



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	LED, LVD, POR, PWM
Number of I/O	15
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.3V
Data Converters	A/D 12x8b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	20-DIP (0.300", 7.62mm)
Supplier Device Package	20-DIP
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc68hrc908jk3cp

7.8.1	Break Status Register (BSR)	91
7.8.2	Reset Status Register (RSR)	92
7.8.3	Break Flag Control Register (BFCR)	94

Section 8. Oscillator (OSC)

8.1	Contents	95
8.2	Introduction	95
8.3	X-tal Oscillator (MC68HC908xxx)	96
8.4	RC Oscillator (MC68HRC908xxx)	97
8.5	I/O Signals	98
8.5.1	Crystal Amplifier Input Pin (OSC1)	98
8.5.2	Crystal Amplifier Output Pin (OSC2/PTA6/RCCLK)	98
8.5.3	Oscillator Enable Signal (SIMOSCEN)	98
8.5.4	X-tal Oscillator Clock (XTALCLK)	98
8.5.5	RC Oscillator Clock (RCCLK)	99
8.5.6	Oscillator Out 2 (2OSCOUT)	99
8.5.7	Oscillator Out (OSCOUT)	99
8.6	Low Power Modes	99
8.6.1	Wait Mode	99
8.6.2	Stop Mode	99
8.7	Oscillator During Break Mode	100

Section 9. Monitor ROM (MON)

9.1	Contents	101
9.2	Introduction	101
9.3	Features	102
9.4	Functional Description	102
9.4.1	Entering Monitor Mode	104
9.4.2	Baud Rate	107
9.4.3	Data Format	108
9.4.4	Echoing	108
9.4.5	Break Signal	109

List of Figures

Figure	Title	Page
1-1	MCU Block Diagram	24
1-2	MCU Pin Assignments	25
2-1	Memory Map	28
2-2	Control, Status, and Data Registers	30
4-1	FLASH Control Register (FLCR)	40
4-2	FLASH Programming Flowchart	45
4-3	FLASH Block Protect Register (FLBPR)	46
5-1	Configuration Register 2 (CONFIG2)	48
5-2	Configuration Register 1 (CONFIG1)	49
6-1	CPU Registers	53
6-2	Accumulator (A)	53
6-3	Index Register (H:X)	54
6-4	Stack Pointer (SP)	55
6-5	Program Counter (PC)	55
6-6	Condition Code Register (CCR)	56
7-1	SIM Block Diagram	73
7-2	SIM I/O Register Summary	74
7-3	SIM Clock Signals	75
7-4	External Reset Timing	77
7-5	Internal Reset Timing	77
7-6	Sources of Internal Reset	77
7-7	POR Recovery	78
7-8	Interrupt Processing	82
7-9	Interrupt Entry	83
7-10	Interrupt Recovery	83



Section 1. General Description

1.1 Contents

- 1.2 Introduction21
- 1.3 Features22
- 1.4 MCU Block Diagram23
- 1.5 Pin Assignments25
- 1.6 Pin Functions26

1.2 Introduction

The MC68H(R)C908JL3 is a member of the low-cost, high-performance M68HC08 Family of 8-bit microcontroller units (MCUs). The M68HC08 Family is based on the customer-specified integrated circuit (CSIC) design strategy. All MCUs in the family use the enhanced M68HC08 central processor unit (CPU08) and are available with a variety of modules, memory sizes and types, and package types.

Table 1-1. Summary of Device Variations

Device	FLASH Memory Size	Pin Count
MC68H(R)C908JL3	4096 bytes	28 pins
MC68H(R)C908JK3	4096 bytes	20 pins
MC68H(R)C908JK1	1536 bytes	20 pins

All references to the MC68H(R)C908JL3 in this data book apply equally to the MC68H(R)C908JK3 and MC68H(R)C908JK1, unless otherwise stated.

Section 2. Memory

2.1 Contents

2.2 Introduction27

2.3 I/O Section29

2.4 Monitor ROM29

2.2 Introduction

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

- 4096 bytes of user FLASH for MC68H(R)C908JL3/JK3
1536 bytes of user FLASH for MC68H(R)C908JK1
- 128 bytes of RAM
- 48 bytes of user-defined vectors
- 960 bytes of Monitor ROM

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
\$0000	Port A Data Register (PTA)	Read:	0	PTA6	PTA5	PTA4	PTA3	PTA2	PTA1	PTA0
		Write:								
		Reset:	Unaffected by reset							
\$0001	Port B Data Register (PTB)	Read:	PTB7	PTB6	PTB5	PTB4	PTB3	PTB2	PTB1	PTB0
		Write:								
		Reset:	Unaffected by reset							
\$0002	Unimplemented	Read:								
		Write:								
\$0003	Port D Data Register (PTD)	Read:	PTD7	PTD6	PTD5	PTD4	PTD3	PTD2	PTD1	PTD0
		Write:								
		Reset:	Unaffected by reset							
\$0004	Data Direction Register A (DDRA)	Read:	0	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0005	Data Direction Register B (DDRB)	Read:	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0006	Unimplemented	Read:								
		Write:								
\$0007	Data Direction Register D (DDRD)	Read:	DDRD7	DDRD6	DDRD5	DDRD4	DDRD3	DDRD2	DDRD1	DDRD0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0008 ↓ \$0009	Unimplemented	Read:								
\$000A	Port D Control Register (PDCR)	Read:	0	0	0	0	SLOWD7	SLOWD6	PTDPU7	PTDPU6
		Write:								
		Reset:	0	0	0	0	0	0	0	0
				= Unimplemented			R	= Reserved		

Figure 2-2. Control, Status, and Data Registers (Sheet 1 of 5)

NOTE: *The time between each FLASH address change (step 6 to step 6), or the time between the last FLASH addressed programmed to clearing the PGM bit (step 6 to step 10), must not exceed the maximum programming time, $t_{PROG\ max}$.*

NOTE: *Programming and erasing of FLASH locations cannot be performed by code being executed from the FLASH memory. While these operations must be performed in the order shown, other unrelated operations may occur between the steps.*

4.8 FLASH Protection

Due to the ability of the on-board charge pump to erase and program the FLASH memory in the target application, provision is made to protect blocks of memory from unintentional erase or program operations due to system malfunction. This protection is done by use of a FLASH Block Protect Register (FLBPR). The FLBPR determines the range of the FLASH memory which is to be protected. The range of the protected area starts from a location defined by FLBPR and ends to the bottom of the FLASH memory (\$FFFF). When the memory is protected, the HVEN bit cannot be set in either ERASE or PROGRAM operations.



6.6.2 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.

6.7 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.

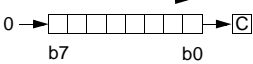
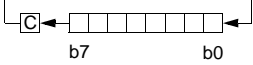
A return-from-interrupt instruction (RTI) in the break routine ends the break interrupt and returns the MCU to normal operation if the break interrupt has been deasserted.

6.8 Instruction Set Summary

6.9 Opcode Map

See [Table 6-2](#).

Table 6-1. Instruction Set Summary

Source Form	Operation	Description	Effect on CCR						Address Mode	Opcode	Operand	Cycles
			V	H	I	N	Z	C				
LSR <i>opr</i> LSRA LSRX LSR <i>opr</i> ,X LSR ,X LSR <i>opr</i> ,SP	Logical Shift Right		↑	–	–	0	↑	↑	DIR INH IX1 IX SP1	34 44 54 64 74 9E64	dd ff ff	4 1 1 4 3 5
MOV <i>opr</i> , <i>opr</i> MOV <i>opr</i> ,X+ MOV # <i>opr</i> , <i>opr</i> MOV X+, <i>opr</i>	Move	(M) _{Destination} ← (M) _{Source} H:X ← (H:X) + 1 (IX+D, DIX+)	0	–	–	↑	↑	–	DD DIX+ IMD IX+D	4E 5E 6E 7E	dd dd dd ii dd dd	5 4 4 4
MUL	Unsigned multiply	X:A ← (X) × (A)	–	0	–	–	–	0	INH	42		5
NEG <i>opr</i> NEGA NEGX NEG <i>opr</i> ,X NEG ,X NEG <i>opr</i> ,SP	Negate (Two's Complement)	M ← –(M) = \$00 – (M) A ← –(A) = \$00 – (A) X ← –(X) = \$00 – (X) M ← –(M) = \$00 – (M) M ← –(M) = \$00 – (M)	↑	–	–	↑	↑	↑	DIR INH INH IX1 IX SP1	30 40 50 60 70 9E60	dd ff ff	4 1 1 4 3 5
NOP	No Operation	None	–	–	–	–	–	–	INH	9D		1
NSA	Nibble Swap A	A ← (A[3:0]:A[7:4])	–	–	–	–	–	–	INH	62		3
ORA # <i>opr</i> ORA <i>opr</i> ORA <i>opr</i> ORA <i>opr</i> ,X ORA <i>opr</i> ,X ORA ,X ORA <i>opr</i> ,SP ORA <i>opr</i> ,SP	Inclusive OR A and M	A ← (A) (M)	0	–	–	↑	↑	–	IMM DIR EXT IX2 IX1 IX SP1 SP2	AA BA CA DA EA FA 9EEA 9EDA	ii dd hh ll ee ff ff ff ff ee ff	2 3 4 4 3 2 4 5
PSHA	Push A onto Stack	Push (A); SP ← (SP) – 1	–	–	–	–	–	–	INH	87		2
PSHH	Push H onto Stack	Push (H); SP ← (SP) – 1	–	–	–	–	–	–	INH	8B		2
PSHX	Push X onto Stack	Push (X); SP ← (SP) – 1	–	–	–	–	–	–	INH	89		2
PULA	Pull A from Stack	SP ← (SP + 1); Pull (A)	–	–	–	–	–	–	INH	86		2
PULH	Pull H from Stack	SP ← (SP + 1); Pull (H)	–	–	–	–	–	–	INH	8A		2
PULX	Pull X from Stack	SP ← (SP + 1); Pull (X)	–	–	–	–	–	–	INH	88		2
ROL <i>opr</i> ROLA ROLX ROL <i>opr</i> ,X ROL ,X ROL <i>opr</i> ,SP	Rotate Left through Carry		↑	–	–	↑	↑	↑	DIR INH INH IX1 IX SP1	39 49 59 69 79 9E69	dd ff ff	4 1 1 4 3 5

At the beginning of an interrupt, the CPU saves the CPU register contents on the stack and sets the interrupt mask (I bit) to prevent additional interrupts. At the end of an interrupt, the RTI instruction recovers the CPU register contents from the stack so that normal processing can resume. **Figure 7-9** shows interrupt entry timing. **Figure 7-10** shows interrupt recovery timing.

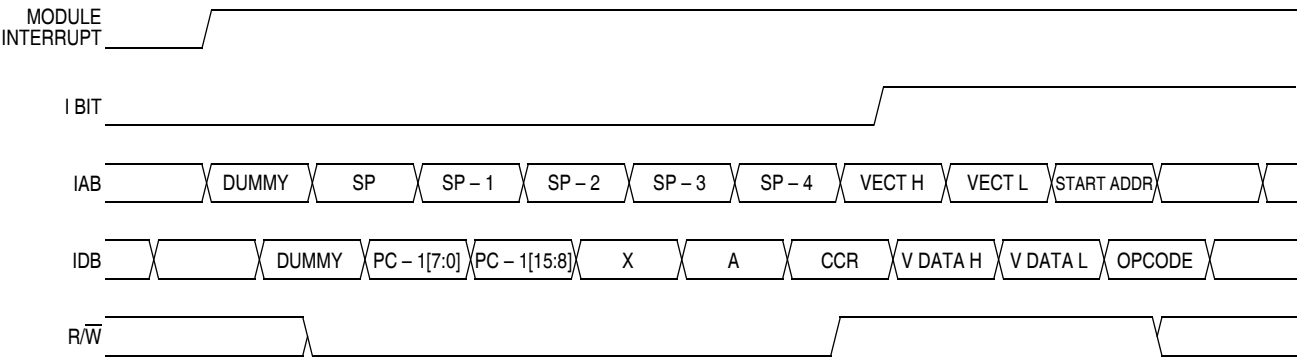


Figure 7-9. Interrupt Entry

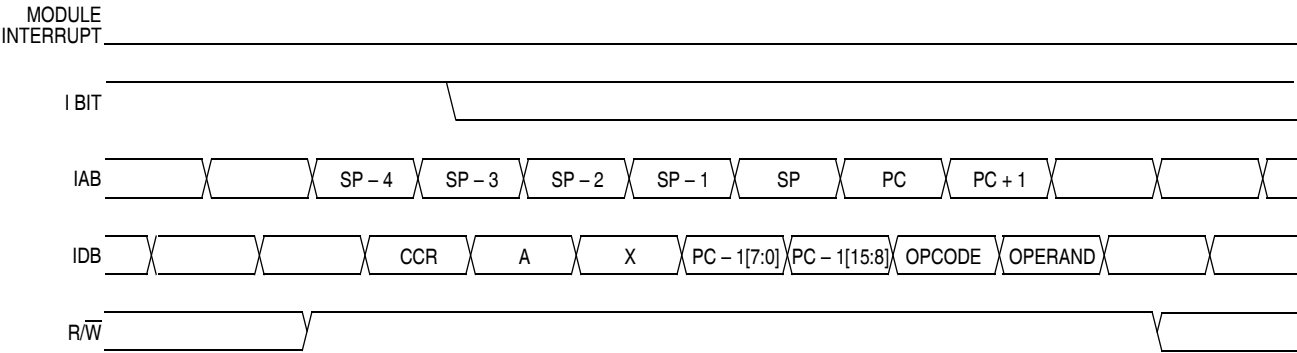


Figure 7-10. Interrupt Recovery

7.6.1.1 Hardware Interrupts

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

set, the SIM proceeds with interrupt processing; otherwise, the next instruction is fetched and executed.

If more than one interrupt is pending at the end of an instruction execution, the highest priority interrupt is serviced first. **Figure 7-11** demonstrates what happens when two interrupts are pending. If an interrupt is pending upon exit from the original interrupt service routine, the pending interrupt is serviced before the LDA instruction is executed.

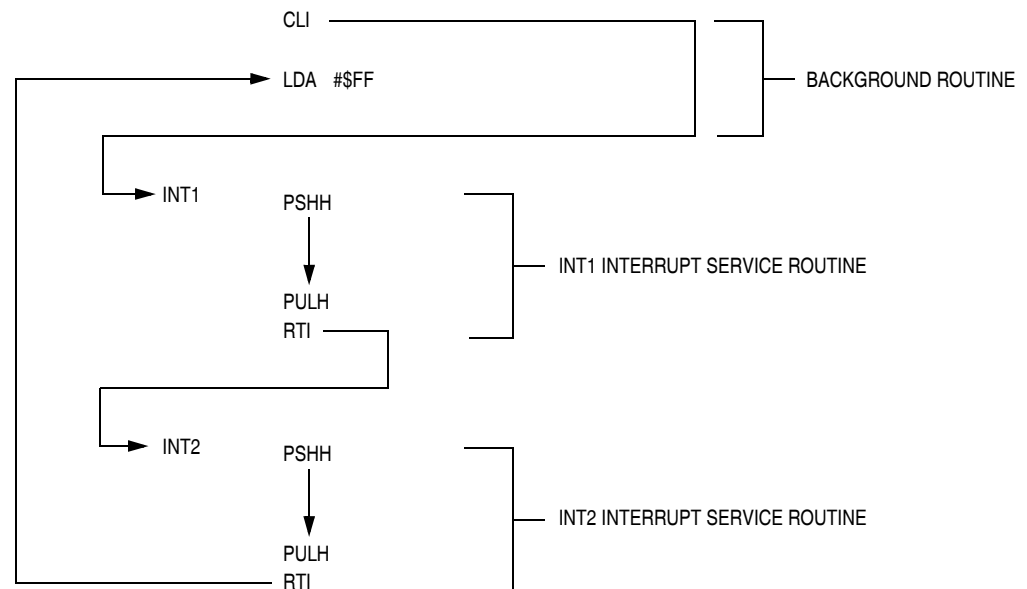


Figure 7-11. Interrupt Recognition Example

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

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

9.3 Features

Features of the monitor ROM include the following:

- Normal user-mode pin functionality
- One pin dedicated to serial communication between monitor ROM and host computer
- Standard mark/space non-return-to-zero (NRZ) communication with host computer
- Execution of code in RAM or FLASH
- FLASH memory security feature¹
- FLASH memory programming interface
- 960 bytes monitor ROM code size
- Monitor mode entry without high voltage, $V_{DD} + V_{HI}$, if reset vector is blank (\$FFFE and \$FFFF contain \$FF)
- Standard monitor mode entry if high voltage, $V_{DD} + V_{HI}$, is applied to $\overline{IRQ1}$

9.4 Functional Description

The monitor ROM receives and executes commands from a host computer. **Figure 9-1** shows a example circuit used to enter monitor mode and communicate with a host computer via a standard RS-232 interface.

Simple monitor commands can access any memory address. In monitor mode, the MCU can execute host-computer code in RAM while most MCU pins retain normal operating mode functions. All communication between the host computer and the MCU is through the PTB0 pin. A level-shifting and multiplexing interface is required between PTB0 and the host computer. PTB0 is used in a wired-OR configuration and requires a pull-up resistor.

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

Table 9-2 is a summary of the vector differences between user mode and monitor mode.

Table 9-2. Monitor Mode Vector Differences

Modes	Functions						
	COP	Reset Vector High	Reset Vector Low	Break Vector High	Break Vector Low	SWI Vector High	SWI Vector Low
User	Enabled	\$FFFE	\$FFFF	\$FFFC	\$FFFD	\$FFFC	\$FFFD
Monitor	Disabled ⁽¹⁾	\$FEFE	\$FEFF	\$FEFC	\$FEFD	\$FEFC	\$FEFD
Notes: 1. If the high voltage ($V_{DD} + V_{HI}$) is removed from the $\overline{IRQ1}$ pin or the \overline{RST} pin, the SIM asserts its COP enable output. The COP is a mask option enabled or disabled by the COPD bit in the configuration register.							

When the host computer has completed downloading code into the MCU RAM, the host then sends a RUN command, which executes an RTI, which sends control to the address on the stack pointer.

9.4.2 Baud Rate

The communication baud rate is dependant on oscillator frequency. The state of PTB3 also affects baud rate if entry to monitor mode is by $\overline{IRQ1} = V_{DD} + V_{HI}$. When PTB3 is high, the divide by ratio is 1024. If the PTB3 pin is at logic zero upon entry into monitor mode, the divide by ratio is 512.

Table 9-3. Monitor Baud Rate Selection

Monitor Mode Entry By:	Input Clock Frequency	PTB3	Baud Rate
$\overline{IRQ1} = V_{DD} + V_{HI}$	4.9152 MHz	0	9600 bps
	9.8304 MHz	1	9600 bps
	4.9152 MHz	1	4800 bps
Blank reset vector, $\overline{IRQ1} = V_{DD}$	9.8304 MHz	X	9600 bps
	4.9152 MHz	X	4800 bps

Section 10. Timer Interface Module (TIM)

10.1 Contents

10.2	Introduction	116
10.3	Features	116
10.4	Pin Name Conventions	116
10.5	Functional Description	117
10.5.1	TIM Counter Prescaler	119
10.5.2	Input Capture	119
10.5.3	Output Compare	119
10.5.3.1	Unbuffered Output Compare	120
10.5.3.2	Buffered Output Compare	120
10.5.4	Pulse Width Modulation (PWM)	121
10.5.4.1	Unbuffered PWM Signal Generation	122
10.5.4.2	Buffered PWM Signal Generation	123
10.5.4.3	PWM Initialization	124
10.6	Interrupts	125
10.7	Wait Mode	125
10.8	TIM During Break Interrupts	126
10.9	I/O Signals	126
10.10	I/O Registers	127
10.10.1	TIM Status and Control Register (TSC)	127
10.10.2	TIM Counter Registers (TCNTH:TCNTL)	129
10.10.3	TIM Counter Modulo Registers (TMODH:TMODL)	130
10.10.4	TIM Channel Status and Control Registers (TSC0:TSC1)	131
10.10.5	TIM Channel Registers (TCH0H/L:TCH1H/L)	135

\$0029	TIM Channel 1 Register High (TCH1H)	Read:	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	
		Write:									
		Reset:	Indeterminate after reset								
\$002A	TIM Channel 1 Register Low (TCH1L)	Read:	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
		Write:									
		Reset:	Indeterminate after reset								
			= Unimplemented								

Figure 10-2. TIM I/O Register Summary

10.5.1 TIM Counter Prescaler

The TIM clock source is one of the seven prescaler outputs. 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.

10.5.2 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 CPU interrupt requests.

10.5.3 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.

overflow occurs before the clearing sequence is complete, then writing logic zero 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 logic one 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.*

TRST — TIM Reset Bit

Setting this write-only bit resets the TIM counter and the TIM prescaler. Setting TRST has no effect on any other registers. Counting resumes from \$0000. TRST is cleared automatically after the TIM counter is reset and always reads as logic zero. Reset clears the TRST bit.

1 = Prescaler and TIM counter cleared

0 = No effect

NOTE: *Setting the TSTOP and TRST bits simultaneously stops the TIM counter at a value of \$0000.*

PS[2:0] — Prescaler Select Bits

These read/write bits select one of the seven prescaler outputs as the input to the TIM counter as [Table 10-2](#) shows. Reset clears the PS[2:0] bits.

Section 12. I/O Ports

12.1 Contents

12.2	Introduction	147
12.3	Port A	148
12.4	Port A Data Register (PTA)	149
12.4.1	Data Direction Register A (DDRA)	150
12.4.2	Port A Input Pull-up Enable Register (PTAPUE)	151
12.5	Port B	153
12.5.1	Port B Data Register (PTB)	153
12.5.2	Data Direction Register B (DDRB)	153
12.6	Port D	155
12.6.1	Port D Data Register (PTD)	155
12.6.2	Data Direction Register D (DDRD)	156
12.6.3	Port D Control Register (PDCR)	157

12.2 Introduction

Twenty three bidirectional input-output (I/O) pins form three parallel ports. All I/O pins are programmable as inputs or outputs.

NOTE: *Connect any unused I/O pins to an appropriate logic level, either V_{DD} or V_{SS} . Although the I/O ports do not require termination for proper operation, termination reduces excess current consumption and the possibility of electrostatic damage.*

12.4.1 Data Direction Register A (DDRA)

Data direction register A determines whether each port A pin is an input or an output. Writing a logic one to a DDRA bit enables the output buffer for the corresponding port A pin; a logic zero disables the output buffer.

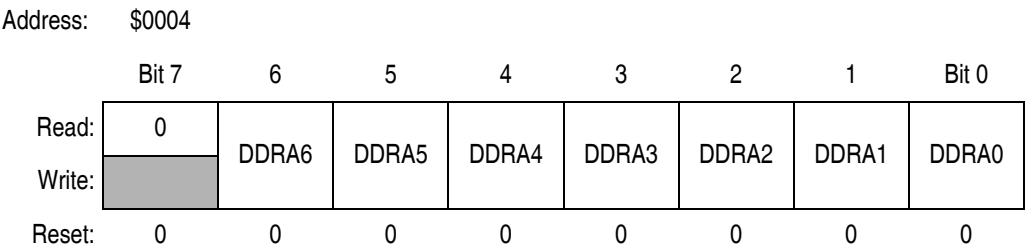


Figure 12-3. Data Direction Register A (DDRA)

DDRA[6:0] — Data Direction Register A Bits

These read/write bits control port A data direction. Reset clears DDRA[6:0], 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 12-4 shows the port A I/O logic.

Address: \$000D

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	PTA6EN	PTAPUE	PTAPUE	PTAPUE	PTAPUE	PTAPUE	PTAPUE	PTAPUE
Write:		6	5	4	3	2	2	0
Reset:	0	0	0	0	0	0	0	0

Figure 12-5. Port A Input Pull-up Enable Register (PTAPUE)

PTA6EN — Enable PTA6 on OSC2

This read/write bit configures the OSC2 pin function when RC oscillator option is selected. This bit has no effect for X-tal oscillator option.

- 1 = OSC2 pin configured for PTA6 I/O, and has all the interrupt and pull-up functions.
- 0 = OSC2 pin outputs the RC oscillator clock (RCCLK)

PTAPUE[6:0] — Port A Input Pull-up Enable bits

These read/write bits are software programmable to enable pull-up devices on port A pins

- 1 = Corresponding port A pin configured to have internal pull if its DDRA bit is set to 0
- 0 = Pull-up device is disconnected on the corresponding port A pin regardless of the state of its DDRA bit.

Table 12-1 summarizes the operation of the port B pins.

Table 12-1. Port A Pin Functions

PTAPUE Bit	DDRA Bit	PTA Bit	I/O Pin Mode	Accesses to DDRB	Accesses to PTB	
				Read/Write	Read	Write
1	0	X ⁽¹⁾	Input, V _{DD} ⁽²⁾	DDRA6-DDRA0	Pin	PTA6-PTA0 ⁽³⁾
0	0	X	Input, Hi-Z ⁽⁴⁾	DDRA6-DDRA0	Pin	PTA6-PTA0 ⁽³⁾
X	1	X	Output	DDRA6-DDRA0	PTA6-PTA0	PTA6-PTA0

1. X = Don't care.
2. I/O pin pulled to V_{DD} by internal pull-up.
3. Writing affects data register, but does not affect input.
4. Hi-Z = High Impedence

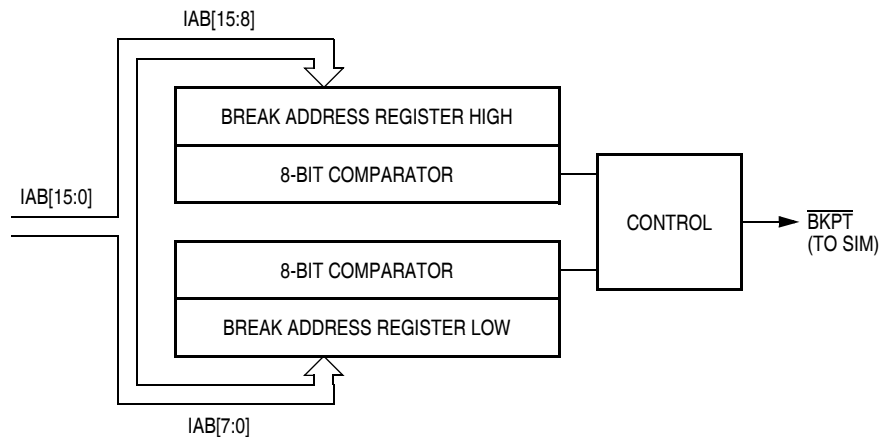


Figure 17-1. Break Module Block Diagram

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
\$FE00	Break Status Register (BSR)	Read:							SBSW	
		Write:	R	R	R	R	R	R	See note	R
		Reset:	0							
\$FE03	Break Flag Control Register (BFCR)	Read:								
		Write:	BCFE	R	R	R	R	R	R	R
		Reset:	0							
\$FE0C	Break Address High Register (BRKH)	Read:								
		Write:	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
		Reset:	0	0	0	0	0	0	0	0
\$FE0D	Break Address low Register (BRKL)	Read:								
		Write:	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
		Reset:	0	0	0	0	0	0	0	0
\$FE0E	Break Status and Control Register (BRKSCR)	Read:								
		Write:	BRKE	BRKA						
		Reset:	0	0	0	0	0	0	0	0

Note: Writing a logic 0 clears SBSW.

 = Unimplemented R = Reserved

Figure 17-2. Break I/O Register Summary