

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	2MHz
Connectivity	-
Peripherals	LVD, POR, PWM
Number of I/O	13
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2.4V ~ 3.6V
Data Converters	A/D 4x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	16-DIP (0.300", 7.62mm)
Supplier Device Package	16-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mchlc908qy4cpe

MC68HLC908QY4
MC68HLC908QT4
MC68HLC908QY2
MC68HLC908QT2
MC68HLC908QY1
MC68HLC908QT1

Data Sheet

To provide the most up-to-date information, the revision of our documents on the World Wide Web will be the most current. Your printed copy may be an earlier revision. To verify you have the latest information available, refer to:

<http://freescale.com/>

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc.
This product incorporates SuperFlash® technology licensed from SST.

© Freescale Semiconductor, Inc., 2004. All rights reserved.

2.4 Input/Output (I/O) Section

Addresses \$0000–\$003F, shown in [Figure 2-2](#), contain most of the control, status, and data registers. Additional I/O registers have these addresses:

- \$FE00 — Break status register, BSR
- \$FE01 — 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 — Interrupt status register 3, INT3
- \$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
- \$FE0D — Reserved
- \$FFBE — FLASH block protect register, FLBPR
- \$FFC0 — Internal OSC trim value — Optional
- \$FFFF — COP control register, COPCTL

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
\$0000	Port A Data Register (PTA) See page 98.	Read:	R	AWUL	PTA5	PTA4	PTA3	PTA2	PTA1	PTA0
		Write:								
		Reset:	Unaffected by reset							
\$0001	Port B Data Register (PTB) See page 100.	Read:	PTB7	PTB6	PTB5	PTB4	PTB3	PTB2	PTB1	PTB0
		Write:								
		Reset:	Unaffected by reset							
\$0002	Unimplemented									
\$0003	Unimplemented									
\$0004	Data Direction Register A (DDRA) See page 98.	Read:	R	R	DDRA5	DDRA4	DDRA3	0	DDRA1	DDRA0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
				= Unimplemented		R	= Reserved		U = Unaffected	

Figure 2-2. Control, Status, and Data Registers (Sheet 1 of 6)

Memory

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
\$0005	Data Direction Register B (DDRB) See page 101.	Read:	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0006 ↓ \$000A	Unimplemented									
\$000B	Port A Input Pullup Enable Register (PTAPUE) See page 99.	Read:	OSC2EN	0	PTAPUE5	PTAPUE4	PTAPUE3	PTAPUE2	PTAPUE1	PTAPUE0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$000C	Port B Input Pullup Enable Register (PTBPUE) See page 102.	Read:	PTBPUE7	PTBPUE6	PTBPUE5	PTBPUE4	PTBPUE3	PTBPUE2	PTBPUE1	PTBPUE0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$000D ↓ \$0019	Unimplemented									
\$001A	Keyboard Status and Control Register (KBSCR) See page 83.	Read:	0	0	0	0	KEYF	0	IMASKK	MODEK
		Write:						ACKK		
		Reset:	0	0	0	0	0	0	0	0
\$001B	Keyboard Interrupt Enable Register (KBIER) See page 84.	Read:	0	AWUIE	KBIE5	KBIE4	KBIE3	KBIE2	KBIE1	KBIE0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$001C	Unimplemented									
\$001D	IRQ Status and Control Register (INTSCR) See page 77.	Read:	0	0	0	0	IRQF	0	IMASK	MODE
		Write:						ACK		
		Reset:	0	0	0	0	0	0	0	0
\$001E	Configuration Register 2 (CONFIG2) ⁽¹⁾ See page 53.	Read:	IRQPUD	IRQEN	R	OSCOPT1	OSCOPT0	R	R	RSTEN
		Write:								
		Reset:	0	0	0	0	0	0	0	0 ⁽²⁾

1. One-time writable register after each reset.

2. RSTEN reset to 0 by a power-on reset (POR) only.


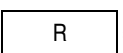
 = Unimplemented  = Reserved U = Unaffected

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

Chapter 3

Analog-to-Digital Converter (ADC)

3.1 Introduction

This section describes the 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 MC68HLC908QY2, MC68HLC908QT2, MC68HLC908QY4, and MC68HLC908QT4.

3.2 Features

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 frequency

3.3 Functional Description

Four ADC channels are available for sampling external sources at pins PTA0, PTA1, PTA4, and PTA5. An analog multiplexer allows the single ADC converter to select one of the four ADC channels as an ADC voltage input (ADCVIN). ADCVIN is converted by the successive approximation register-based counters. The ADC resolution is eight bits. When the conversion is completed, ADC puts the result in the ADC data register and sets a flag or generates an interrupt.

Figure 3-2 shows a block diagram of the ADC.

3.3.1 ADC Port I/O Pins

PTA0, PTA1, PTA4, and PTA5 are general-purpose I/O pins that are shared with the ADC channels. The channel select bits (ADC status and control register (ADSCR), \$003C), define which ADC channel/port pin will be used as 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 which is in use by the ADC will return a 0 if the corresponding DDR bit is at 0. If the DDR bit is 1, the value in the port data latch is read.

4.6.3 Keyboard Interrupt Enable Register

The keyboard interrupt enable register (KBIER) enables or disables the auto wakeup to operate as a keyboard/auto wakeup interrupt input.

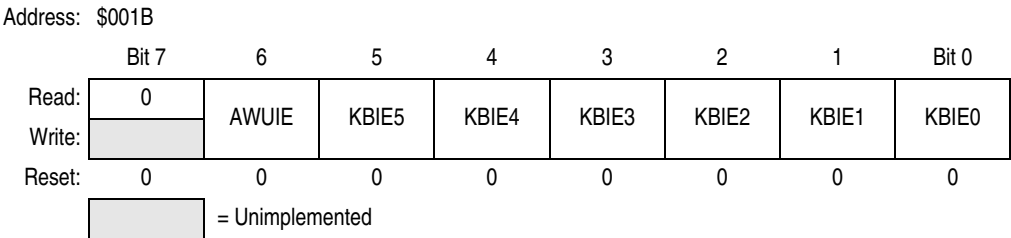


Figure 4-4. Keyboard Interrupt Enable Register (KBIER)

AWUIE — Auto Wakeup Interrupt Enable Bit

This read/write bit enables the auto wakeup interrupt input to latch interrupt requests. Reset clears AWUIE.

- 1 = Auto wakeup enabled as interrupt input
- 0 = Auto wakeup not enabled as interrupt input

NOTE

KBIE5–KBIE0 bits are not used in conjunction with the auto wakeup feature. To see a description of these bits, see [9.7.2 Keyboard Interrupt Enable Register](#).

Configuration Register (CONFIG)

IRQPUD — $\overline{\text{IRQ}}$ Pin Pullup Control Bit

- 1 = Internal pullup is disconnected
- 0 = Internal pullup is connected between $\overline{\text{IRQ}}$ pin and V_{DD}

IRQEN — $\overline{\text{IRQ}}$ Pin Function Selection Bit

- 1 = Interrupt request function active in pin
- 0 = Interrupt request function inactive in pin

OSCOPT1 and 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 — $\overline{\text{RST}}$ Pin Function Selection

- 1 = Reset function active in pin
- 0 = Reset function inactive in pin

NOTE

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

Address: \$001F

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	COPRS	LVISTOP	LVIRSTD	LVIPWRD	LVDLVR	SSREC	STOP	COPD
Write:								
Reset:	0	0	0	0	U	0	0	0
POR:	0	0	0	0	0	0	0	0

U = Unaffected

Figure 5-2. Configuration Register 1 (CONFIG1)

COPRS (Out of STOP Mode) — COP Reset Period Selection Bit

- 1 = COP reset short cycle = $8176 \times \text{BUSCLKX4}$
- 0 = COP reset long cycle = $262,128 \times \text{BUSCLKX4}$

COPRS (In STOP Mode) — Auto Wakeup Period Selection Bit

- 1 = Auto wakeup short cycle = $512 \times \text{INTRCOSC}$
- 0 = Auto wakeup long cycle = $16,384 \times \text{INTRCOSC}$

LVISTOP — LVI Enable in Stop Mode Bit

When the LVIPWRD bit is clear, setting the LVISTOP bit enables the LVI to operate during stop mode. Reset clears LVISTOP.

- 1 = LVI enabled during stop mode
- 0 = LVI disabled during stop mode

LVIRSTD — LVI Reset Disable Bit

LVIRSTD disables the reset signal from the LVI module. Unlike other configuration bits, the LVIRSTD can be written at any time.

- 1 = LVI module resets disabled
- 0 = LVI module resets enabled

Central Processor Unit (CPU)

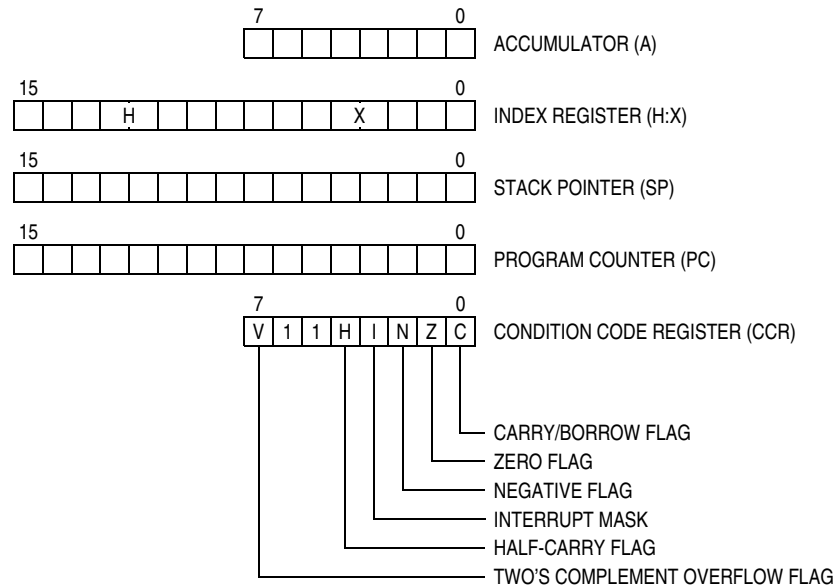


Figure 7-1. CPU Registers

7.3.1 Accumulator

The accumulator is a general-purpose 8-bit register. The CPU uses the accumulator to hold operands and the results of arithmetic/logic operations.

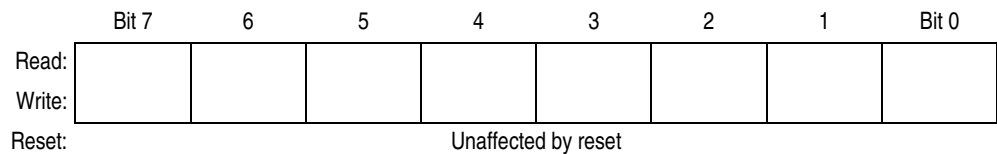


Figure 7-2. Accumulator (A)

7.3.2 Index Register

The 16-bit index register allows indexed addressing of a 64-Kbyte memory space. H is the upper byte of the index register, and X is the lower byte. H:X is the concatenated 16-bit index register.

In the indexed addressing modes, the CPU uses the contents of the index register to determine the conditional address of the operand.

The index register can serve also as a temporary data storage location.

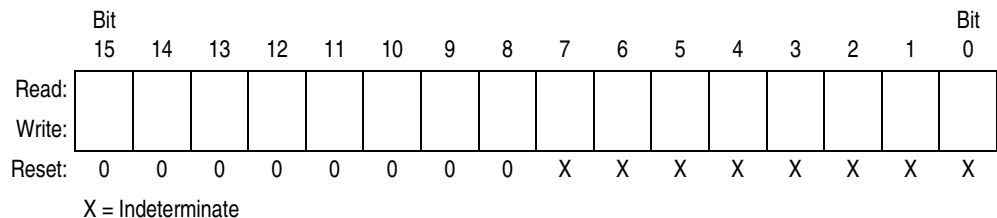


Figure 7-3. Index Register (H:X)

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

Source Form	Operation	Description	Effect on CCR						Address Mode	Opcode	Operand	Cycles
			V	H	I	N	Z	C				
SWI	Software Interrupt	PC \leftarrow (PC) + 1; Push (PCL) SP \leftarrow (SP) - 1; Push (PCH) SP \leftarrow (SP) - 1; Push (X) SP \leftarrow (SP) - 1; Push (A) SP \leftarrow (SP) - 1; Push (CCR) SP \leftarrow (SP) - 1; I \leftarrow 1 PCH \leftarrow Interrupt Vector High Byte PCL \leftarrow Interrupt Vector Low Byte	-	-	1	-	-	-	INH	83		9
TAP	Transfer A to CCR	CCR \leftarrow (A)	\uparrow	\uparrow	\uparrow	\uparrow	\uparrow	\uparrow	INH	84		2
TAX	Transfer A to X	X \leftarrow (A)	-	-	-	-	-	-	INH	97		1
TPA	Transfer CCR to A	A \leftarrow (CCR)	-	-	-	-	-	-	INH	85		1
TST <i>opr</i> TSTA TSTX TST <i>opr</i> ,X TST ,X TST <i>opr</i> ,SP	Test for Negative or Zero	(A) - \$00 or (X) - \$00 or (M) - \$00	0	-	-	\uparrow	\uparrow	-	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 \leftarrow (SP) + 1	-	-	-	-	-	-	INH	95		2
TXA	Transfer X to A	A \leftarrow (X)	-	-	-	-	-	-	INH	9F		1
TXS	Transfer H:X to SP	(SP) \leftarrow (H:X) - 1	-	-	-	-	-	-	INH	94		2
WAIT	Enable Interrupts; Wait for Interrupt	I bit \leftarrow 0; Inhibit CPU clocking until interrupted	-	-	0	-	-	-	INH	8F		1

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

7.8 Opcode Map

See [Table 7-2](#).

Chapter 9

Keyboard Interrupt Module (KBI)

9.1 Introduction

The keyboard interrupt module (KBI) provides six independently maskable external interrupts, which are accessible via the PTA0–PTA5 pins.

9.2 Features

Features of the keyboard interrupt module include:

- Six keyboard interrupt pins with separate keyboard interrupt enable bits and one keyboard interrupt mask
- Software configurable pullup device if input pin is configured as input port bit
- Programmable edge-only or edge and level interrupt sensitivity
- Exit from low-power modes

9.3 Functional Description

The keyboard interrupt module controls the enabling/disabling of interrupt functions on the six port A pins. These six pins can be enabled/disabled independently of each other.

9.3.1 Keyboard Operation

Writing to the KBIE0–KBIE5 bits in the keyboard interrupt enable register (KBIER) independently enables or disables each port A pin as a keyboard interrupt pin. Enabling a keyboard interrupt pin in port A also enables its internal pullup device irrespective of PTAPUEx bits in the port A input pullup enable register (see [12.2.3 Port A Input Pullup Enable Register](#)). 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 interrupt inputs 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 interrupt input does not latch an interrupt request if another keyboard pin is already low. To prevent losing an interrupt request on one input because another input is still low, software can disable the latter input 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 input is low.

Chapter 10

Low-Voltage Inhibit (LVI)

10.1 Introduction

This section describes the low-voltage inhibit (LVI) module, which monitors the voltage on the V_{DD} pin and can force a reset when the V_{DD} voltage falls below the LVI trip falling voltage, V_{TRIPF} .

10.2 Features

Features of the LVI module include:

- Programmable LVI reset
- Programmable power consumption
- Selectable LVI trip voltage
- Programmable stop mode operation

10.3 Functional Description

Figure 10-1 shows the structure of the LVI module. LVISTOP, LVIPWRD, LVDLVR, and LVIRSTD are user selectable options found in the configuration register (CONFIG1). See [Chapter 5 Configuration Register \(CONFIG\)](#).

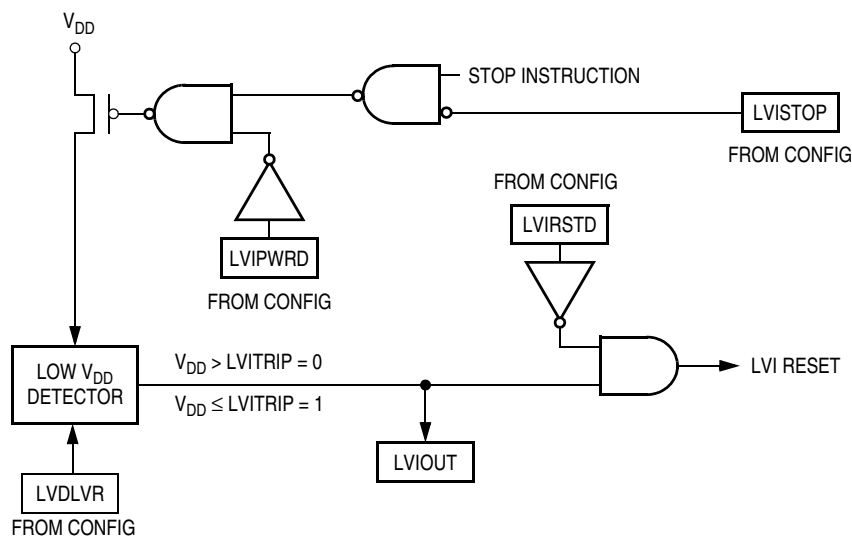


Figure 10-1. LVI Module Block Diagram

Figure 12-3 shows the port A I/O logic.

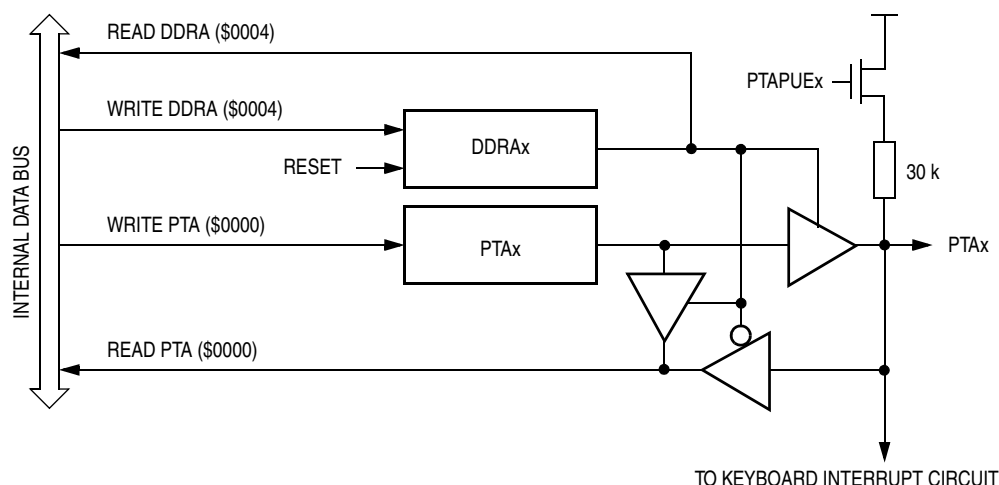


Figure 12-3. Port A I/O Circuit

NOTE

Figure 12-3 does not apply to PTA2

When DDRAx is a 1, reading address \$0000 reads the PTAx data latch. When DDRAx is a 0, reading address \$0000 reads the voltage level on the pin. The data latch can always be written, regardless of the state of its data direction bit.

12.2.3 Port A Input Pullup Enable Register

The port A input pullup enable register (PTAPUE) contains a software configurable pullup device for each of the six port A pins. Each bit is individually configurable and requires the corresponding data direction register, DDRAx, to be configured as input. Each pullup device is automatically and dynamically disabled when its corresponding DDRAx bit is configured as output.

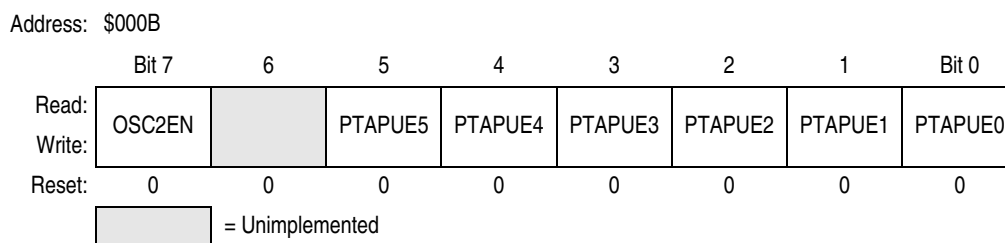


Figure 12-4. Port A Input Pullup Enable Register (PTAPUE)

OSC2EN — Enable PTA4 on OSC2 Pin

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

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

0 = OSC2 pin configured for PTA4 I/O, having all the interrupt and pullup functions

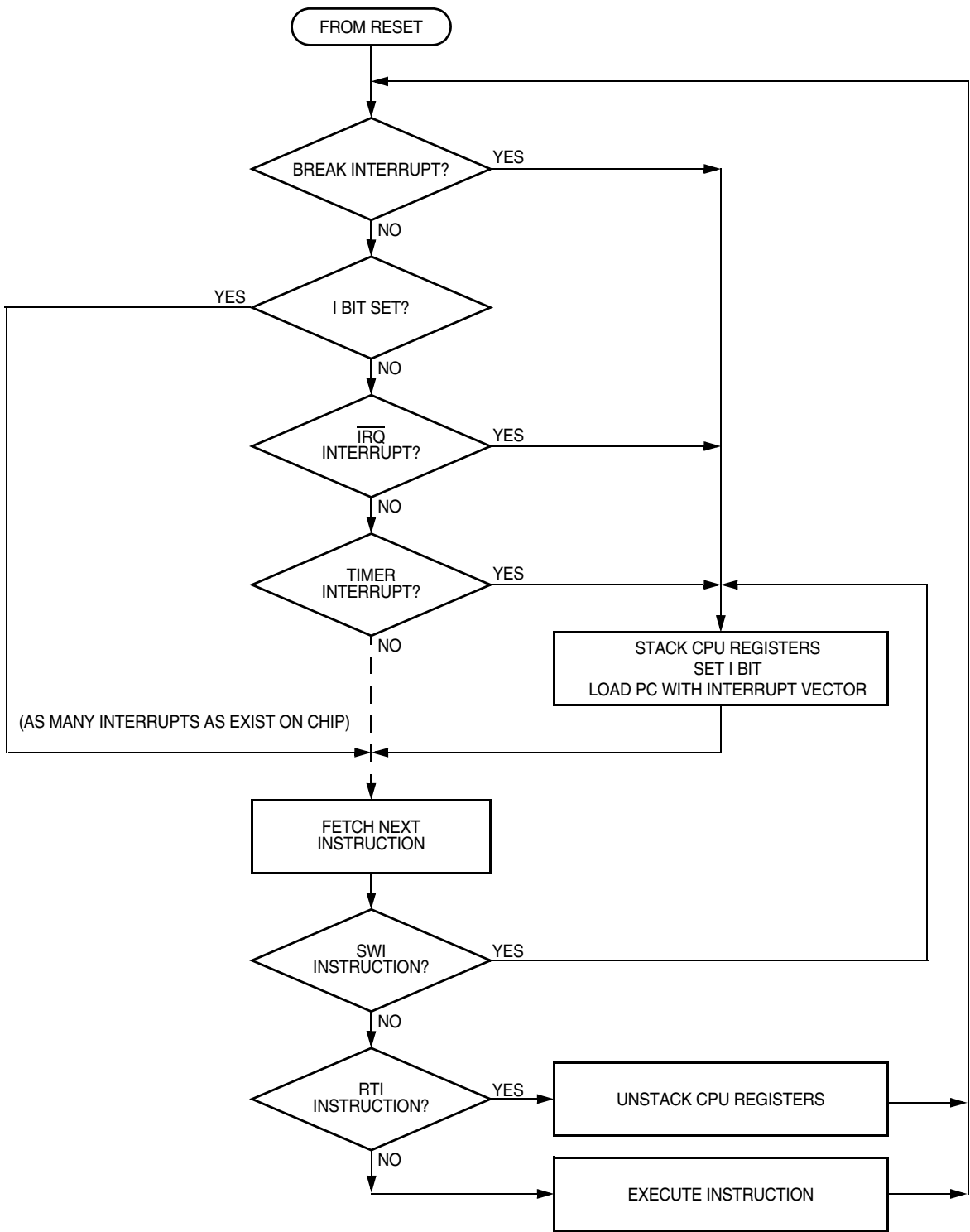


Figure 13-7. Interrupt Processing

state by forcing it to the SWI vector location. Refer to the break interrupt subsection of each module to see how each module is affected by the break state.

13.6.5 Status Flag Protection in Break Mode

The SIM controls whether status flags contained in other modules can be cleared during break mode. The user can select whether flags are protected from being cleared by properly initializing the break clear flag enable bit (BCFE) in the break flag control register (BFCR).

Protecting flags in break mode ensures that set flags will not be cleared while in break mode. This protection allows registers to be freely read and written during break mode without losing status flag information.

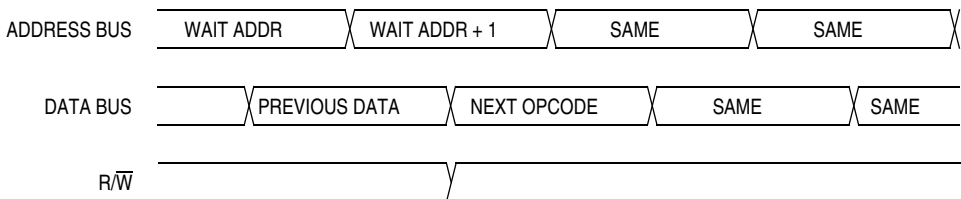
Setting the BCFE bit enables the clearing mechanisms. Once cleared in break mode, a flag remains cleared even when break mode is exited. Status flags with a two-step clearing mechanism — for example, a read of one register followed by the read or write of another — are protected, even when the first step is accomplished prior to entering break mode. Upon leaving break mode, execution of the second step will clear the flag as normal.

13.7 Low-Power Modes

Executing the WAIT or STOP instruction puts the MCU in a low power-consumption mode for standby situations. The SIM holds the CPU in a non-clocked state. The operation of each of these modes is described below. Both STOP and WAIT clear the interrupt mask (I) in the condition code register, allowing interrupts to occur.

13.7.1 Wait Mode

In wait mode, the CPU clocks are inactive while the peripheral clocks continue to run. [Figure 13-14](#) shows the timing for wait mode entry.



NOTE: Previous data can be operand data or the WAIT opcode, depending on the last instruction.

Figure 13-14. Wait Mode Entry Timing

A module that is active during wait mode can wake up the CPU with an interrupt if the interrupt is enabled. Stacking for the interrupt begins one cycle after the WAIT instruction during which the interrupt occurred. In wait mode, the CPU clocks are inactive. 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.

Wait mode can also be exited by a reset (or break in emulation mode). A break interrupt during wait mode sets the SIM break stop/wait bit, SBSW, in the break status register (BSR). If the COP disable bit, COPD, in the configuration register is 0, then the computer operating properly module (COP) is enabled and remains active in wait mode.

13.8.1 SIM Reset Status Register

The SRSR register contains flags that show the source of the last reset. The status register will automatically clear after reading SRSR. A power-on reset sets the POR bit and clears all other bits in the register. All other reset sources set the individual flag bits but do not clear the register. More than one reset source can be flagged at any time depending on the conditions at the time of the internal or external reset. For example, the POR and LVI bit can both be set if the power supply has a slow rise time.

Address: \$FE01

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


 = Unimplemented

Figure 13-19. SIM Reset Status Register (SRSR)

POR — Power-On Reset Bit

- 1 = Last reset caused by POR circuit
- 0 = Read of SRSR

PIN — External Reset Bit

- 1 = Last reset caused by external reset pin ($\overline{\text{RST}}$)
- 0 = POR or read of SRSR

COP — Computer Operating Properly Reset Bit

- 1 = Last reset caused by COP counter
- 0 = POR or read of SRSR

ILOP — Illegal Opcode Reset Bit

- 1 = Last reset caused by an illegal opcode
- 0 = POR or read of SRSR

ILAD — Illegal Address Reset Bit (illegal attempt to fetch an opcode from an unimplemented address)

- 1 = Last reset caused by an opcode fetch from an illegal address
- 0 = POR or read of SRSR

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 $\overline{\text{IRQ}} \neq V_{\text{TST}}$
- 0 = POR or read of SRSR

LVI — Low Voltage Inhibit Reset bit

- 1 = Last reset caused by LVI circuit
- 0 = POR or read of SRSR

14.4.1 TIM Counter Prescaler

The TIM clock source is one of the seven prescaler outputs or the TIM clock pin, TCLK. The prescaler generates seven clock rates from the internal bus clock. The prescaler select bits, PS[2:0], in the TIM status and control register (TSC) select the TIM clock source.

14.4.2 Input Capture

With the input capture function, the TIM can capture the time at which an external event occurs. When an active edge occurs on the pin of an input capture channel, the TIM latches the contents of the TIM counter into the TIM channel registers, TCHxH:TCHxL. The polarity of the active edge is programmable. Input captures can generate TIM central processor unit (CPU) interrupt requests.

14.4.3 Output Compare

With the output compare function, the TIM can generate a periodic pulse with a programmable polarity, duration, and frequency. When the counter reaches the value in the registers of an output compare channel, the TIM can set, clear, or toggle the channel pin. Output compares can generate TIM CPU interrupt requests.

14.4.3.1 Unbuffered Output Compare

Any output compare channel can generate unbuffered output compare pulses as described in [14.4.3 Output Compare](#). The pulses are unbuffered because changing the output compare value requires writing the new value over the old value currently in the TIM channel registers.

An unsynchronized write to the TIM channel registers to change an output compare value could cause incorrect operation for up to two counter overflow periods. For example, writing a new value before the counter reaches the old value but after the counter reaches the new value prevents any compare during that counter overflow period. Also, using a TIM overflow interrupt routine to write a new, smaller output compare value may cause the compare to be missed. The TIM may pass the new value before it is written.

Use the following methods to synchronize unbuffered changes in the output compare value on channel x:

- When changing to a smaller value, enable channel x output compare interrupts and write the new value in the output compare interrupt routine. The output compare interrupt occurs at the end of the current output compare pulse. The interrupt routine has until the end of the counter overflow period to write the new value.
- When changing to a larger output compare value, enable TIM overflow interrupts and write the new value in the TIM overflow interrupt routine. The TIM overflow interrupt occurs at the end of the current counter overflow period. Writing a larger value in an output compare interrupt routine (at the end of the current pulse) could cause two output compares to occur in the same counter overflow period.

14.4.3.2 Buffered Output Compare

Channels 0 and 1 can be linked to form a buffered output compare channel whose output appears on the TCH0 pin. The TIM channel registers of the linked pair alternately control the output.

Setting the MS0B bit in TIM channel 0 status and control register (TSC0) links channel 0 and channel 1. The output compare value in the TIM channel 0 registers initially controls the output on the TCH0 pin. Writing to the TIM channel 1 registers enables the TIM channel 1 registers to synchronously control the output after the TIM overflows. At each subsequent overflow, the TIM channel registers (0 or 1) that

control the output are the ones written to last. TSC0 controls and monitors the buffered output compare function, and TIM channel 1 status and control register (TSC1) is unused. While the MS0B bit is set, the channel 1 pin, TCH1, is available as a general-purpose I/O pin.

NOTE

In buffered output compare operation, do not write new output compare values to the currently active channel registers. User software should track the currently active channel to prevent writing a new value to the active channel. Writing to the active channel registers is the same as generating unbuffered output compares.

14.4.4 Pulse Width Modulation (PWM)

By using the toggle-on-overflow feature with an output compare channel, the TIM can generate a PWM signal. The value in the TIM counter modulo registers determines the period of the PWM signal. The channel pin toggles when the counter reaches the value in the TIM counter modulo registers. The time between overflows is the period of the PWM signal.

As [Figure 14-3](#) shows, the output compare value in the TIM channel registers determines the pulse width of the PWM signal. The time between overflow and output compare is the pulse width. Program the TIM to clear the channel pin on output compare if the state of the PWM pulse is logic 1 (ELSxA = 0). Program the TIM to set the pin if the state of the PWM pulse is logic 0 (ELSxA = 1).

The value in the TIM counter modulo registers and the selected prescaler output determines the frequency of the PWM output. The frequency of an 8-bit PWM signal is variable in 256 increments. Writing \$00FF (255) to the TIM counter modulo registers produces a PWM period of 256 times the internal bus clock period if the prescaler select value is 000. See [14.9.1 TIM Status and Control Register](#).

The value in the TIM channel registers determines the pulse width of the PWM output. The pulse width of an 8-bit PWM signal is variable in 256 increments. Writing \$0080 (128) to the TIM channel registers produces a duty cycle of 128/256 or 50%.

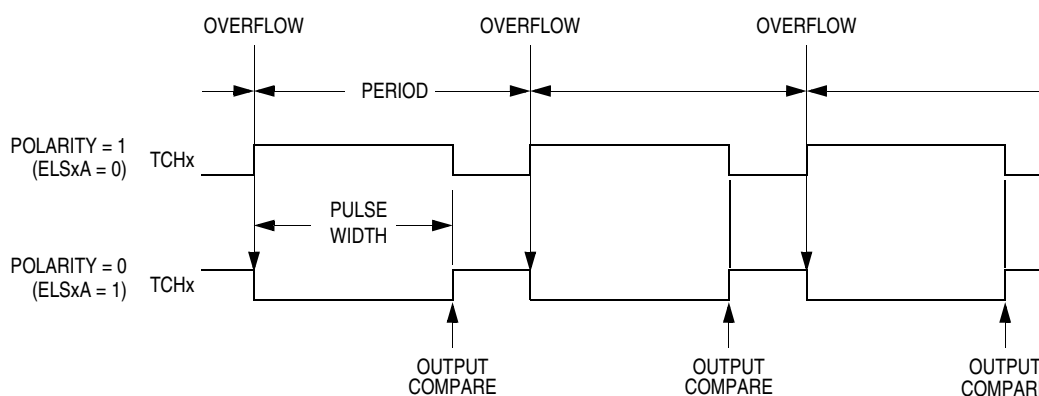


Figure 14-3. PWM Period and Pulse Width

Address: \$0021	TCNTH							
	Bit 7	6	5	4	3	2	1	Bit 0
Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Write:								
Reset:	0	0	0	0	0	0	0	0

Address: \$0022	TCNTL							
	Bit 7	6	5	4	3	2	1	Bit 0
Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Write:								
Reset:	0	0	0	0	0	0	0	0


 = Unimplemented

Figure 14-5. TIM Counter Registers (TCNTH:TCNTL)

14.9.3 TIM Counter Modulo Registers

The read/write TIM modulo registers contain the modulo value for the TIM counter. 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. Reset sets the TIM counter modulo registers.

Address: \$0023	TMODH							
	Bit 7	6	5	4	3	2	1	Bit 0
Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Write:								
Reset:	1	1	1	1	1	1	1	1

Address: \$0024	TMODL							
	Bit 7	6	5	4	3	2	1	Bit 0
Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Write:								
Reset:	1	1	1	1	1	1	1	1

Figure 14-6. TIM Counter Modulo Registers (TMODH:TMODL)

NOTE

Reset the TIM counter before writing to the TIM counter modulo registers.

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.

Address: \$FE02

	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

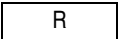
0 = COP enabled during break interrupt.

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.

Address: \$FE00

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

 = 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.

Address: \$FE03

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

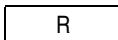
 = Reserved

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

15.3.1.2 Forced Monitor Mode

If entering monitor mode without high voltage on \overline{IRQ} , then startup port pin requirements and conditions, (PTA1/PTA4) are not in effect. This is to reduce circuit requirements when performing in-circuit programming.

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 reset vector has been programmed, the traditional method of applying a voltage, V_{TST} , to \overline{IRQ} must be used to enter monitor mode.

If monitor mode was entered as a result of the reset vector being blank, the COP is always disabled regardless of the state of \overline{IRQ} .

If the voltage applied to the \overline{IRQ} is less than V_{TST} , the MCU will come out of reset in user mode. Internal circuitry monitors the reset vector fetches and will assert an internal reset if it detects that the reset vectors are erased (\$FF). When the MCU comes out of reset, it is forced into monitor mode without requiring high voltage on the \overline{IRQ} pin. Once out of reset, the monitor code is initially executing with the internal clock at its default frequency.

If \overline{IRQ} is held high, all pins will default to regular input port functions except for PTA0 and PTA5 which will operate as a serial communication port and OSC1 input respectively (refer to Figure 15-11). That will allow the clock to be driven from an external source through OSC1 pin.

If \overline{IRQ} is held low, all pins will default to regular input port function except for PTA0 which will operate as serial communication port. Refer to Figure 15-12.

Regardless of the state of the \overline{IRQ} pin, it will not function as a port input pin in monitor mode. Bit 2 of the Port A data register will always read 0. The BIH and BIL instructions will behave as if the \overline{IRQ} pin is enabled, regardless of the settings in the configuration register. See Chapter 5 Configuration Register (CONFIG).

The COP module is disabled in forced monitor mode. Any reset other than a power-on reset (POR) will automatically force the MCU to come back to the forced monitor mode.

15.3.1.3 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.

NOTE

Exiting monitor mode after it has been initiated by having a blank reset vector requires a power-on reset (POR). Pulling \overline{RST} (when \overline{RST} pin available) low will not exit monitor mode in this situation.

Table 15-2 summarizes the differences between user mode and monitor mode regarding vectors.

Table 15-2. 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

16.10 Analog-to-Digital (ADC) Converter Characteristics

16.10.1 ADC Electrical Operating Conditions

The ADC accuracy characteristics below are guaranteed over two operating conditions as stated here.

	Characteristic	Symbol	Min	Max	Unit
Condition A	ATD supply	V_{DD}	2.7	3.6	V
	ADC internal clock	f_{ADIC}	0.008	1	MHz
	Ambient temperature	T_A	T_L	T_H	°C
Condition B	ATD supply	V_{DD}	2.3	2.7	V
	ADC internal clock	f_{ADIC}	8	63	kHz
	Ambient temperature	T_A	0	T_H	°C

16.10.2 ADC Performance Characteristics

Characteristic	Symbol	Min	Max	Unit	Comments
Input voltages	V_{ADIN}	V_{SS}	V_{DD}	V	—
Resolution (1 LSB) Condition A Condition B	RES	10.5 8.99	14.1 10.5	mV	—
Absolute accuracy (Total unadjusted error) Condition A Condition B	E_{TUE}	— —	± 1.5 ± 2.0	LSB	Includes quantization
Conversion range	V_{AIN}	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⁾	I_{IL}	—	± 1	μA	—
ADC supply current ($V_{DD} = 3 V$)	I_{ADAD}	Typical = 0.45		mA	Enabled

1. Source impedances greater than 10 k Ω 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.