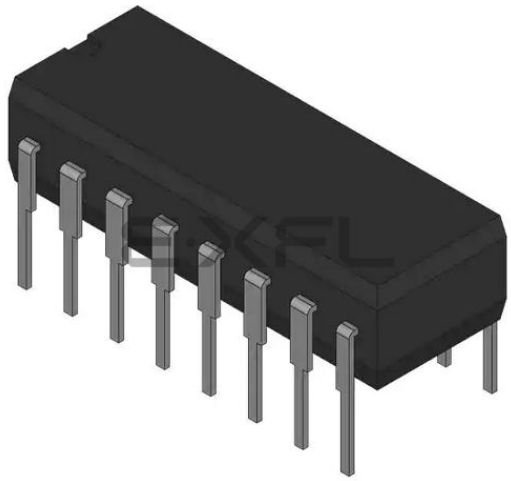


Welcome to E-XFL.COM

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	SCI
Peripherals	LVD, POR, PWM
Number of I/O	13
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	192 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 4x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	16-SOIC (0.295", 7.50mm Width)
Supplier Device Package	16-SOIC
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc908kx8vdwe

The following revision history table summarizes changes contained in this document. For your convenience, the page number designators have been linked to the appropriate location.

Revision History

Date	Revision Level	Description	Page Number(s)
April, 2001	0.1	Label for pin 9 corrected in Figure 1-1 and Figure 1-2	19, 20
		$FF is the erase state of the FLASH, not $00.	82, 252, 255
		First bulleted paragraph under the subsection 15.5 Interrupts reworded for clarity	177
		Revision to the description of the CHxMAX bit and the note that follows that description	183
		Forced monitor mode information added to Table 16-1.	192
		In Figure 16-10. Monitor Data Format, resistor value for connection between VTST and IRQ1 changed from 10 kΩ to 1 kΩ.	194
February, 2002	1.0	7.2 Features — Corrected third bullet	71
		7.7.3 ICG Trim Register — Corrected description of the TRIM7:TRIM0 bits	97
		14.2 Features — Corrected divide by factors in first bullet	165
		Figure 14-1. Timebase Block Diagram — Corrected divide-by-2 blocks	166
		Table 14-1. Timebase Divider Selection — Corrected last divider tap entry	167
		Section 15. Timer Interface Module (TIM) — Timer discrepancies corrected throughout this section	169
		17.4 Thermal Characteristics — Corrected SOIC thermal resistance and maximum junction temperature	202
		17.5 5.0-Vdc DC Electrical Characteristics and — Corrected footnote for VDD supply current in stop mode	203 and 204
		Appendix B. MC68HC08KX8 — Added to supply exception information for the MC68HC08KX8	215
March, 2004	2.0	Reformatted to current publication standards	Throughout
		2.7 FLASH Page Erase Operation — Updated procedure	33
		2.8 FLASH Mass Erase Operation — Updated procedure	33
		2.9 FLASH Program/Read Operation — Updated procedure	34
		Figure 5-1. COP Block Diagram — Updated figure	53
		Table 6-1. Instruction Set Summary — Added WAIT instruction	69
		Section 7. Internal Clock Generator Module (ICG) — Updated with new information	71 through 98
		14.2 Features — Corrected values given in the first bullet	165
		Table 15-3. Mode, Edge, and Level Selection — Reworked for clarity	182
		17.11 Memory Characteristics — Updated table with new information	210
July, 2005	2.1	Updated to meet Freescale identity guidelines.	Throughout

Chapter 11
Input/Output (I/O) Ports (PORTS)

Chapter 12
Serial Communications Interface Module (SCI)

- Serial communications interface (SCI) module
- 5-bit keyboard interrupt (KBI) with wakeup feature
- 13 general-purpose input/output (I/O) ports:
  - Five shared with KBI and TIM, with 15-mA source/15-mA sink capabilities and with programmable pullups on general- purpose input ports
  - Four shared with ADC
  - Two shared with SCI
- Low-voltage inhibit (LVI) module with software selectable trip points, 2.6-V or 4.3-V trip point
- Timebase module (TBM) with
  - Clock prescaler for eight user-selectable, periodic real-time interrupts
  - Active clock source in stop mode for periodic wakeup from stop using external crystal or internal oscillator
- External asynchronous interrupt pin with internal pullup ($\overline{\text{IRQ1}}$)
- System protection features:
  - Computer operating properly (COP) reset
  - Low-voltage detection with reset
  - Illegal opcode detection with reset
  - Illegal address detection with reset
- 16-pin plastic dual in-line (PDIP) or small outline (SOIC) package
- Low-power design fully static with stop and wait modes
- Internal power-up reset circuit requiring no external pins
- –40°C to +125°C operation

Features of the CPU08 include:

- Enhanced HC05 programming model
- Extensive loop control functions
- 16 addressing modes, eight more than the M68HC05
- 16-bit index register and stack pointer
- Memory-to-memory data transfers
- Fast 8 × 8 multiply instruction
- Fast 16/8 divide instruction
- Binary-coded decimal (BCD) instructions
- Optimization for controller applications
- Third party C language support

1.3 MCU Block Diagram

Figure 1-1 shows the structure of the MC68HC908KX8.

1.4 Pin Assignments

Figure 1-2 shows the pin assignments for MC68HC908KX8.

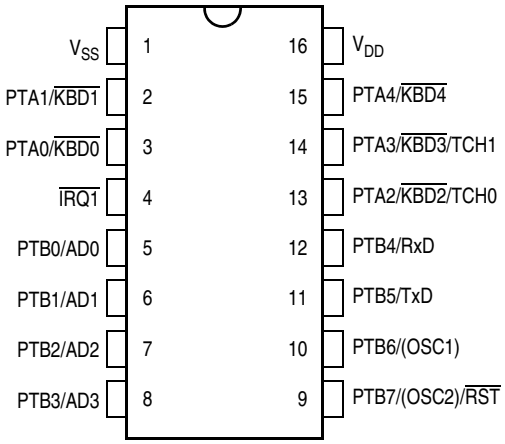
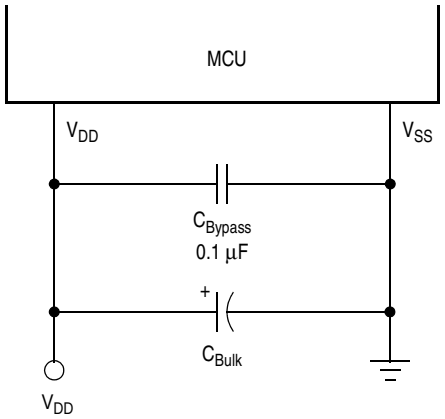


Figure 1-2. PDIP and SOIC Pin Assignments

1.4.1 Supply Pins (V_{DD} and V_{SS})

V_{DD} and V_{SS} are the power supply and ground pins. The MCU operates from a single power supply.

Fast signal transitions on MCU pins place high, short-duration current demands on the power supply. To prevent noise problems, take special care to provide power supply bypassing at the MCU as shown in Figure 1-3. Place the bypass capacitors as close to the MCU power pins as possible. Use high-frequency response ceramic capacitors for C_{Bypass}. C_{Bulk} are optional bulk current bypass capacitors for use in applications that require the port pins to source high-current levels.



Note: Component values shown represent typical applications.

Figure 1-3. Power Supply Bypassing

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0022	Timer Counter Register Low (TCNTL) See page 164.	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 (TMODH) See page 165.	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 (TMODL) See page 165.	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 (TSC0) See page 165.	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 (TCH0H) See page 168.	Read:	Bit 15	14	13	12	11	10	9	Bit 8
		Write:								
		Reset:	Indeterminate after reset							
$0027	Timer Channel 0 Register Low (TCH0L) See page 168.	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
		Reset:	Indeterminate after reset							
$0028	Timer Channel 1 Status and Control Register (TSC1) See page 165.	Read:	CH1F	CH1IE	0	MS1A	ELS1B	ELS1A	TOV1	CH1MAX
		Write:	0							
		Reset:	0	0	0	0	0	0	0	0
$0029	Timer Channel 1 Register High (TCH1H) See page 168.	Read:	Bit 15	14	13	12	11	10	9	Bit 8
		Write:								
		Reset:	Indeterminate after reset							
$002A	Timer Channel 1 Register Low (TCH1L) See page 168.	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
		Reset:	Indeterminate after reset							
$002B ↓ $0035	Unimplemented									
$0036	ICG Control Register (ICGCR) See page 87.	Read:	CMIE	CMF	CMON	CS	ICGON	ICGS	ECGON	ECGS
		Write:		0(1)						
		Reset:	0	0	0	0	1	0	0	0
$0037	ICG Multiplier Register (ICGMR) See page 88.	Read:		N6	N5	N4	N3	N2	N1	N0
		Write:								
		Reset:	0	0	0	1	0	1	0	1

= Unimplemented R = Reserved U = Unaffected

1. See 7.7.1 ICG Control Register for method of clearing the CMF bit.

Figure 2-2. Control, Status, and Data Registers (Sheet 3 of 5)

Z — Zero Flag

The CPU sets the zero flag when an arithmetic operation, logic operation, or data manipulation produces a result of $00.

1 = Zero result
0 = Non-zero result

C — Carry/Borrow Flag

The CPU sets the carry/borrow flag when an addition operation produces a carry out of bit 7 of the accumulator or when a subtraction operation requires a borrow. Some instructions — such as bit test and branch, shift, and rotate — also clear or set the carry/borrow flag.

1 = Carry out of bit 7
0 = No carry out of bit 7

6.4 Arithmetic/Logic Unit (ALU)

The ALU performs the arithmetic and logic operations defined by the instruction set.

Refer to the *CPU08 Reference Manual* (document order number CPU08RM/AD) for a description of the instructions and addressing modes and more detail about the architecture of the CPU.

6.5 Low-Power Modes

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

6.5.1 Wait Mode

The WAIT instruction:

- Clears the interrupt mask (I bit) in the condition code register, enabling interrupts. After exit from wait mode by interrupt, the I bit remains clear. After exit by reset, the I bit is set.
- Disables the CPU clock

6.5.2 Stop Mode

The STOP instruction:

- Clears the interrupt mask (I bit) in the condition code register, enabling external interrupts. After exit from stop mode by external interrupt, the I bit remains clear. After exit by reset, the I bit is set.
- Disables the CPU clock

After exiting stop mode, the CPU clock begins running after the oscillator stabilization delay.

6.6 CPU During Break Interrupts

If a break module is present on the MCU, the CPU starts a break interrupt by:

- Loading the instruction register with the SWI instruction
- Loading the program counter with $FFFC:$FFFD or with $FEFC:$FEFD in monitor mode

The break interrupt begins after completion of the CPU instruction in progress. If the break address register match occurs on the last cycle of a CPU instruction, the break interrupt begins immediately.

A return-from-interrupt instruction (RTI) in the break routine ends the break interrupt and returns the MCU to normal operation if the break interrupt has been deasserted.

Chapter 7
Internal Clock Generator Module (ICG)

7.1 Introduction

The internal clock generator module (ICG) is used to create a stable clock source for the microcontroller without using any external components. The ICG generates the oscillator output clock (CGMXCLK), which is used by the computer operating properly (COP), low-voltage inhibit (LVI), and other modules. The ICG also generates the clock generator output (CGMOUT), which is fed to the system integration module (SIM) to create the bus clocks. The bus frequency will be one-fourth the frequency of CGMXCLK and one-half the frequency of CGMOUT. Finally, the ICG generates the timebase clock (TBMCLK), which is used in the timebase module (TBM).

7.2 Features

Features of the ICG include:

- Selectable external clock generator, either one pin external source or two pin crystal, multiplexed with port pins
- Internal clock generator with programmable frequency output in integer multiples of a nominal frequency (307.2 kHz ± 25%)
- Frequency adjust (trim) register to improve variability to ± 2%
- Bus clock software selectable from either internal or external clock (bus frequency range from 76.8 kHz ± 25% to 9.75 MHz ± 25% in 76.8 kHz increments — note that for the MC68HC908KX8, MC68HC908KX2, and MC68HC08KX8, do not exceed the maximum bus frequency of 8 MHz at 5.0 V and 4 MHz at 3.0 V
- Timebase clock automatically selected externally, if external clock is available
- Clock monitor for both internal and external clocks

7.3 Functional Description

As shown in Figure 7-1, the ICG contains these major submodules:

- Clock enable circuit
- Internal clock generator
- External clock generator
- Clock monitor circuit
- Clock selection circuit

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0039	ICG DCO Divider Control Register (ICGDVR) See page 89.	Read:					DDIV3	DDIV2	DDIV1	DDIV0
		Write:								
		Reset:	0	0	0	0	U	U	U	U
$003A	ICG DCO Stage Control Register (ICGDSR) See page 89.	Read:	DSTG7	DSTG6	DSTG5	DSTG4	DSTG3	DSTG2	DSTG1	DSTG0
		Write:	R	R	R	R	R	R	R	R
		Reset:	U	U	U	U	U	U	U	U

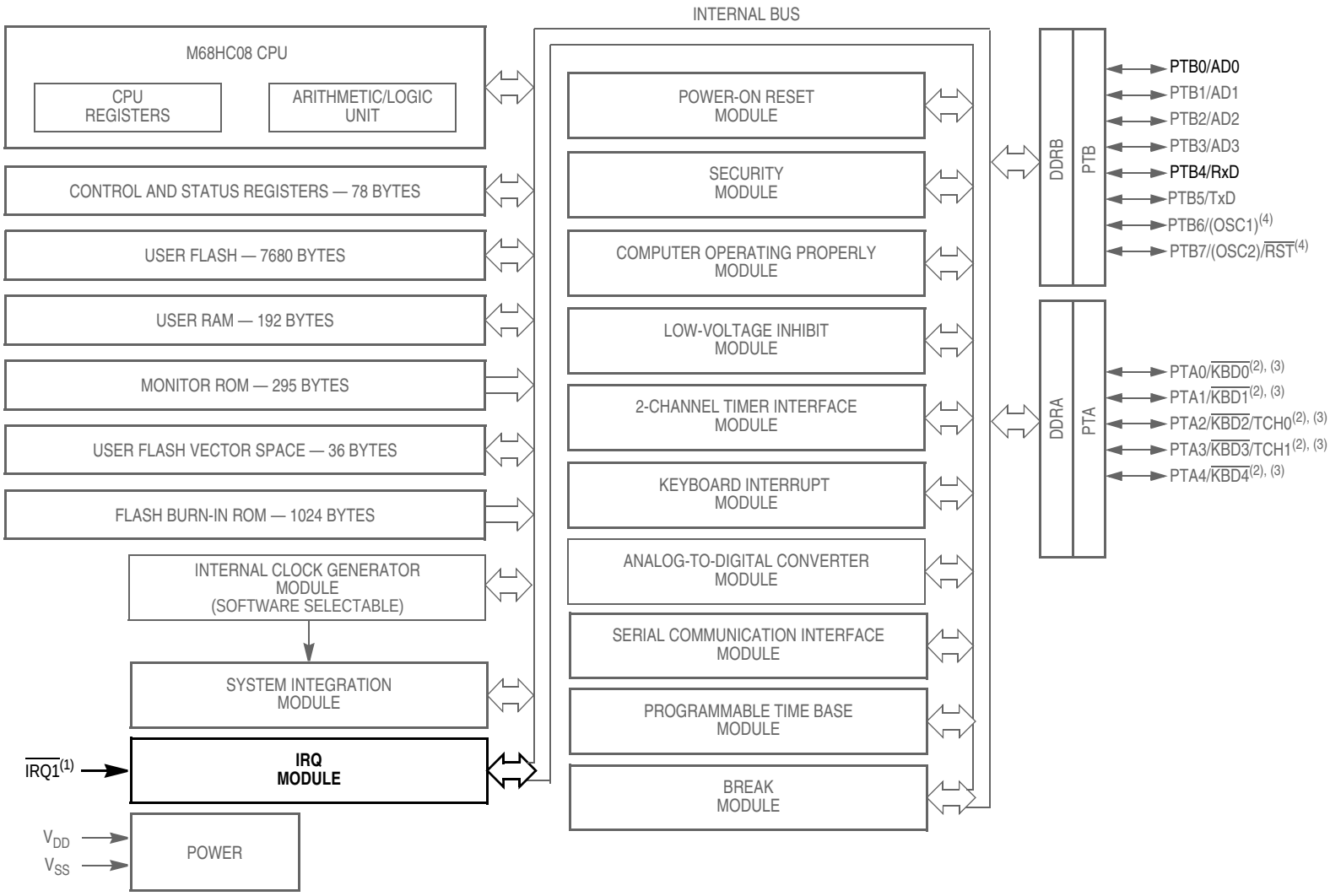
= Unimplemented R = Reserved U = Unaffected

Figure 7-10. ICG Module I/O Register Summary (Continued)

Table 7-5. ICG Module Register Bit Interaction Summary

Condition	Register Bit Results for Given Condition											
	CMIE	CMF	CMON	CS	ICGON	ICGS	ECGON	ECGS	N[6:0]	TRIM[7:0]	DDIV[3:0]	DSTB[7:0]
Reset	0	0	0	0	1	0	0	0	$15	$80	—	—
OSCENINSTOP = 0, STOP = 1	0	0	0	—	—	0	—	0	—	—	—	—
EXTCLKEN = 0	0	0	0	0	1	—	0	0	—	—	uw	uw
CMF = 1	—	(1)	1	—	1	—	1	—	uw	uw	uw	uw
CMON = 0	0	0	(0)	—	—	—	—	—	—	—	—	—
CMON = 1	—	—	(1)	—	1	—	1	—	uw	uw	uw	uw
CS = 0	—	—	—	(0)	1	—	—	—	—	—	uw	uw
CS = 1	—	—	—	(1)	—	—	1	—	—	—	—	—
ICGON = 0	0	0	0	1	(0)	0	1	—	—	—	—	—
ICGON = 1	—	—	—	—	(1)	—	—	—	—	—	uw	uw
ICGS = 0	us	—	us	uc	—	(0)	—	—	—	—	—	—
ECGON = 0	0	0	0	0	1	—	(0)	0	—	—	uw	uw
ECGS = 0	us	—	us	us	—	—	—	(0)	—	—	—	—
IOFF = 1	—	1*	(1)	1	(1)	0	(1)	—	uw	uw	uw	uw
EOFF = 1	—	1*	(1)	0	(1)	—	(1)	0	uw	uw	uw	uw
N = written	(0)	(0)	(0)	—	—	0*	—	—	—	—	—	—
TRIM = written	(0)	(0)	(0)	—	—	0*	—	—	—	—	—	—

—Register bit is unaffected by the given condition.
0, 1Register bit is forced clear or set (respectively) in the given condition.
0*, 1*Register bit is temporarily forced clear or set (respectively) in the given condition.
(0), (1)Register bit must be clear or set (respectively) for the given condition to occur.
us, uc, uwRegister bit cannot be set, cleared, or written (respectively) in the given condition.



Notes:
1. Pin contains integrated pullup resistor
2. High-current source/sink pin
3. Pin contains software selectable pullup resistor if general function I/O pin is configured as input.

Figure 8-1. Block Diagram Highlighting IRQ Block and Pins

DDRA4–DDRA0 — Data Direction Register A Bits

These read/write bits control port A data direction. Reset clears DDRA4–DDRA0, configuring all port A pins as inputs.

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

NOTE

Avoid glitches on port A pins by writing to the port A data register before changing data direction register A bits from 0 to 1.

Figure 11-4 shows the port A I/O logic.

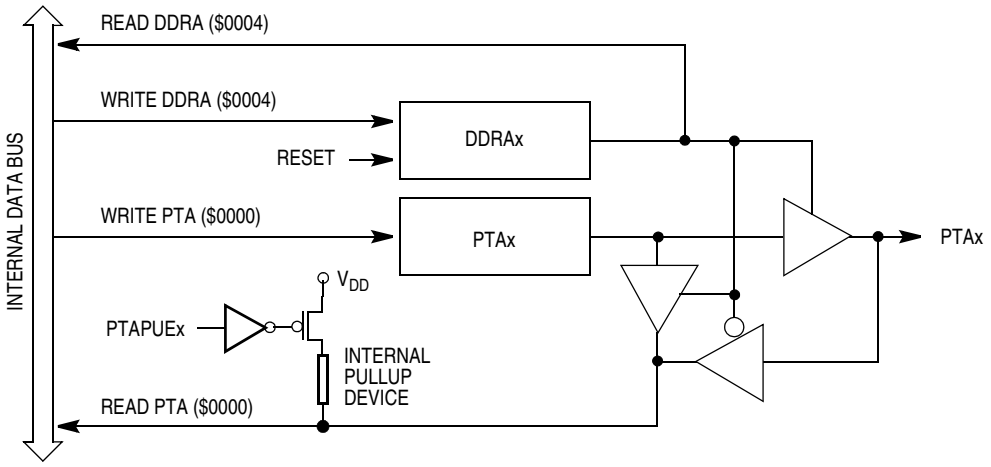


Figure 11-4. Port A I/O Circuit

When bit DDRAx is a 1, reading address $0000 reads the PTAx data latch. When bit 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. Table 11-1 summarizes the operation of the port A pins.

Table 11-1. Port A Pin Functions

PTAPUE Bit	DDRA Bit	PTA Bit	I/O Pin Mode	Accesses to DDRA	Accesses to PTA	
				Read/Write	Read	Write
1	0	X	Input, V_{DD}[1]	DDRA4–DDRA0	Pin	PTA4–PTA0[2]
0	0	X	Input, Hi-Z	DDRA4–DDRA0	Pin	PTA4–PTA0[3]
X	1	X	Output	DDRA4–DDRA0	PTA4–PTA0	PTA4–PTA0

X = Don't care
Hi-Z = High impedance

1. I/O pin pulled up to V_{DD} by internal pulllup device
2. Writing affects data register, but does not affect input.

11.2.3 Port A Input Pullup Enable Register

The port A input pullup enable register (PTAPUE) contains a software configurable pullup device for each of the five port A pins. Each bit is individually configurable and requires that the data direction register, DDRA, bit be configured as an input. Each pullup is automatically disabled when a port bit's DDRA is configured for output mode.

Chapter 12
Serial Communications Interface Module (SCI)

12.1 Introduction

The serial communications interface (SCI) allows asynchronous communications with peripheral devices and other microcontroller unit (MCU).

12.2 Features

The SCI module's features include:

- Full-duplex operation
- Standard mark/space non-return-to-zero (NRZ) format
- Choice of baud rate clock source:
  - Internal bus clock
  - CGMXCLK
- 32 programmable baud rates
- Programmable 8-bit or 9-bit character length
- Separately enabled transmitter and receiver
- Separate receiver and transmitter central processor unit (CPU) interrupt requests
- Programmable transmitter output polarity
- Two receiver wakeup methods:
  - Idle line wakeup
  - Address mark wakeup
- Interrupt-driven operation with eight interrupt flags:
  - Transmitter empty
  - Transmission complete
  - Receiver full
  - Idle receiver input
  - Receiver overrun
  - Noise error
  - Framing error
  - Parity error
- Receiver framing error detection
- Hardware parity checking
- 1/16 bit-time noise detection

12.4.3.1 Character Length

The receiver can accommodate either 8-bit or 9-bit data. The state of the M bit in SCI control register 1 (SCC1) determines character length. When receiving 9-bit data, bit R8 in SCI control register 2 (SCC2) is the ninth bit (bit 8). When receiving 8-bit data, bit R8 is a copy of the eighth bit (bit 7).

12.4.3.2 Character Reception

During an SCI reception, the receive shift register shifts characters in from the RxD pin. The SCI data register (SCDR) is the read-only buffer between the internal data bus and the receive shift register.

After a complete character shifts into the receive shift register, the data portion of the character transfers to the SCDR. The SCI receiver full bit, SCRF, in SCI status register 1 (SCS1) becomes set, indicating that the received byte can be read. If the SCI receive interrupt enable bit, SCRIE, in SCC2 is also set, the SCRF bit generates a receiver CPU interrupt request.

12.4.3.3 Data Sampling

The receiver samples the RxD pin at the RT clock rate. The RT clock is an internal signal with a frequency 16 times the baud rate. To adjust for baud rate mismatch, the RT clock is resynchronized at these times (see Figure 12-7):

- After every start bit
- After the receiver detects a data bit change from 1 to 0 (after the majority of data bit samples at RT8, RT9, and RT10 returns a valid 1 and the majority of the next RT8, RT9, and RT10 samples returns a valid 0)

To locate the start bit, data recovery logic does an asynchronous search for a 0 preceded by three 1s. When the falling edge of a possible start bit occurs, the RT clock begins to count to 16.

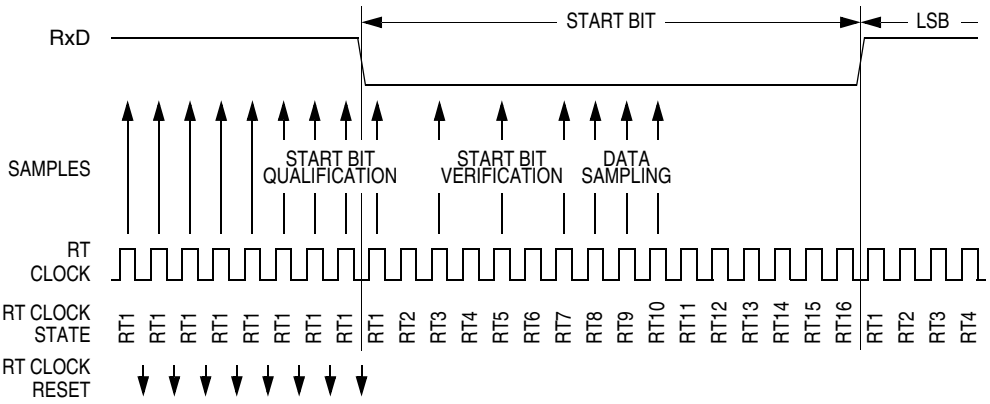


Figure 12-7. Receiver Data Sampling

Table 12-8. SCI Baud Rate Selection Examples

SCP[1:0]	Prescaler Divisor (PD)	SCR[2:1:0]	Baud Rate Divisor (BD)	Baud Rate ($f_{BAUDCLK}$ = 4.9152 MHz)
00	1	000	1	76,800
00	1	001	2	38,400
00	1	010	4	19,200
00	1	011	8	9600
00	1	100	16	4800
00	1	101	32	2400
00	1	110	64	1200
00	1	111	128	600
01	3	000	1	25,600
01	3	001	2	12,800
01	3	010	4	6400
01	3	011	8	3200
01	3	100	16	1600
01	3	101	32	800
01	3	110	64	400
01	3	111	128	200
10	4	000	1	19,200
10	4	001	2	9600
10	4	010	4	4800
10	4	011	8	2400
10	4	100	16	1200
10	4	101	32	600
10	4	110	64	300
10	4	111	128	150
11	13	000	1	5908
11	13	001	2	2954
11	13	010	4	1477
11	13	011	8	739
11	13	100	16	369
11	13	101	32	185
11	13	110	64	92
11	13	111	128	46

Chapter 13
System Integration Module (SIM)

13.1 Introduction

This section describes the system integration module (SIM), which supports up to 24 external and/or internal interrupts. The SIM is a system state controller that coordinates the central processor unit (CPU) and exception timing. Together with the CPU, the SIM controls all microcontroller unit (MCU) activities.

Figure 13-1 is a summary of the SIM input/output (I/O) registers. A block diagram of the SIM is shown in Figure 13-2.

The SIM is responsible for:

- Bus clock generation and control for CPU and peripherals:
  - Stop/wait/reset entry and recovery
  - Internal clock control
- Master reset control, including power-on reset (POR) and computer operating properly (COP) timeout
- Interrupt control:
  - Acknowledge timing
  - Arbitration control timing
  - Vector address generation
- CPU enable/disable timing
- Modular architecture expandable to 128 interrupt sources

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$FE01	SIM Reset Status Register (SRSR) See page 148.	Read:	POR	0	COP	ILOP	ILAD	MENRST	LVI	0
		Write:								
		POR:	1	0	0	0	0	0	0	0
$FE04	Interrupt Status Register 1 (INT1) See page 149.	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	Interrupt Status Register 2 (INT2) See page 150.	Read:	IF14	IF13	IF12	IF11	IF10	IF9	IF8	IF7
		Write:	R	R	R	R	R	R	R	R
		Reset:	0	0	0	0	0	0	0	0
$FE06	Interrupt Status Register 3 (INT3) See page 150.	Read:	IF22	IF21	IF20	IF19	IF18	IF17	IF16	IF15
		Write:	R	R	R	R	R	R	R	R
		Reset:	0	0	0	0	0	0	0	0

= Unimplemented R = Reserved

Figure 13-1. SIM I/O Register Summary

14.7 Timebase Control Register

The timebase has one register, the timebase control register (TBCR), which is used to enable the timebase interrupts and set the rate.

Address: $001C

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	TBIF	TBR2	TBR1	TBR0	0	TBIE	TBON	R
Write:					TACK			
Reset:	0	0	0	0	0	0	0	0

= Unimplemented R = Reserved

Figure 14-2. Timebase Control Register (TBCR)

TBIF — Timebase Interrupt Flag

This read-only flag bit is set when the timebase counter has rolled over.

1 = Timebase interrupt pending
0 = Timebase interrupt not pending

TBR2–TBR0 — Timebase Divider Selection Bits

These read/write bits select the tap in the counter to be used for timebase interrupts as shown in Table 14-1.

NOTE

Do not change TBR2–TBR0 bits while the timebase is enabled (TBON = 1).

TACK— Timebase ACKnowledge Bit

The TACK bit is a write-only bit and always reads as 0. Writing a 1 to this bit clears TBIF, the timebase interrupt flag bit. Writing a 0 to this bit has no effect.

1 = Clear timebase interrupt flag
0 = No effect

TBIE — Timebase Interrupt Enabled Bit

This read/write bit enables the timebase interrupt when the TBIF bit becomes set. Reset clears the TBIE bit.

1 = Timebase interrupt is enabled.
0 = Timebase interrupt is disabled.

TBON — Timebase Enabled Bit

This read/write bit enables the timebase. Timebase may be turned off to reduce power consumption when its function is not necessary. The counter can be initialized by clearing and then setting this bit. Reset clears the TBON bit.

1 = Timebase is enabled.
0 = Timebase is disabled and the counter initialized to 0s.

15.8 I/O Registers

These I/O registers control and monitor operation of the TIM:

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

15.8.1 TIM Status and Control Register

The TIM status and control register (TSC):

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

Address: $0020

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

= Unimplemented

Figure 15-5. TIM Status and Control Register (TSC)

TOF — TIM Overflow Flag Bit

This read/write flag is set when the TIM counter reaches the modulo value programmed in the TIM counter modulo registers. Clear TOF by reading the TIM status and control register when TOF is set and then writing a 0 to TOF. If another TIM overflow occurs before the clearing sequence is complete, then writing 0 to TOF has no effect. Therefore, a TOF interrupt request cannot be lost due to inadvertent clearing of TOF. Reset clears the TOF bit. Writing a 1 to TOF has no effect.

1 = TIM counter has reached modulo value.
0 = TIM counter has not reached modulo value.

TOIE — TIM Overflow Interrupt Enable Bit

This read/write bit enables TIM overflow interrupts when the TOF bit becomes set. Reset clears the TOIE bit.

1 = TIM overflow interrupts enabled
0 = TIM overflow interrupts disabled

TSTOP — TIM Stop Bit

This read/write bit stops the TIM counter. Counting resumes when TSTOP is cleared. Reset sets the TSTOP bit, stopping the TIM counter until software clears the TSTOP bit.

1 = TIM counter stopped
0 = TIM counter active

NOTE

Do not set the TSTOP bit before entering wait mode if the TIM is required to exit wait mode.

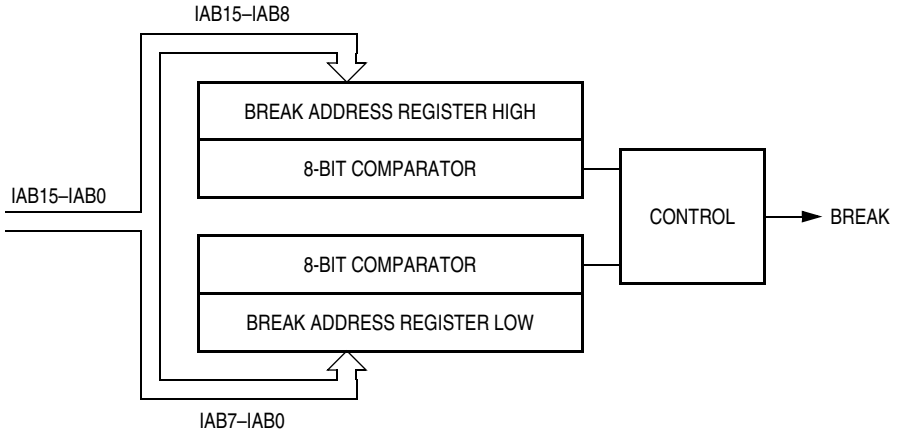


Figure 16-1. Break Module Block Diagram

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$FE00	SIM Break Status Register (SBSR) See page 172.	Read:	0	0	0	1	0	0	BW	0
		Write:	R	R	R	R	R	R	NOTE	R
		Reset:	0	0	0	1	0	0	0	0
$FE03	SIM Break Flag Control Register (SBFCR) See page 173.	Read:	BCFE	R	R	R	R	R	R	R
		Write:								
		Reset:	0							
$FE09	Break Address Register High (BRKH) See page 172.	Read:	Bit 15	14	13	12	11	10	9	Bit 8
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$FE0A	Break Address Register Low (BRKL) See page 172.	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$FE0B	Break Status and Control Register (BRKSCR) See page 171.	Read:	BRKE	BRKA	0	0	0	0	0	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$FE02	Break Auxiliary Register (BRKAR) See page 173.	Read:	0	0	0	0	0	0	0	BDCOP
		Write:								
		Reset:	0	0	0	0	0	0	0	0

Note: Writing a 0 clears BW.

= Unimplemented R = Reserved

Figure 16-2. I/O Register Summary

16.2.1.2 CPU During Break Interrupts

The CPU starts a break interrupt by:

- Loading the instruction register with the SWI instruction
- Loading the program counter with $FFFC and $FFFD ($FEFC and $FEFD in monitor mode)

The break interrupt begins after completion of the CPU instruction in progress. If the break address register match occurs on the last cycle of a CPU instruction, the break interrupt begins immediately.

16.2.1.3 TIM1 and TIM2 During Break Interrupts

A break interrupt stops the timer counters.

16.2.1.4 COP During Break Interrupts

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

16.2.2 Break Module Registers

These registers control and monitor operation of the break module:

- Break status and control register (BRKSCR)
- Break address register high (BRKH)
- Break address register low (BRKL)
- SIM break status register (SBSR)
- SIM break flag control register (SBFCR)

16.2.2.1 Break Status and Control Register

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

Address: $FE0B

	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 16-3. Break Status and Control Register (BRKSCR)

BRKE — Break Enable Bit

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

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

BRKA — Break Active Bit

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

1 = When read, break address match
0 = When read, no break address match

16.2.2.4 Break Flag Control Register

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

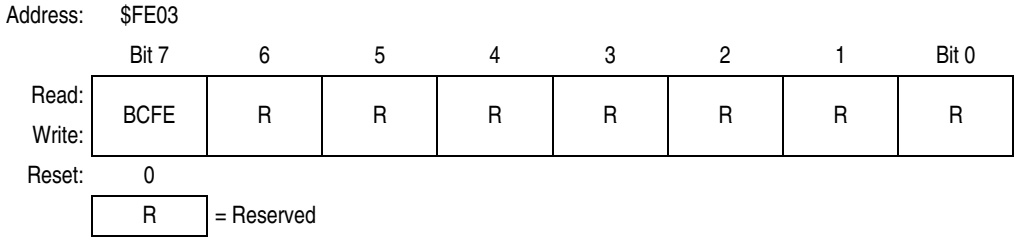


Figure 16-7. SIM Break Flag Control Register (SBFCR)

BCFE — Break Clear Flag Enable Bit

This read/write bit enables software to clear status bits by accessing status registers while the MCU is in a break state. To clear status bits during the break state, the BCFE bit must be set.

1 = Status bits clearable during break
0 = Status bits not clearable during break

16.2.2.5 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.

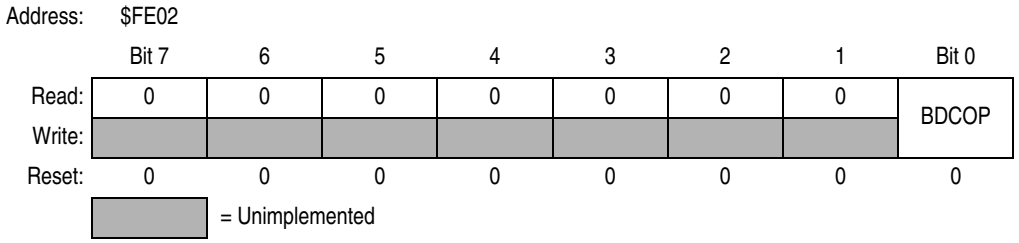


Figure 16-8. Break Auxiliary Register (BRKAR)

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

16.2.3 Low-Power Modes

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

16.2.3.1 Wait Mode

If enabled, the break module is active in wait mode. In the break routine, the user can subtract one from the return address on the stack if SBSW is set. Clear the BW bit by writing 0 to it.

16.2.3.2 Stop Mode

A break interrupt causes exit from stop mode and sets the BW bit in the break status register.

16.3.1.10 Commands

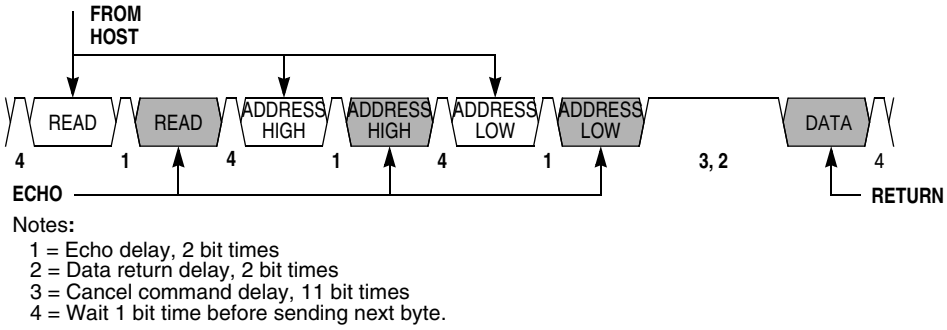
The monitor ROM firmware uses these commands:

- READ, read memory
- WRITE, write memory
- IREAD, indexed read
- IWRITE, indexed write
- READSP, read stack pointer
- RUN, run user program

The monitor ROM firmware echoes each received byte back to the PTA0 pin for error checking. An 11-bit delay at the end of each command allows the host to send a break character to cancel the command. A delay of two bit times occurs before each echo and before READ, IREAD, or READSP data is returned. The data returned by a read command appears after the echo of the last byte of the command.

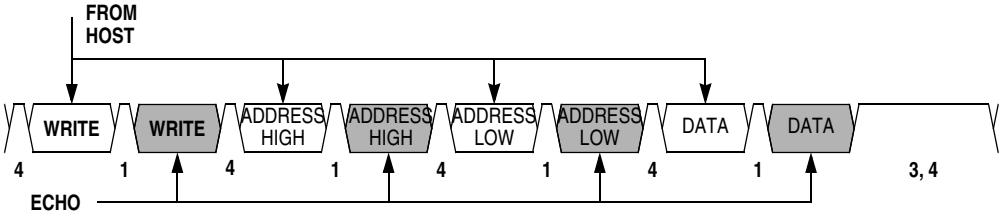
NOTE

Wait one bit time after each echo before sending the next byte.



Notes:
1 = Echo delay, 2 bit times
2 = Data return delay, 2 bit times
3 = Cancel command delay, 11 bit times
4 = Wait 1 bit time before sending next byte.

Figure 16-12. Read Transaction



Notes:
1 = Echo delay, 2 bit times
3 = Cancel command delay, 11 bit times
4 = Wait 1 bit time before sending next byte.

Figure 16-13. Write Transaction