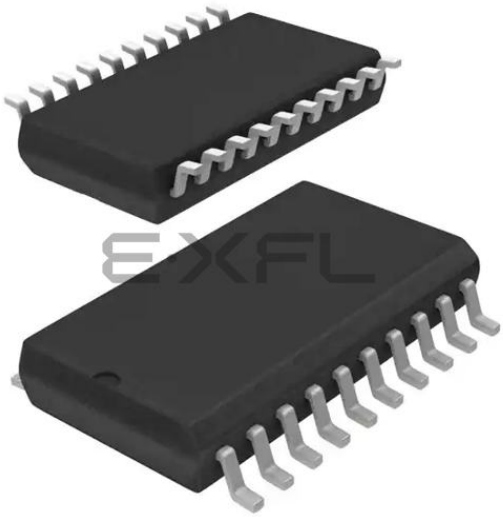


Welcome to E-XFL.COM

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	LVR, POR, PWM
Number of I/O	18
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 7x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc68hc908lb8vdwe

Chapter 4
Op Amp/Comparator Module

Chapter 5
Configuration Register (CONFIG)

Chapter 6
Computer Operating Properly (COP) Module

Chapter 2
Memory

2.1 Introduction

The CPU08 can address 64 Kbytes of memory space. The memory map, shown in Figure 2-1, includes:

- System registers
- 8192 bytes of user FLASH memory
- 128 bytes of random-access memory (RAM)
- 674 bytes of FLASH programming routines read-only memory (ROM)
- 34 bytes of user-defined vectors

2.2 Unimplemented Memory Locations

Accessing an unimplemented location can cause an illegal address reset. In the memory map (Figure 2-1) and in register figures in this document, unimplemented locations are shaded.

2.3 Reserved Memory Locations

Accessing a reserved location can have unpredictable effects on microcontroller (MCU) operation. In the Figure 2-1 and in register figures in this document, reserved locations are marked with the word Reserved or with the letter R.

2.4 Register Section

Most of the control, status, and data registers are in the zero page area of $0000–$0058. Additional I/O registers have these addresses:

- $FE00; break status register, BSR
- $FE01; SIM reset status register, SRSR
- $FE02; break auxiliary register, BRKAR
- $FE03; break flag control register, BFCR
- $FE04; interrupt status register 1, INT1
- $FE05; interrupt status register 2, INT2
- $FE06; reserved
- $FE07; reserved
- $FE08; FLASH control register, FLCR
- $FE09; break address register high, BRKH
- $FE0A; break address register low, BRKL
- $FE0B; break status and control register, BRKSCR
- $FE0C; LVI status register, LVISR
- $FF7E; FLASH block protect register, FLBPR

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$004C	PWM 1 Value Register Low (PVAL1L) See page 155.	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
$004D	PWM Disable Mapping Write Once Register (DISMAP) See page 158.	Read:	0	0	0	0	0	0	MAP1	MAP0
		Write:								
		Reset:	0	0	0	0	0	0	1	1
$004E ↓ $004F	Unimplemented									
$0050	Reserved		Reserved							
$0051	HRP Control Register (HRPCTRL) See page 105.	Read:		SHTLVL	HRPOE	SHTIF	SHTIE	SHTEN	HRP-MODE(1)	HRPEN
		Write:								
		Reset		0	0	0	0	0	0	0
$0052	HRP Duty Cycle Register High (HRPDCH) See page 107.	Read:	DC10	DC9	DC8	DC7	DC6	DC5	DC4	DC3
		Write:								
		Reset	0	0	0	0	0	0	0	0
$0053	HRP Duty Cycle Register Low (HRPDCL) See page 107.	Read:	DC2	DC1	DC0	STEP4	STEP3	STEP3	STEP1	STEP0
		Write:								
		Reset	0	0	0	0	0	0	0	0
$0054	HRP Period Register High (HRPPERH) See page 107.	Read:	P10	P9	P8	P7	P6	P5	P4	P3
		Write:								
		Reset	0	0	0	0	0	0	0	0
$0055	HRP Period Register Low (HRPPERL) See page 107.	Read:	P2	P1	P0	STEP4	STEP3	STEP2	STEP1	STEP0
		Write:								
		Reset	0	0	0	0	0	0	0	0
$0056	HRP Dead Time Register (HRP_DT) See page 108.	Read:	DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0
		Write:								
		Reset	0	0	0	0	1	0	0	0
$0057	HRP Timebase Register High (HRPTBH) See page 108.	Read:	TB15	TB14	TB13	TB12	TB11	TB10	TB9	TB8
		Write:								
		Reset	0	0	0	0	0	0	0	0

1. When HRPMODE bit = 0, STEP[4:0] are mapped into the HRPPERL register — when HRPMODE = 1, STEP[4:0] are mapped into the HRPDCL register.

= Unimplemented R = Reserved

Bold = Buffered U = Unaffected

Figure 2-2. Control, Status, and Data Registers (Sheet 6 of 8)

the input signal. The ADC overrides the port I/O logic by forcing that pin as input to the ADC. The remaining ADC channels/port pins are controlled by the port I/O logic and can be used as general-purpose I/O. Writes to the port register or data direction register (DDR) will not have any effect on the port pin that is selected by the ADC. Read of a port pin in use by the ADC will return a logic 0. If the DDR bit is at 1, the value in the port data latch is read.

3.3.2 Voltage Conversion

When the input voltage to the ADC equals V_{REFH}, the ADC converts the signal to $FF (full scale). If the input voltage equals V_{REFL}, the ADC converts it to $00. Input voltages between V_{REFH} and V_{REFL} are a straight-line linear conversion.

V_{REFH} and V_{REFL} are internally connected to V_{DD} and V_{SS} respectively.

3.3.3 Conversion Time

Conversion starts after a write to the ADC status and control register (ADSCR). One conversion will take between 16 and 17 ADC clock cycles. The ADIVx bit should be set to provide a 1-MHz ADC clock frequency.

$$\text{Conversion time} = \frac{\text{16 to 17 ADC cycles}}{\text{ADC frequency}}$$

$$\text{Number of bus cycles} = \text{conversion time} \times \text{bus frequency}$$

3.3.4 Conversion

In continuous conversion mode, the ADC data register will be filled with new data after each conversion. Data from the previous conversion will be overwritten whether that data has been read or not. Conversions will continue until the ADCO bit is cleared. The COCO bit is set after the first conversion and will stay set until the next write of the ADC status and control register or the next read of the ADC data register.

In single conversion mode, conversion begins with a write to the ADSCR. Only one conversion occurs between writes to the ADSCR.

3.3.5 Accuracy and Precision

The conversion process is monotonic and has no missing codes.

3.4 Monotonicity

The conversion process is monotonic and has no missing codes.

3.5 Interrupts

When the AIEN bit is set, the ADC module is capable of generating CPU interrupts after each ADC conversion. A CPU interrupt is generated if the COCO bit is at 0. The COCO bit is not used as a conversion complete flag when interrupts are enabled.

0 = Comparator mode selected

OACE — Op Amp/Comparator Enable Bit

Setting of the corresponding bit in the register enables the associated op amp/comparator and connects it to the op amp/comparator pins.

1 = Op amp/comparator is connected to pins and powered on
0 = Op amp/comparator is disconnected from pins and powered off

NOTE

Enabling the op amp/comparator prevents PTB[5:7] from being used as standard I/O. However, the PTB7 pin can be shared with AD6 and FAULT if the ADC and PWM modules are also enabled.

4.7 Application Information

We make the following assumptions during the design of the operational amplifier.

1. The signal amplified by the operational amplifier is sampled by the internal ADC.
2. Noise resulting from the operation of other circuitry within the MCU will appear at the output of the circuit due to the amplification set by user.

We recommend the following.

1. An external 500pF capacitor should be added between the output of the operational amplifier (PTB7) and VSS. This capacitor will act as a filter to internal bus noise caused by the operation of other digital circuitry in the MCU.
2. Care should be taken to ensure proper filtering at or around the operation bus speed in the amplification circuit, to prevent noise from being amplified along with the desired signal.
3. The maximum frequency of the signal to be amplified should be limited to 200kHz. This will ensure that the filtering element will not affect the gain of the desired signal.
4. Do not set the gain of the amplifier to less than 5 (except in the unity gain buffer).
5. Use the circuit component values suggested for the common amplfier configurations shown in the following figures (Figure 4-4, Figure 4-5, and Figure 4-6).

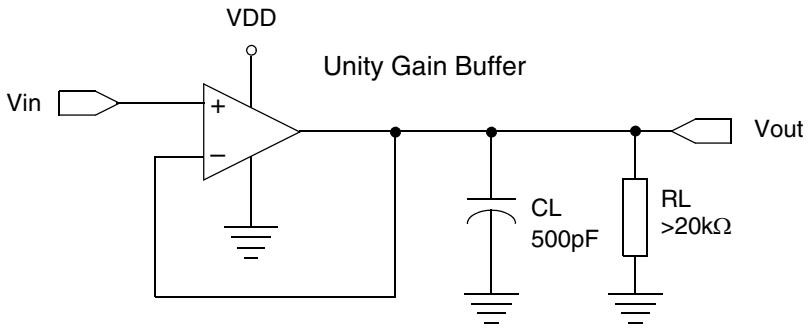


Figure 4-4. Suggested Application Circuit for Unity Gain Buffer

9.3 Functional Description

Writing to the KBIE6–KBIE0 bits in the keyboard interrupt enable register independently enables or disables each port A pin as a keyboard interrupt pin. Enabling a keyboard interrupt pin also enables its internal pullup device. A logic 0 applied to an enabled keyboard interrupt pin latches a keyboard interrupt request. A keyboard interrupt is latched when one or more keyboard pins goes low after all were high. The MODEK bit in the keyboard status and control register controls the triggering mode of the keyboard interrupt.

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

If the MODEK bit is set, the keyboard interrupt pins are both falling edge- and low-level sensitive, and both of the following actions must occur to clear a keyboard interrupt request:

- Vector fetch or software clear — A vector fetch generates an interrupt acknowledge signal to clear the interrupt request. Software may generate the interrupt acknowledge signal by writing a 1 to the ACKK bit in the keyboard status and control register (INTKBSCR). The ACKK bit is useful in applications that poll the keyboard interrupt pins and require software to clear the keyboard interrupt request. Writing to the ACKK bit prior to leaving an interrupt service routine can also prevent spurious interrupts due to noise. Setting ACKK does not affect subsequent transitions on the keyboard interrupt pins. A falling edge that occurs after writing to the ACKK bit latches another interrupt request. If the keyboard interrupt mask bit, IMASKK, is clear, the CPU loads the program counter with the vector address at locations $FFE0 and $FFE1.
- 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 0 for software to read the pin.

11.7 Keyboard Interrupt Module (KBI)

11.7.1 Wait Mode

The keyboard interrupt (KBI) 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.

11.7.2 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.

11.8 High Resolution PWM (HRP)

11.8.1 Wait Mode

The HRP remains active after the execution of a WAIT instruction. In wait mode the HRP registers are not accessible by the CPU. Any enabled CPU interrupt request from the HRP can bring the MCU out of wait mode. If HRP functions are not required during wait mode, reduce power consumption by stopping the HRP before executing the WAIT instruction.

11.8.2 Stop Mode

The HRP is inactive after the execution of a STOP instruction. The TOP and BOT outputs are both set to logic 0 and the HRPEN bit in the HRPCTRL register is set to 0 after execution of the STOP instruction. The STOP instruction does not affect other register conditions or the state of the HRP counters. When the MCU exits stop mode after an external interrupt, the HRP is inactive because the HRPEN bit is set to 0.

NOTE

The HRP shutdown pin remains active during Stop mode.

11.9 Low-Voltage Inhibit Module (LVI)

11.9.1 Wait Mode

If enabled, the low-voltage inhibit (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.

11.9.2 Stop Mode

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

13.3.3 XTAL Oscillator

The XTAL oscillator circuit is designed for use with an external crystal or ceramic resonator to provide an accurate clock source. In this configuration, the OSC2 pin is dedicated to the external crystal circuit. The OSC2EN bit in the port C pullup enable register has no effect when this clock mode is selected.

In its typical configuration, the XTAL oscillator is connected in a Pierce oscillator configuration, as shown in Figure 13-2. This figure shows only the logical representation of the internal components and may not represent actual circuitry. The oscillator configuration uses five components:

- Crystal, X_1
- Fixed capacitor, C_1
- Tuning capacitor, C_2 (can also be a fixed capacitor)
- Feedback resistor, R_B
- Series resistor, R_S (optional)

NOTE

The series resistor (R_S) is included in the diagram to follow strict Pierce oscillator guidelines and may not be required for all ranges of operation, especially with high frequency crystals. Refer to the crystal manufacturer's data for more information.

13.3.4 RC Oscillator

The RC oscillator circuit is designed for use with external R to provide a clock source with tolerance less than 25%. See Figure 13-3.

In its typical configuration, the RC oscillator requires two external components, one R and one C. In the MC68HC908LB8, the capacitor is internal to the chip. The R value should have a tolerance of 1% or less, to obtain a clock source with less than 25% tolerance. The oscillator configuration uses one component, R_{EXT}.

In this configuration, the OSC2 pin can be left in the reset state as PTC1. Or, the OSC2EN bit in the port C pullup enable register can be set to enable the OSC2 function on the pin without affecting the clocks.

Chapter 15
Pulse Width Modulator with Fault Input (PWM)

15.1 Introduction

This section describes the pulse-width modulator with fault input (PWM). The MC68HC908LB8 PWM module can generate two independent PWM signals. These PWM signals are edge-aligned. A block diagram of the PWM module is shown in Figure 15-2.

A 12-bit timer PWM counter is common to both channels. PWM resolution is one clock period for edge-aligned operation. The clock period is dependent on the internal operating frequency (BUSCLK) and a programmable prescaler.

The highest resolution for edge-aligned operation is 125 ns (BUSCLK = 8 MHz).

A summary of the PWM registers is shown in Figure 15-3.

15.2 Features

Features of the PWMMC include:

- Two independent PWM signals
- Edge-aligned PWM signals
- PWM signal polarity control
- Programmable fault protection

in Figure 15-4. Again, the timer modulus register is used to determine the maximum count. The PWM period will equal:

(timer modulus) x (PWM clock period)

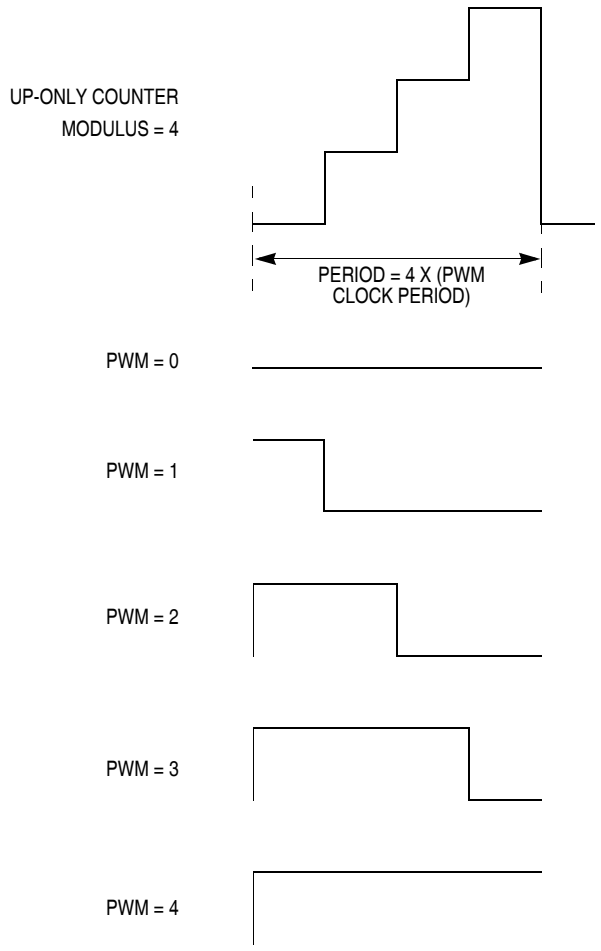


Figure 15-4. Edge-Aligned PWM (Positive Polarity)

15.3.2 Prescaler

To permit lower PWM frequencies, a prescaler is provided which will divide the PWM clock frequency by 1, 2, 4, or 8. Table 15-1 shows how setting the prescaler bits in PWM control register 2 affects the PWM clock frequency. This prescaler is buffered and will not be used by the PWM generator until the LDOK bit is set and a new PWM reload cycle begins.

	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

Bold = Buffered

Figure 15-19. PWMx Value Registers Low (PVALxL)

The 16-bit signed value stored in this register determines the duty cycle of the PWM. The duty cycle is defined as:

(PWM value/modulus) x 100

Writing a number less than or equal to 0 causes the PWM to be off for the entire PWM period. Writing a number greater than or equal to the 12-bit modulus causes the PWM to be on for the entire PWM period.

To avoid erroneous PWM pulses, this value is buffered and will not be used by the PWM generator until the LDOK bit has been set and the next PWM load cycle begins.

NOTE

When reading these registers, the value read is the buffer (not necessarily the value the PWM generator is currently using).

15.8.4 PWM Control Register 1

PWM control register 1 (PCTL1) controls PWM enabling/disabling, the location of the PWM Fault bit, the loading of new modulus, prescaler, PWM values, and the PWM correction method.

Address: $0040

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	FPOS	PWMINT	PWMF	0	0	LDOK	PWMEN
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 15-20. PWM Control Register 1 (PCTL1)

FPOS — Fault Pin Position Bit

This read/write bit allows the user to select the location of the Fault pin.

1 = Fault pin functionality is placed on PTB2
0 = Fault pin functionality is placed on PTB7

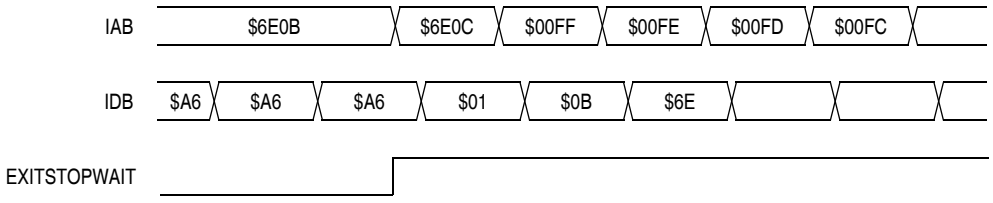
NOTE

Placing the Fault pin on PTB7 will not affect the ADC or the op amp/comparator connections. This is to allow the output of the op amp/comparator to be used as the input to the Fault pin and for this same signal to be simultaneously measured by the ADC.

PWMINT — PWM Interrupt Enable Bit

This read/write bit allows the user to enable and disable PWM CPU interrupts. If set, a CPU interrupt will be pending when the PWMF flag is set.

1 = Enable PWM CPU interrupts
0 = Disable PWM CPU interrupts



Note: EXITSTOPWAIT = $\overline{\text{RST}}$ pin or CPU interrupt

Figure 17-13. Wait Recovery from Interrupt

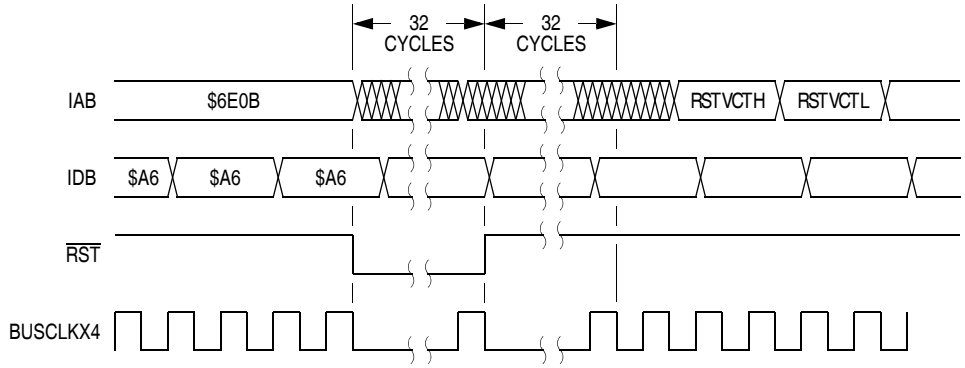


Figure 17-14. Wait Recovery from Internal Reset

17.6.2 Stop Mode

In stop mode, the SIM counter is reset and the system clocks are disabled. An interrupt request from a module can cause an exit from stop mode. Stacking for interrupts begins after the selected stop recovery time has elapsed. Reset also causes an exit from stop mode.

The SIM disables the clock generator module outputs (BUSCLKX2 and BUSCLKX4) in stop mode, stopping the CPU and peripherals. Stop recovery time is selectable using the SSREC bit in the mask option register (MOR). If SSREC is set, stop recovery is reduced from the normal delay of 4096 BUSCLKX4 cycles down to 32. This is ideal for applications using canned oscillators that do not require long startup times from stop mode.

NOTE

External crystal applications should use the full stop recovery time by clearing the SSREC bit.

The SIM counter is held in reset from the execution of the STOP instruction until the beginning of stop recovery. It is then used to time the recovery period. Figure 17-15 shows stop mode entry timing. Figure 17-16 shows stop mode recovery time from interrupt or break.

NOTE

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

TMODH:TMODL, control the modulo value of the TIM counter. Software can read the TIM counter value at any time without affecting the counting sequence.

The two TIM channels are programmable independently as input capture or output compare channels. If a channel is configured as input capture, then an internal pullup device may be enabled for that channel. .

Figure 18-3 summarizes the timer registers.

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0020	Timer Status and Control Register (T1SC) See page 195.	Read:	TOF	TOIE	TSTOP	0	0	PS2	PS1	PS0
		Write:	0			TRST				
		Reset:	0	0	1	0	0	0	0	0
$0021	Timer Counter Register High (T1CNTH) See page 196.	Read:	Bit 15	14	13	12	11	10	9	Bit 8
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0022	Timer Counter Register Low (T1CNTL) See page 196.	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0023	Timer Counter Modulo Register High (T1MODH) See page 197.	Read:	Bit 15	14	13	12	11	10	9	Bit 8
		Write:								
		Reset:	1	1	1	1	1	1	1	1
$0024	Timer Counter Modulo Register Low (T1MODL) See page 197.	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
		Reset:	1	1	1	1	1	1	1	1
$0025	Timer Channel 0 Status and Control Register (T1SC0) See page 198.	Read:	CH0F	CH0IE	MS0B	MS0A	ELS0B	ELS0A	TOV0	CH0MAX
		Write:	0							
		Reset:	0	0	0	0	0	0	0	0
$0026	Timer Channel 0 Register High (T1CH0H) See page 201.	Read:	Bit 15	14	13	12	11	10	9	Bit 8
		Write:								
		Reset:	Indeterminate after reset							
$0027	Timer Channel 0 Register Low (T1CH0L) See page 201.	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
		Reset:	Indeterminate after reset							
$0028	Timer Channel 1 Status and Control Register (T1SC1) See page 198.	Read:	CH1F	CH1IE	0	MS1A	ELS1B	ELS1A	TOV1	CH1MAX
		Write:	0							
		Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 18-3. TIM I/O Register Summary (Sheet 1 of 2)

- TIM overflow flag (TOF) — The TOF bit is set when the TIM counter reaches the modulo value programmed in the TIM counter modulo registers. The TIM overflow interrupt enable bit, TOIE, enables TIM overflow CPU interrupt requests. TOF and TOIE are in the TIM status and control register.
- TIM channel flags (CH1F:CH0F) — The CHxF bit is set when an input capture or output compare occurs on channel x. Channel x TIM CPU interrupt requests are controlled by the channel x interrupt enable bit, CHxIE. Channel x TIM CPU interrupt requests are enabled when CHxIE = 1. CHxF and CHxIE are in the TIM channel x status and control register.

18.5 Low-Power Modes

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

18.5.1 Wait Mode

The TIM remains active after the execution of a WAIT instruction. In wait mode, the TIM registers are not accessible by the CPU. Any enabled CPU interrupt request from the TIM can bring the MCU out of wait mode.

If TIM functions are not required during wait mode, reduce power consumption by stopping the TIM before executing the WAIT instruction.

18.5.2 Stop Mode

The TIM is inactive after the execution of a STOP instruction. The STOP instruction does not affect register conditions or the state of the TIM counter. TIM operation resumes when the MCU exits stop mode after an external interrupt.

18.6 TIM During Break Interrupts

A break interrupt stops the TIM counter.

The system integration module (SIM) controls whether status bits in other modules can be cleared during the break state. The BCFE bit in the SIM break flag control register (SBFCR) enables software to clear status bits during the break state. See 17.7.3 Break Flag Control Register.

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

To protect status bits during the break state, write a 0 to the BCFE bit. With BCFE at 0 (its default state), software can read and write I/O registers during the break state without affecting status bits. Some status bits have a 2-step read/write clearing procedure. If software does the first step on such a bit before the break, the bit cannot change during the break state as long as BCFE is at 0. After the break, doing the second step clears the status bit.

18.7 I/O Signals

Port B shares its pins with the TIM. Only TCH0 is available on a port pin. It is programmable independently as an input capture pin or an output compare pin. TCH0 can be configured as buffered output compare or buffered PWM pins.

Table 18-2. Mode, Edge, and Level Selection

MSxB:MSxA	ELSxB:ELSxA	Mode	Configuration
X0	00	Output preset	Pin under port control; initial output level high
X1	00		Pin under port control; initial output level low
00	01	Input capture	Capture on rising edge only
00	10		Capture on falling edge only
00	11		Capture on rising or falling edge
01	01	Output compare or PWM	Toggle output on compare
01	10		Clear output on compare
01	11		Set output on compare
1X	01	Buffered output compare or buffered PWM	Toggle output on compare
1X	10		Clear output on compare
1X	11		Set output on compare

TOVx — Toggle On Overflow Bit

When channel x is an output compare channel, this read/write bit controls the behavior of the channel x output when the TIM counter overflows. When channel x is an input capture channel, TOVx has no effect.

Reset clears the TOVx bit.

1 = Channel x pin toggles on TIM counter overflow.

0 = Channel x pin does not toggle 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 1, setting the CHxMAX bit forces the duty cycle of buffered and unbuffered PWM signals to 100%. As Figure 18-12 shows, 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.

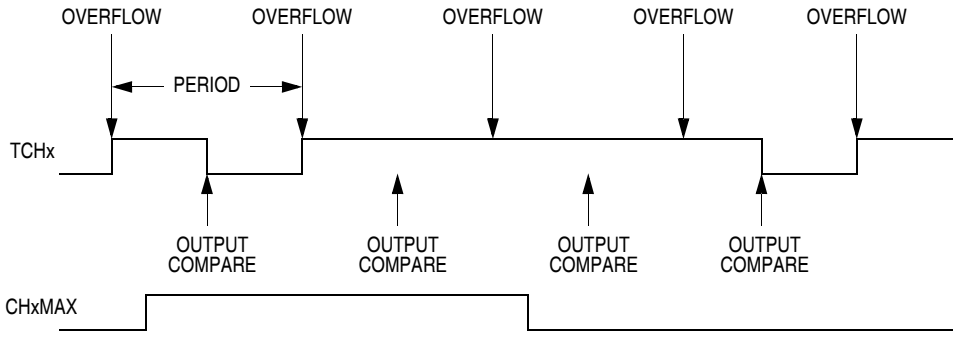


Figure 18-12. CHxMAX Latency

If entering monitor mode without high voltage on $\overline{IRQ}$ (above condition set 2 or 3, where applied voltage is V_{DD} or V_{SS}), then startup port pin requirements and conditions, (PTA1/PTA4) are not in effect. This is to reduce circuit requirements when performing in-circuit programming.

19.3.1.1 Normal Monitor Mode

$\overline{RST}$ and OSC1 functions will be active on the PTA5 and PTC0 pins, respectively, as long as V_{TST} is applied to the $\overline{IRQ}$ pin. If the $\overline{IRQ}$ pin is lowered (no longer V_{TST}) then the chip will still be operating in monitor mode, but the pin functions will be determined by the settings in the configuration register when V_{TST} was lowered. See Chapter 5 Configuration Register (CONFIG).

When monitor mode is entered with V_{TST} on $\overline{IRQ}$, the computer operating properly (COP) is disabled as long as V_{TST} is applied to $\overline{IRQ}$. This condition states that as long as V_{TST} is maintained on the $\overline{IRQ}$ pin after entering monitor mode, then the COP will be disabled.

19.3.1.2 Forced Monitor Mode

If the voltage applied to the $\overline{IRQ1}$ is less than V_{TST}, the MCU will come out of reset in user mode. However, when the reset vector is erased ($FFFF), the MCU is forced into monitor mode without requiring high voltage on the $\overline{IRQ1}$ pin. Once out of reset, the monitor code is initially executing off the internal clock at its default frequency.

If $\overline{IRQ}$ is tied high (V_{DD}), all pins will default to regular input port functions except for PTA0 and PTC0 which will operate as a serial communication port and OSC1 input respectively (refer to Figure 19-11). That will allow the clock to be driven from an external source through OSC1 pin.

If $\overline{IRQ}$ is tied low, all pins will default to regular input port function except for PTA0 which will operate as serial communication port. Refer to Figure 19-12. Regardless of the state of the $\overline{IRQ}$ pin, it will not function as a port input pin in monitor mode.

The COP module is disabled in forced monitor mode.

NOTE

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

19.3.1.3 Monitor Vectors

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

Table 19-2 summarizes the differences between user mode and monitor mode regarding vectors.

Table 19-2. Mode Difference

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

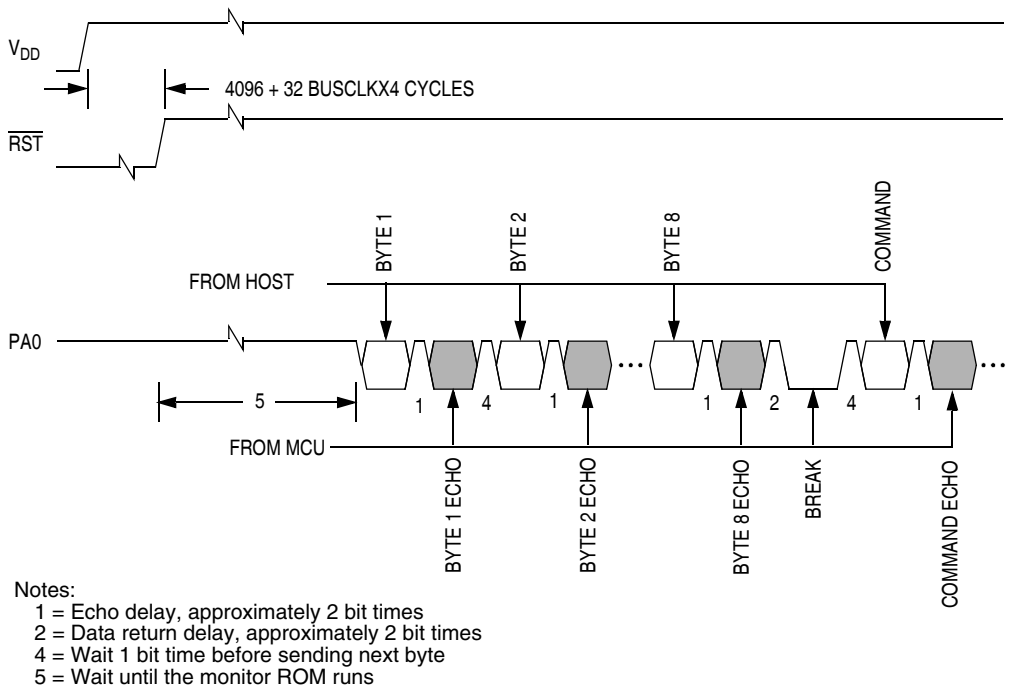


Figure 19-18. Monitor Mode Entry Timing

To determine whether the security code entered is correct, check to see if bit 6 of RAM address $80 is set. If it is, then the correct security code has been entered and FLASH can be accessed.

If the security sequence fails, the device should be reset by a power-on reset and brought up in monitor mode to attempt another entry. After failing the security sequence, the FLASH module can also be mass erased by executing an erase routine that was downloaded into internal RAM. The mass erase operation clears the security code locations so that all eight security bytes become $FF (blank).

Characteristic[1]	Symbol	Min	Typ[2]	Max	Unit
V_{DD} supply current Run[3] Wait[4] Stop[5] –40°C to 125°C[6] –40°C to 125°C with LVI enabled[6]	I_{DD}	 — — — —	 18 12 1 140	 25 15 10 300	 mA mA µA µA
I/O ports Hi-Z leakage current[6]	I_{IL}	–10	—	+10	µA
Input current	I_{In}	–1	—	+1	µA
Pullup resistors (as input only) Ports PTA6/KBD6–PTA0/KBD0, PTC2–PTC0, $\overline{RST}$, $\overline{IRQ}$	R_{PU}	16	26	36	kΩ
Capacitance Ports (as input or output)	C_{Out} C_{In}	— —	— —	12 8	pF
Monitor mode entry voltage	V_{TST}	V_{DD} + 2.5	—	9.1	V
Low-voltage inhibit, trip falling voltage	V_{TRIPF}	3.90	4.20	4.50	V
Low-voltage inhibit, trip rising voltage	V_{TRIPR}	4.00	4.30	4.60	V
Low-voltage inhibit reset/recover hysteresis (V_{TRIPF} + V_{HYS} = V_{TRIPR})	V_{HYS}	—	100	—	mV
POR rearm voltage[7]	V_{POR}	0	—	100	mV
POR reset voltage[8]	V_{PORRST}	0	700	800	mV
POR rise time ramp rate[9]	R_{POR}	0.035	—	—	V/ms

NOTES:
1. V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc, T_A = T_A (min) to T_A (max), 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 (f_{OSC} = 32 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 run I_{DD}. Measured with all modules enabled.
4. Wait I_{DD} measured using external square wave clock source (f_{OSC} = 32 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}. Measured with ICG and LVI enabled.
5. Stop I_{DD} is measured with OSC1 = V_{SS}.
6. Pullups and pulldowns are disabled. Port B leakage is specified in 20.8 5.0-Volt ADC Characteristics.
7. Maximum is highest voltage that POR is guaranteed.
8. Maximum is highest voltage that POR is possible.
9. 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.

20.6 5.0-Volt Control Timing

Characteristic[1]	Symbol	Min	Max	Unit
Internal operating frequency	f_{OP} (f_{Bus})	—	8	MHz
Internal clock period (1/f_{OP})	t_{CYC}	125	—	ns
$\overline{RST}$ input pulse width low[2]	t_{RL}	750	—	ns
$\overline{IRQ}$ interrupt pulse width low[3] (edge-triggered)	t_{ILIH}	50	—	ns
$\overline{IRQ}$ interrupt pulse period	t_{ILIL}	Note 5	—	t_{CYC}