD3	
PTB4	PTC4	PTD4	
PTB5	PTC5	PTD5	
PTB6	PTC6	PTD6	
PTB7	PTC7	PTD7	
Internal pads are unconnected.			

Figure 1-4. 20-Pin PDIP and SOIC Pin Assignments

NOTE: *In 20-pin package, the PTD0 and PTD1 internal pads are bonded together to PTD0/1 pin.*

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$003C	USB Control Register 1 (UCR1)	Read: Write:	T1SEQ	STALL1	TX1E	FRESUM	TP1SIZ3	TP1SIZ2	TP1SIZ1	TP1SIZ0
		Reset:	0	0	0	0	0	0	0	0
$003D	USB Status Register 0 (USR0)	Read:	R0SEQ	SETUP	0	0	RP0SIZ3	RP0SIZ2	RP0SIZ1	RP0SIZ0
		Write:								
		Reset:	Unaffected by reset							
$003E	USB Status Register 1 (USR1)	Read:	R2SEQ	TXACK	TXNAK	TXSTL	RP2SIZ3	RP2SIZ2	RP2SIZ1	RP2SIZ0
		Write:								
		Reset:	U	0	0	0	U	U	U	U
$003F	Unimplemented	Read: Write:								
$FE00	Break Status Register (BSR)	Read: Write:	R	R	R	R	R	R	SBSW See note	R
		Reset:							0	
$FE01	Reset Status Register (RSR)	Read:	POR	PIN	COP	ILOP	ILAD	USB	LVI	0
		Write:								
		POR:	1	0	0	0	0	0	0	0
$FE02	Reserved	Read: Write:	R	R	R	R	R	R	R	R
$FE03	Break Flag Control Register (BFCR)	Read: Write:	BCFE	R	R	R	R	R	R	R
		Reset:	0							
$FE04	Interrupt Status Register 1 (INT1)	Read:	IF6	IF5	IF4	IF3	IF2	IF1	0	0
		Write:	R	R	R	R	R	R	R	R
		Reset:	0	0	0	0	0	0	0	0
$FE05	Reserved	Read: Write:	R	R	R	R	R	R	R	R

Note: Writing a logic 0 clears SBSW.

= Unimplemented R = Reserved U = Unaffected by reset

Figure 2-2. Control, Status, and Data Registers (Sheet 7 of 8)

Section 5. Configuration Register (CONFIG)

5.1 Contents

5.2 Introduction 65
5.3 Functional Description 66

5.2 Introduction

This section describes the configuration register (CONFIG). This write-once-after-reset register controls the following options:

- USB reset
- Low voltage inhibit
- Stop mode recovery time (2048 or 4096 OSCXCLK cycles)
- COP timeout period ($2^{18} – 2^4$ or $2^{13} – 2^4$ OSCXCLK cycles)
- STOP instruction
- Computer operating properly module (COP)

8.3.1 Bus Timing

In user mode, the internal bus frequency is the oscillator frequency divided by two.

8.3.2 Clock Startup from POR or LVI Reset

When the power-on reset (POR) module or the low-voltage inhibit module generates a reset, the clocks to the CPU and peripherals are inactive and held in an inactive phase until after the 4096 OSCXCLK cycle POR timeout has completed. The $\overline{\text{RST}}$ pin is driven low by the SIM during this entire period. The IBUS clocks start upon completion of the timeout.

8.3.3 Clocks in Stop Mode and Wait Mode

Upon exit from stop mode by an interrupt, break, or reset, the SIM allows OSCXCLK to clock the SIM counter. The CPU and peripheral clocks do not become active until after the stop delay timeout. This timeout is selectable as 4096 or 2048 OSCXCLK cycles. (See **8.7.2 Stop Mode**.)

In wait mode, the CPU clocks are inactive. The SIM also produces two sets of clocks for other modules. Refer to the wait mode subsection of each module to see if the module is active or inactive in wait mode. Some modules can be programmed to be active in wait mode.

8.4 Reset and System Initialization

The MCU has these reset sources:

- Power-on reset module (POR)
- External reset pin ($\overline{\text{RST}}$)
- Computer operating properly module (COP)
- Illegal opcode
- Illegal address
- Universal serial bus module (USB)
- Low-voltage inhibit module (LVI)

8.6.1.1 Hardware Interrupts

A hardware interrupt does not stop the current instruction. Processing of a hardware interrupt begins after completion of the current instruction. When the current instruction is complete, the SIM checks all pending hardware interrupts. If interrupts are not masked (I bit clear in the condition code register) and if the corresponding interrupt enable bit is set, the SIM proceeds with interrupt processing; otherwise, the next instruction is fetched and executed.

If more than one interrupt is pending at the end of an instruction execution, the highest priority interrupt is serviced first. **Figure 8-11** demonstrates what happens when two interrupts are pending. If an interrupt is pending upon exit from the original interrupt service routine, the pending interrupt is serviced before the LDA instruction is executed.

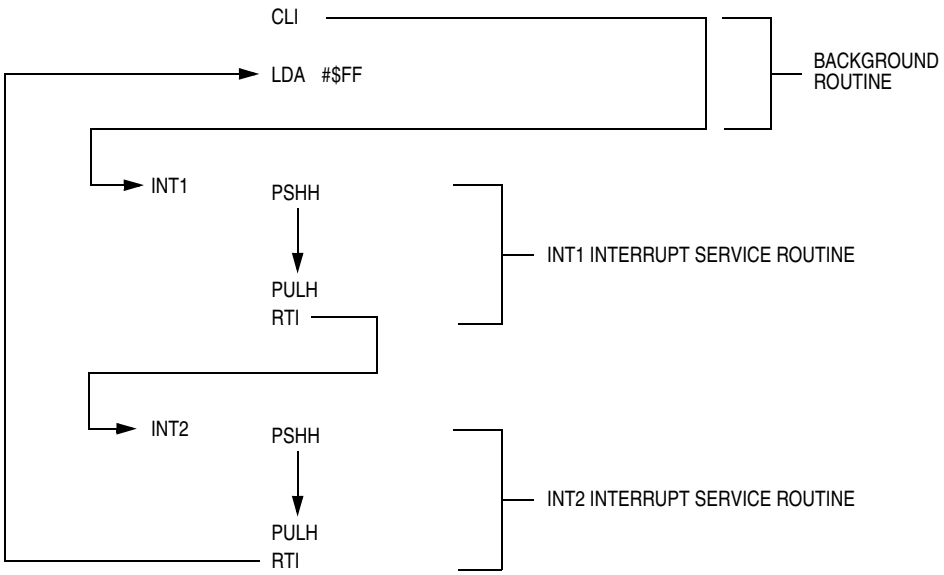


Figure 8-11. Interrupt Recognition Example

The LDA opcode is prefetched by both the INT1 and INT2 RTI instructions. However, in the case of the INT1 RTI prefetch, this is a redundant operation.

NOTE: *To maintain compatibility with the M6805 Family, the H register is not pushed on the stack during interrupt entry. If the interrupt service routine modifies the H register or uses the indexed addressing mode, software should save the H register and then restore it prior to exiting the routine.*

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0022	USB Endpoint 0 Data Register 2 (UE0D2)	Read:	UE0R27	UE0R26	UE0R25	UE0R24	UE0R23	UE0R22	UE0R21	UE0R20
		Write:	UE0T27	UE0T26	UE0T25	UE0T24	UE0T23	UE0T22	UE0T21	UE0T20
		Reset:	Unaffected by reset							
$0023	USB Endpoint 0 Data Register 3 (UE0D3)	Read:	UE0R37	UE0R36	UE0R35	UE0R34	UE0R33	UE0R32	UE0R31	UE0R30
		Write:	UE0T37	UE0T36	UE0T35	UE0T34	UE0T33	UE0T32	UE0T31	UE0T30
		Reset:	Unaffected by reset							
$0024	USB Endpoint 0 Data Register 4 (UE0D4)	Read:	UE0R47	UE0R46	UE0R45	UE0R44	UE0R43	UE0R42	UE0R41	UE0R40
		Write:	UE0T47	UE0T46	UE0T45	UE0T44	UE0T43	UE0T42	UE0T41	UE0T40
		Reset:	Unaffected by reset							
$0025	USB Endpoint 0 Data Register 5 (UE0D5)	Read:	UE0R57	UE0R56	UE0R55	UE0R54	UE0R53	UE0R52	UE0R51	UE0R50
		Write:	UE0T57	UE0T56	UE0T55	UE0T54	UE0T53	UE0T52	UE0T51	UE0T50
		Reset:	Unaffected by reset							
$0026	USB Endpoint 0 Data Register 6 (UE0D6)	Read:	UE0R67	UE0R66	UE0R65	UE0R64	UE0R63	UE0R62	UE0R61	UE0R60
		Write:	UE0T67	UE0T66	UE0T65	UE0T64	UE0T63	UE0T62	UE0T61	UE0T60
		Reset:	Unaffected by reset							
$0027	USB Endpoint 0 Data Register 7 (UE0D7)	Read:	UE0R77	UE0R76	UE0R75	UE0R74	UE0R73	UE0R72	UE0R71	UE0R70
		Write:	UE0T77	UE0T76	UE0T75	UE0T74	UE0T73	UE0T72	UE0T71	UE0T70
		Reset:	Unaffected by reset							
$0028	USB Endpoint 1 Data Register 0 (UE1D0)	Read:								
		Write:	UE1T07	UE1T06	UE1T05	UE1T04	UE1T03	UE1T02	UE1T01	UE1T00
		Reset:	Unaffected by reset							
$0029	USB Endpoint 1 Data Register 1 (UE1D1)	Read:								
		Write:	UE1T17	UE1T16	UE1T15	UE1T14	UE1T13	UE1T12	UE1T11	UE1T10
		Reset:	Unaffected by reset							
$002A	USB Endpoint 1 Data Register 2 (UE1D2)	Read:								
		Write:	UE1T27	UE1T26	UE1T25	UE1T24	UE1T23	UE1T22	UE1T21	UE1T20
		Reset:	Unaffected by reset							
$002B	USB Endpoint 1 Data Register 3 (UE1D3)	Read:								
		Write:	UE1T37	UE1T36	UE1T35	UE1T34	UE1T33	UE1T32	UE1T31	UE1T30
		Reset:	Unaffected by reset							

= Unimplemented U = Unaffected by reset

Figure 9-1. USB I/O Register Summary (Sheet 2 of 4)

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0036	USB Endpoint 2 Data Register 6 (UE2D6)	Read:	UE2R67	UE2R66	UE2R65	UE2R64	UE2R63	UE2R62	UE2R61	UE2R60
		Write:	UE2T67	UE2T66	UE2T65	UE2T64	UE2T63	UE2T62	UE2T61	UE2T60
		Reset:	Unaffected by reset							
$0037	USB Endpoint 2 Data Register 7 (UE2D7)	Read:	UE2R77	UE2R76	UE2R75	UE2R74	UE2R73	UE2R72	UE2R71	UE2R70
		Write:	UE2T77	UE2T76	UE2T75	UE2T74	UE2T73	UE2T72	UE2T71	UE2T70
		Reset:	Unaffected by reset							
$0038	USB Address Register (UADDR)	Read:	USBEN	UADD6	UADD5	UADD4	UADD3	UADD2	UADD1	UADD0
		Write:								
		Reset:	0*	0	0	0	0	0	0	0

* USBEN bit is reset by POR or LVI reset only.

$0039	USB Interrupt Register 0 (UIR0)	Read:	EOPIE	SUSPND	TXD2IE	RXD2IE	TXD1IE	0	TXD0IE	RXD0IE
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$003A	USB Interrupt Register 1 (UIR1)	Read:	EOPF	RSTF	TXD2F	RXD2F	TXD1F	RESUMF	TXD0F	RXD0F
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$003B	USB Control Register 0 (UCR0)	Read:	T0SEQ	0	TX0E	RX0E	TP0SIZ3	TP0SIZ2	TP0SIZ1	TP0SIZ0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$003C	USB Control Register 1 (UCR1)	Read:	T1SEQ	STALL1	TX1E	FRESUM	TP1SIZ3	TP1SIZ2	TP1SIZ1	TP1SIZ0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$003D	USB Status Register 0 (USR0)	Read:	R0SEQ	SETUP	0	0	RP0SIZ3	RP0SIZ2	RP0SIZ1	RP0SIZ0
		Write:								
		Reset:	Unaffected by reset							
$003E	USB Status Register 1 (USR1)	Read:	R2SEQ	TXACK	TXNAK	TXSTL	RP2SIZ3	RP2SIZ2	RP2SIZ1	RP2SIZ0
		Write:								
		Reset:	U	0	0	0	U	U	U	U

= Unimplemented U = Unaffected by reset

Figure 9-1. USB I/O Register Summary (Sheet 4 of 4)

To enable the next data packet transmission, TX2E also must be set. If the TXD2F bit is not cleared, a NAK handshake will be returned in the next IN transaction.

Reset clears this bit. Writing to TXD2F has no effect.

1 = Transmit on endpoint 2 has occurred
0 = Transmit on endpoint 2 has not occurred

RXD2F — Endpoint 2 Data Receive Flag

This read-only bit is set after the USB module has received a data packet and responded with an ACK handshake packet. Software must clear this flag by writing a logic 1 to the RXD2FR bit after all of the received data has been read. Software also must set the RX2E bit to 1 to enable the next data packet reception. If the RXD2F bit is not cleared, a NAK handshake will be returned in the next OUT transaction.

Reset clears this bit. Writing to RXD2F has no effect.

1 = Receive on endpoint 2 has occurred
0 = Receive on endpoint 2 has not occurred

TXD1F — Endpoint 1 Data Transmit Flag

This read-only bit is set after the data stored in the endpoint 1 transmit buffer has been sent and an ACK handshake packet from the host is received. Once the next set of data is ready in the transmit buffers, software must clear this flag by writing a logic 1 to the TXD1FR bit. To enable the next data packet transmission, TX1E also must be set. If the TXD1F bit is not cleared, a NAK handshake will be returned in the next IN transaction.

Reset clears this bit. Writing to TXD1F has no effect.

1 = Transmit on endpoint 1has occurred
0 = Transmit on endpoint 1has not occurred

RESUMF — Resume Flag

This read-only bit is set when USB bus activity is detected while the SUSPND bit is set. Software must clear this flag by writing a logic 1 to the RESUMFR bit. Reset clears this bit. Writing a logic 0 to RESUMF has no effect.

1 = USB bus activity has been detected
0 = No USB bus activity has been detected

10.3 Features

Features of the monitor ROM include the following:

- Normal user-mode pin functionality
- One pin dedicated to serial communication between monitor ROM and host computer
- Standard mark/space non-return-to-zero (NRZ) communication with host computer
- Execution of code in RAM or FLASH
- FLASH memory security feature[1]
- FLASH memory programming interface
- 976 bytes monitor ROM code size
- Monitor mode entry without high voltage, V_{DD} + V_{HI}, if reset vector is blank ($FFFE and $FFFF contain $FF)
- Standard monitor mode entry if high voltage, V_{DD} + V_{HI}, is applied to $\overline{\text{IRQ}}$

10.4 Functional Description

The monitor ROM receives and executes commands from a host computer. **Figure 10-1** shows a example circuit used to enter monitor mode and communicate with a host computer via a standard RS-232 interface.

Simple monitor commands can access any memory address. In monitor mode, the MCU can execute host-computer code in RAM while most MCU pins retain normal operating mode functions. All communication between the host computer and the MCU is through the PTA0 pin. A level-shifting and multiplexing interface is required between PTA0 and the host computer. PTA0 is used in a wired-OR configuration and requires a pull-up resistor.

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

11.10.5 TIM Channel Registers

These read/write registers contain the captured TIM counter value of the input capture function or the output compare value of the output compare function. The state of the TIM channel registers after reset is unknown.

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 Address: $0011

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

TCH0L Address: $0012

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

TCH1H Address: $0014

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

TCH1L Address: $0015

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

Figure 11-9. TIM Channel Registers (TCH0H/L:TCH1H/L)

0 = $\overline{\text{IRQ}}$ interrupt requests on falling edges only

13.9 IRQ Option Control Register

The IRQ option control register controls and monitors the external interrupt function available on the PTE4 pin. It also disables/enables the pullup resistor on the $\overline{\text{IRQ}}$ pin.

- Controls pullup option on $\overline{\text{IRQ}}$ pin
- Enables PTE4 pin for external interrupts to IRQ
- Shows the state of the PTE4 interrupt flag

Address: $001C

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	0	PTE4IF	PTE4IE	IRQPD
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 13-4. IRQ Option Control Register (IOCR)

PTE4IF — PTE4 Interrupt Flag

This read-only status bit is high when a falling edge on PTE4 pin is detected. PTE4IF bit clears when the IOCR is read.

1 = Falling edge on PTE4 is detected and PTE4IE is set
0 = Falling edge on PTE4 is not detected or PTE4IE is clear

PTE4IE — PTE4 Interrupt Enable

This read/write bit enables or disables the interrupt function on the PTE4 pin to trigger the IRQ interrupt. Setting the PTE4IE bit and clearing the USBEN bit in the USB address register configure the PTE4 pin for interrupt function to the IRQ interrupt. Setting PTE4IE also enables the internal pullup on PTE4 pin.

1 = PTE4 interrupt enabled; triggers IRQ interrupt
0 = PTE4 interrupt disabled

IRQPD — $\overline{\text{IRQ}}$ Pullup Disable

- Return of all enabled keyboard interrupt pins to logic 1 — As long as any enabled keyboard interrupt pin is at logic 0, the keyboard interrupt remains set.

The vector fetch or software clear and the return of all enabled keyboard interrupt pins to logic 1 may occur in any order.

If the MODEK bit is clear, the keyboard interrupt pin is falling-edge-sensitive only. With MODEK clear, a vector fetch or software clear immediately clears the keyboard interrupt request.

Reset clears the keyboard interrupt request and the MODEK bit, clearing the interrupt request even if a keyboard interrupt pin stays at logic 0.

The keyboard flag bit (KEYF) in the keyboard status and control register can be used to see if a pending interrupt exists. The KEYF bit is not affected by the keyboard interrupt mask bit (IMASKK) which makes it useful in applications where polling is preferred.

To determine the logic level on a keyboard interrupt pin, use the data direction register to configure the pin as an input and read the data register.

NOTE: *Setting a keyboard interrupt enable bit (KBIEx) forces the corresponding keyboard interrupt pin to be an input, overriding the data direction register. However, the data direction register bit must be a logic 0 for software to read the pin.*

14.6 Keyboard Initialization

When a keyboard interrupt pin is enabled, it takes time for the pullup device to reach a logic 1. Therefore, a false interrupt can occur as soon as the pin is enabled.

To prevent a false interrupt on keyboard initialization:

1. Mask keyboard interrupts by setting the IMASKK bit in the keyboard status and control register.
2. Enable the KBI pins by setting the appropriate KBIEx bits in the keyboard interrupt enable register.

3. Write to the ACKK bit in the keyboard status and control register to clear any false interrupts.
4. Clear the IMASKK bit.

An interrupt signal on an edge-triggered pin can be acknowledged immediately after enabling the pin. An interrupt signal on an edge- and level-triggered interrupt pin must be acknowledged after a delay that depends on the external load.

Another way to avoid a false interrupt:

1. Configure the keyboard pins as outputs by setting the appropriate DDRA bits in data direction register A.
2. Write logic 1s to the appropriate port A data register bits.
3. Enable the KBI pins by setting the appropriate KBIEx bits in the keyboard interrupt enable register.

14.7 Low-Power Modes

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

14.7.1 Wait Mode

The keyboard module remains active in wait mode. Clearing the IMASKK bit in the keyboard status and control register enables keyboard interrupt requests to bring the MCU out of wait mode.

14.7.2 Stop Mode

The keyboard module remains active in stop mode. Clearing the IMASKK bit in the keyboard status and control register enables keyboard interrupt requests to bring the MCU out of stop mode.

15.4.4 Power-On Reset

The power-on reset (POR) circuit in the SIM clears the COP prescaler 4096 OSCXCLK cycles after power-up.

15.4.5 Internal Reset

An internal reset clears the SIM counter and the COP counter.

15.4.6 Reset Vector Fetch

A reset vector fetch occurs when the vector address appears on the data bus. A reset vector fetch clears the COP prescaler.

15.4.7 COPD (COP Disable)

The COPD signal reflects the state of the COP disable bit (COPD) in the configuration register (CONFIG).

15.4.8 COPRS (COP Rate Select)

The COPRS signal reflects the state of the COP rate select bit (COPRS) in the configuration register (CONFIG).

Address: $001F

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	URSTD	LVID	SSREC	COPRS	STOP	COPD
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 15-2. Configuration Register (CONFIG)

17.5.2 Stop Mode

A break interrupt causes exit from stop mode and sets the SBSW bit in the break status register. See **8.8 SIM Registers**.

17.6 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)

17.6.1 Break Status and Control Register

The break status and control register contains break module enable and status bits.

Address: $FE0E

	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 17-3. 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 logic 0 to bit 7. Reset clears the BRKE bit.

1 = Breaks enabled on 16-bit address match
0 = Breaks disabled

A.7.1 DC Electrical Characteristics

Characteristic[1]	Symbol	Min	Typ[2]	Max	Unit
Regulator output voltage	V_{REG}	3.0	3.3	3.6	V
Output high voltage (I_{Load} = −2.0 mA) PTA0–PTA7, PTB0–PTB7, PTC0–PTC7, PTE0–PTE2	V_{OH}	V_{REG}−0.8	—	—	V
Output low voltage (I_{Load} = 1.6 mA) All I/O pins (I_{Load} = 25 mA) PTD0–PTD1 in ILDD mode (I_{Load} = 10 mA) PTE3–PTE4 with USB disabled	V_{OL}	— — —	— — —	0.4 0.5 0.4	V
Input high voltage All ports, OSC1 $\overline{IRQ}$, $\overline{RST}$	V_{IH}	$0.7 \times V_{REG}$ $0.7 \times V_{DD}$	— —	V_{REG} V_{DD}	V
Input low voltage All ports, OSC1 $\overline{IRQ}$, $\overline{RST}$	V_{IL}	V_{SS} V_{SS}	— —	$0.3 \times V_{REG}$ $0.3 \times V_{DD}$	V
Output low current (V_{OL} = 2.0 V) PTD2–PTD5 in LDD mode	I_{OL}	17	22	27	mA
V_{DD} supply current, V_{DD} = 5.25V, f_{OP} = 3MHz Run, with low speed USB[3] Run, with USB suspended[3] Wait, with low speed USB[4] Wait, with USB suspended[4] Stop[5] 0 °C to 70°C	I_{DD}	 — — — — —	 5.0 4.5 3.0 2.5 30	 7.5 6.5 5.0 4.0 100	 mA mA mA mA µA
I/O ports Hi-Z leakage current	I_{IL}	—	—	± 10	µA
Input current	I_{IN}	—	—	± 1	µA
Capacitance Ports (as input or output)	C_{Out} C_{In}	— —	— —	12 8	pF
POR re-arm voltage[6]	V_{POR}	0	—	100	mV
POR rise-time ramp rate[7]	R_{POR}	0.035	—	—	V/ms
Monitor mode entry voltage	$V_{DD}+V_{HI}$	$1.4 \times V_{DD}$		$2 \times V_{DD}$	V
Pullup resistors Port A, port B, port C, PTE0–PTE2, $\overline{RST}$, $\overline{IRQ}$ PTE3–PTE4 (with USB module disabled) D– (with USB module enabled)	R_{PU}	25 4 1.2	40 5 1.5	55 6 2.0	kΩ
LVI reset	V_{LVR}	2.4	2.7	3.0	V