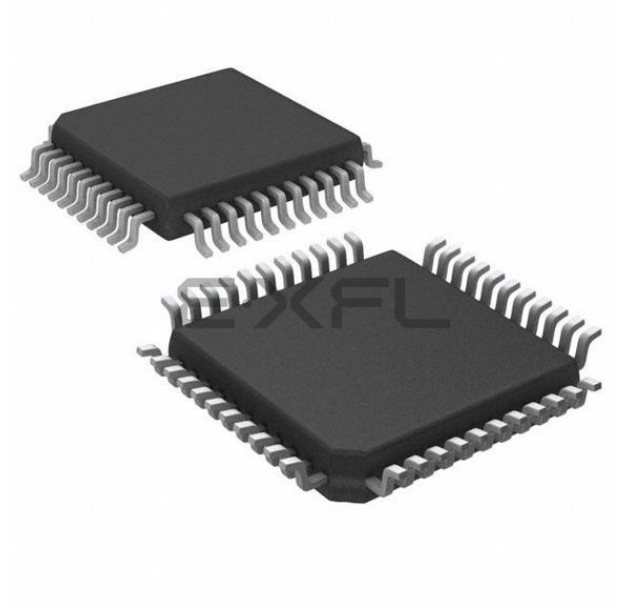


Welcome to E-XFL.COM

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"

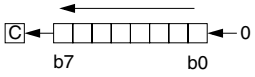
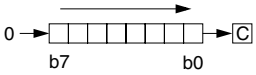
Details	
Product Status	Not For New Designs
Core Processor	HC08
Core Size	8-Bit
Speed	8MHz
Connectivity	I²C, IRSCI, SCI, SPI
Peripherals	LED, LVD, POR, PWM
Number of I/O	32
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-QFP
Supplier Device Package	44-QFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc908ap16acfbe

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$004D	MMIIC Data Receive Register (MMDRR)	Read:	MMRD7	MMRD6	MMRD5	MMRD4	MMRD3	MMRD2	MMRD1	MMRD0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$004E	MMIIC CRC Data Register (MMCRDR)	Read:	MMCRCD7	MMCRCD6	MMCRCD5	MMCRCD4	MMCRCD3	MMCRCD2	MMCRCD1	MMCRCD0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$004F	MMIIC Frequency Divider Register (MMFDR)	Read:	0	0	0	0	0	MMBR2	MMBR1	MMBR0
		Write:								
		Reset:	0	0	0	0	0	1	0	0
$0050	Reserved	Read:	R	R	R	R	R	R	R	R
		Write:								
		Reset:								
$0051	Timebase Control Register (TBCR)	Read:	TBIF	TBR2	TBR1	TBR0	0	TBIE	TBON	R
		Write:					TACK			
		Reset:	0	0	0	0	0	0	0	0
$0052	Unimplemented	Read:								
		Write:								
		Reset:								
$0053	Unimplemented	Read:								
		Write:								
		Reset:								
$0054	Unimplemented	Read:								
		Write:								
		Reset:								
$0055	Unimplemented	Read:								
		Write:								
		Reset:								
$0056	Unimplemented	Read:								
		Write:								
		Reset:								
$0057	ADC Status and Control Register (ADSCR)	Read:	COCO	AIEN	ADCO	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0
		Write:								
		Reset:	0	0	0	1	1	1	1	1
$0058	ADC Clock Control Register (ADICLK)	Read:	ADIV2	ADIV1	ADIV0	ADICLK	MODE1	MODE0	0	0
		Write:								R
		Reset:	0	0	0	0	0	0	0	0
$0059	ADC Data Register High 0 (ADRH0)	Read:	ADx	ADx	ADx	ADx	ADx	ADx	ADx	ADx
		Write:	R	R	R	R	R	R	R	R
		Reset:	0	0	0	0	0	0	0	0

U = Unaffected X = Indeterminate = Unimplemented R = Reserved

Figure 2-2. Control, Status, and Data Registers (Sheet 7 of 9)

Table 4-1. Instruction Set Summary

Source Form	Operation	Description	Effect on CCR						Address Mode	Opcode	Operand	Cycles
			V	H	I	N	Z	C				
LDX #*opr* LDX *opr* LDX *opr* LDX *opr*,X LDX *opr*,X LDX ,X LDX *opr*,SP LDX *opr*,SP	Load X from M	X ← (M)	0	–	–	↕	↕	–	IMM DIR EXT IX2 IX1 IX SP1 SP2	AE BE CE DE EE FE 9EEE 9EDE	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5
LSL *opr* LSLA LSLX LSL *opr*,X LSL ,X LSL *opr*,SP	Logical Shift Left (Same as ASL)		↕	–	–	↕	↕	↕	DIR INH INH IX1 IX SP1	38 48 58 68 78 9E68	dd ff ff	4 1 1 4 3 5
LSR *opr* LSRA LSR*X* LSR *opr*,X LSR ,X LSR *opr*,SP	Logical Shift Right		↕	–	–	0	↕	↕	DIR INH INH IX1 IX SP1	34 44 54 64 74 9E64	dd ff ff	4 1 1 4 3 5
MOV *opr*,*opr* MOV *opr*,X+ MOV #*opr*,*opr* MOV X+,*opr*	Move	$(M)_{Destination} \leftarrow (M)_{Source}$ 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 *opr* NEGA NEGX NEG *opr*,X NEG ,X NEG *opr*,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 #*opr* ORA *opr* ORA *opr* ORA *opr*,X ORA *opr*,X ORA ,X ORA *opr*,SP ORA *opr*,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 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

NOTE

Route V_{SSA} carefully for maximum noise immunity and place bypass capacitors as close as possible to the package.

6.4.4 Oscillator Output Frequency Signal (CGMXCLK)

CGMXCLK is the oscillator output signal. It runs at the full speed of the oscillator, and is generated directly from the crystal oscillator circuit, the RC oscillator circuit, or the internal oscillator circuit.

6.4.5 CGM Reference Clock (CGMRCLK)

CGMRCLK is a buffered version of CGMXCLK, this clock is the reference clock for the phase-locked-loop circuit.

6.4.6 CGM VCO Clock Output (CGMVCLK)

CGMVCLK is the clock output from the VCO.

6.4.7 CGM Base Clock Output (CGMOUT)

CGMOUT is the clock output of the CGM. This signal goes to the SIM, which generates the MCU clocks. CGMOUT is a 50 percent duty cycle clock running at twice the bus frequency. CGMOUT is software programmable to be either the oscillator output, CGMXCLK, divided by two or the divided VCO clock, CGMPCLK, divided by two.

6.4.8 CGM CPU Interrupt (CGMINT)

CGMINT is the interrupt signal generated by the PLL lock detector.

6.5 CGM Registers

The following registers control and monitor operation of the CGM:

- PLL control register (PCTL)
  (See 6.5.1 PLL Control Register.)
- PLL bandwidth control register (PBWC)
  (See 6.5.2 PLL Bandwidth Control Register.)
- PLL multiplier select registers (PMSH and PMSL)
  (See 6.5.3 PLL Multiplier Select Registers.)
- PLL VCO range select register (PMRS)
  (See 6.5.4 PLL VCO Range Select Register.)
- PLL reference divider select register (PMDS)
  (See 6.5.5 PLL Reference Divider Select Register.)

7.7.1 SIM Break Status Register

The SIM break status register (SBSR) contains a flag to indicate that a break caused an exit from stop mode or wait mode.

Address: $FE00

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

Note: Writing a logic 0 clears SBSW. R = Reserved

Figure 7-20. SIM Break Status Register (SBSR)

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.

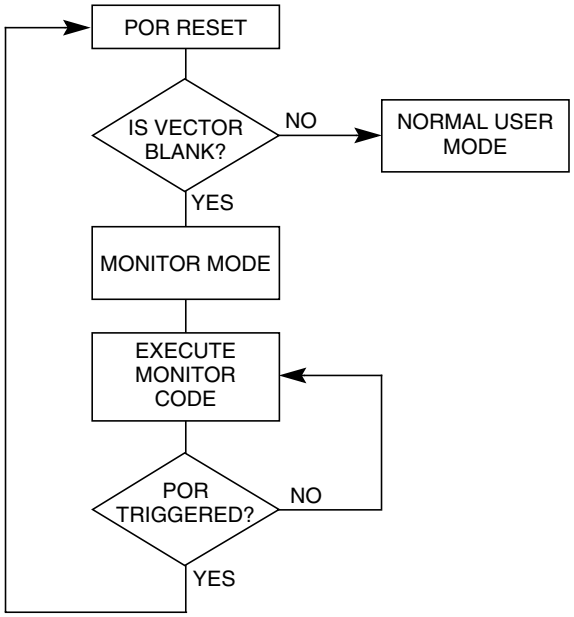


Figure 8-2. Low-Voltage Monitor Mode Entry Flowchart

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 $\overline{RST}$ low will not exit monitor mode in this situation.

Table 8-2 summarizes the differences between user mode and monitor mode vectors.

Table 8-2. Mode Differences (Vectors)

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

8.3.2 Data Format

Communication with the monitor ROM is in standard non-return-to-zero (NRZ) mark/space data format. Transmit and receive baud rates must be identical.

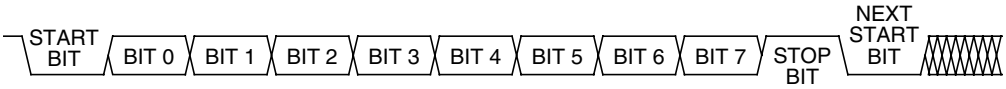


Figure 8-3. Monitor Data Format

```
                ORG     RAM
        :
FILE_PTR:
BUS_SPD         DS.B    1       ; Indicates 4x bus frequency
DATASIZE        DS.B    1       ; Data size to be programmed
START_ADDR      DS.W    1       ; FLASH start address
DATAARRAY       DS.B    64      ; Reserved data array

PRGRNGE         EQU     $FC34
FLASH_START     EQU     $EE00

                ORG     FLASH
INITIALISATION:
        MOV     #20,    BUS_SPD
        MOV     #64,    DATASIZE
        LDHX    #FLASH_START
        STHX    START_ADDR
        RTS
MAIN:
        BSR     INITIALISATION
        :
        :
        LDHX    #FILE_PTR
        JSR     PRGRNGE
```

8.5.2 ERARNGE

ERARNGE is used to erase a range of locations in FLASH.

Table 8-12. ERARNGE Routine

Routine Name	ERARNGE
Routine Description	Erase a page or the entire array
Calling Address	$FCE4
Stack Used	9 bytes
Data Block Format	Bus speed (BUS_SPD) Data size (DATASIZE) Starting address (ADDRH) Starting address (ADDRL)

There are two sizes of erase ranges: a page or the entire array. The ERARNGE will erase the page (512 consecutive bytes) in FLASH specified by the address ADDRH:ADDRL. This address can be any address within the page. Calling ERARNGE with ADDRH:ADDRL equal to $FFFF will erase the entire FLASH array (mass erase). Therefore, care must be taken when calling this routine to prevent an accidental mass erase.

The ERARNGE routine do not use a data array. The DATASIZE byte is a dummy byte that is also not used.

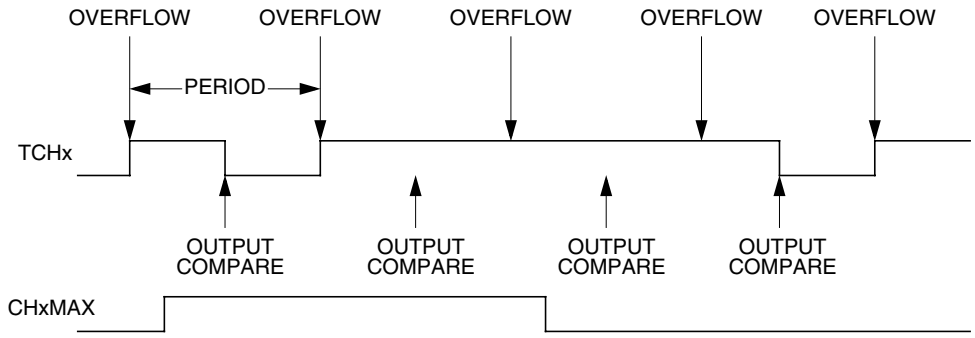


Figure 9-11. CHxMAX Latency

9.9.5 TIM Channel Registers

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

In input capture mode (MSxB:MSxA = 0:0), reading the high byte of the TIM channel x registers (TCHxH) inhibits input captures until the low byte (TCHxL) is read.

In output compare mode (MSxB:MSxA ≠ 0:0), writing to the high byte of the TIM channel x registers (TCHxH) inhibits output compares until the low byte (TCHxL) is written.

Address: T1CH0H, $0026 and T2CH0H, $0031

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

Figure 9-12. TIM Channel 0 Register High (TCH0H)

Address: T1CH0L, $0027 and T2CH0L $0032

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

Figure 9-13. TIM Channel 0 Register Low (TCH0L)

Address: T1CH1H, $0029 and T2CH1H, $0034

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

Figure 9-14. TIM Channel 1 Register High (TCH1H)

11.4 Functional Description

Figure 11-2 shows the structure of the SCI module. The SCI allows full-duplex, asynchronous, NRZ serial communication among the MCU and remote devices, including other MCUs. The transmitter and receiver of the SCI operate independently, although they use the same baud rate generator. During normal operation, the CPU monitors the status of the SCI, writes the data to be transmitted, and processes received data.

The baud rate clock source for the SCI can be selected via the configuration bit, SCIBDSRC, of the CONFIG2 register ($001D).

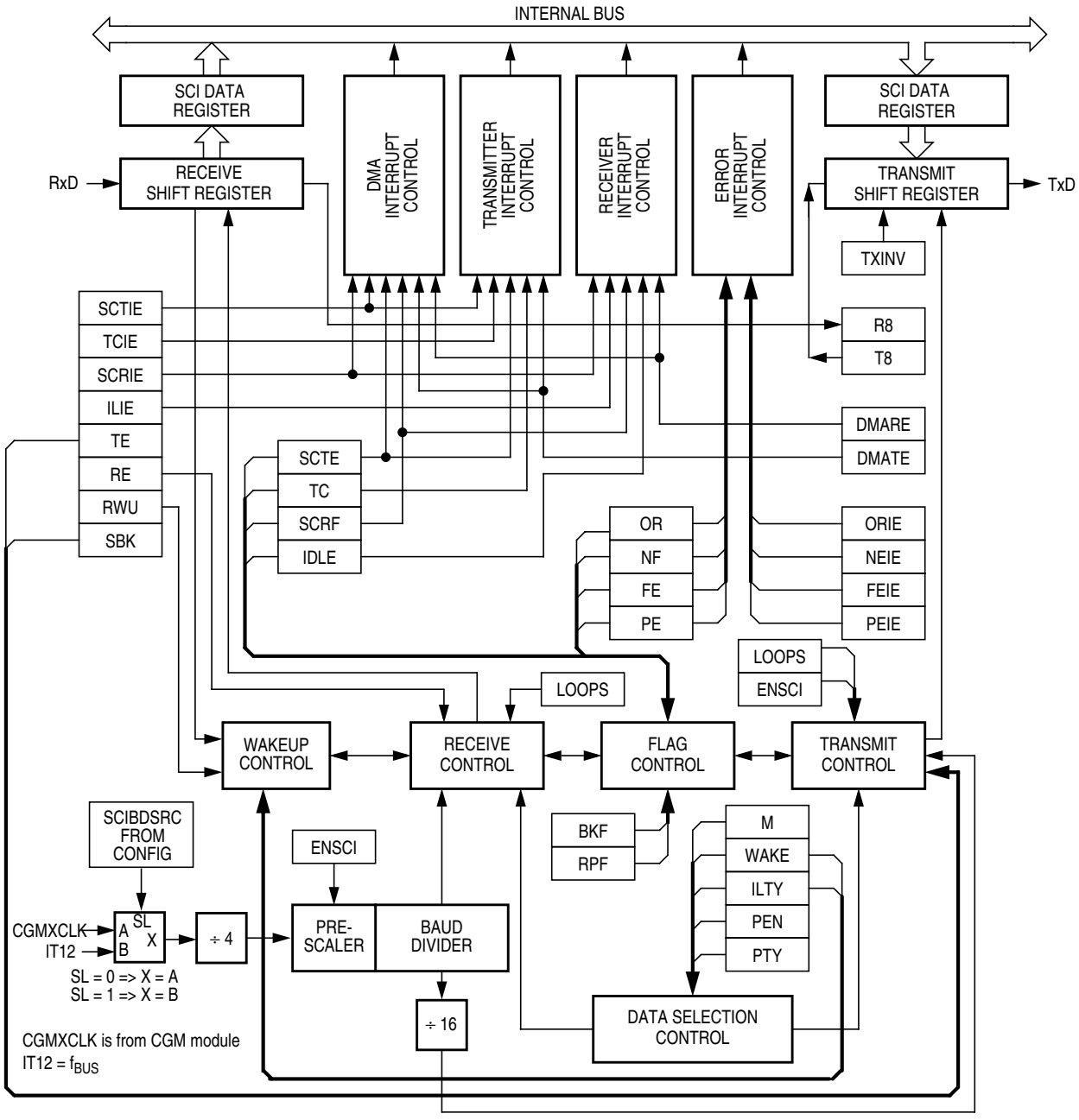


Figure 11-2. SCI Module Block Diagram

$$\left|\frac{170 - 176}{170}\right| \times 100 = 3.53\%$$

12.5.3.6 Receiver Wakeup

So that the MCU can ignore transmissions intended only for other receivers in multiple-receiver systems, the receiver can be put into a standby state. Setting the receiver wakeup bit, RWU, in IRSCC2 puts the receiver into a standby state during which receiver interrupts are disabled.

Depending on the state of the WAKE bit in IRSCC1, either of two conditions on the RxD pin can bring the receiver out of the standby state:

- Address mark — An address mark is a logic 1 in the most significant bit position of a received character. When the WAKE bit is set, an address mark wakes the receiver from the standby state by clearing the RWU bit. The address mark also sets the SCI receiver full bit, SCRF. Software can then compare the character containing the address mark to the user-defined address of the receiver. If they are the same, the receiver remains awake and processes the characters that follow. If they are not the same, software can set the RWU bit and put the receiver back into the standby state.
- Idle input line condition — When the WAKE bit is clear, an idle character on the RxD pin wakes the receiver from the standby state by clearing the RWU bit. The idle character that wakes the receiver does not set the receiver idle bit, IDLE, or the SCI receiver full bit, SCRF. The idle line type bit, ILTY, determines whether the receiver begins counting logic 1s as idle character bits after the start bit or after the stop bit.

NOTE

Clearing the WAKE bit after the RxD pin has been idle may cause the receiver to wake up immediately.

12.5.3.7 Receiver Interrupts

The following sources can generate CPU interrupt requests from the SCI receiver:

- SCI receiver full (SCRF) — The SCRF bit in IRSCS1 indicates that the receive shift register has transferred a character to the IRSCDR. SCRF can generate a receiver interrupt request. Setting the SCI receive interrupt enable bit, SCRIE, in IRSCC2 enables the SCRF bit to generate receiver CPU interrupts.
- Idle input (IDLE) — The IDLE bit in IRSCS1 indicates that 10 or 11 consecutive logic 1s shifted in from the RxD pin. The idle line interrupt enable bit, ILIE, in IRSCC2 enables the IDLE bit to generate CPU interrupt requests.

12.5.3.8 Error Interrupts

The following receiver error flags in IRSCS1 can generate CPU interrupt requests:

- Receiver overrun (OR) — The OR bit indicates that the receive shift register shifted in a new character before the previous character was read from the IRSCDR. The previous character remains in the IRSCDR, and the new character is lost. The overrun interrupt enable bit, ORIE, in IRSCC3 enables OR to generate SCI error CPU interrupt requests.
- Noise flag (NF) — The NF bit is set when the SCI detects noise on incoming data or break characters, including start, data, and stop bits. The noise error interrupt enable bit, NEIE, in IRSCC3 enables NF to generate SCI error CPU interrupt requests.

This read/write bit controls the timing relationship between the serial clock and SPI data. (See Figure 13-4 and Figure 13-6.) To transmit data between SPI modules, the SPI modules must have identical CPHA values. When CPHA = 0, the $\overline{SS}$ pin of the slave SPI module must be set to logic 1 between bytes. (See Figure 13-12.) Reset sets the CPHA bit.

MMRW — MMIIC Master Read/Write

This bit is transmitted out as bit 0 of the calling address when the module sets the MMAST bit to enter master mode. The MMRW bit determines the transfer direction of the data bytes that follows. When it is "1", the module is in master receive mode. When it is "0", the module is in master transmit mode. Reset clears this bit.

1 = Master mode receive
0 = Master mode transmit

MMCRCEF — MMIIC CRC Error Flag

This flag is set when a CRC error is detected, and cleared when no CRC error is detected. The MMCRCEF is only meaningful after receiving a PEC data. This flag is unaffected by reset.

1 = CRC error detected on PEC byte
0 = No CRC error detected on PEC byte

14.6.4 MMIIC Status Register (MMSR)

Address: $004B

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	MMRXIF	MMTXIF	MMATCH	MMSRW	MMRXAK	MMCRCBF	MMTXBE	MMRXBF
Write:	0	0						
Reset:	0	0	0	0	1	0	1	0

= Unimplemented

Figure 14-7. MMIIC Status Register (MMSR)

MMRXIF — MMIIC Receive Interrupt Flag

This flag is set after the data receive register (MMDRR) is loaded with a new received data. Once the MMDRR is loaded with received data, no more received data can be loaded to the MMDRR register until the CPU reads the data from the MMDRR to clear MMRXBF flag. MMRXIF generates an interrupt request to CPU if the MMIEN bit in MMCR is also set. This bit is cleared by writing "0" to it or by reset; or when the MMEN = 0.

1 = New data in data receive register (MMDRR)
0 = No data received

MMTXIF — MMIIC Transmit Interrupt Flag

This flag is set when data in the data transmit register (MMDTR) is downloaded to the output circuit, and that new data can be written to the MMDTR. MMTXIF generates an interrupt request to CPU if the MMIEN bit in MMCR is also set. This bit is cleared by writing "0" to it or when the MMEN = 0.

1 = Data transfer completed
0 = Data transfer in progress

MMATCH — MMIIC Address Match Flag

This flag is set when the received data in the data receive register (MMDRR) is a calling address which matches with the address or its extended addresses (MMEXTAD = 1) specified in the address register (MMADR). The MMATCH flag is set at the 9th clock of the calling address and will be cleared on the 9th clock of the next receiving data. Note: slave transmits do not clear MMATCH.

1 = Received address matches MMADR
0 = Received address does not match

14.8.4 Write Byte/Word

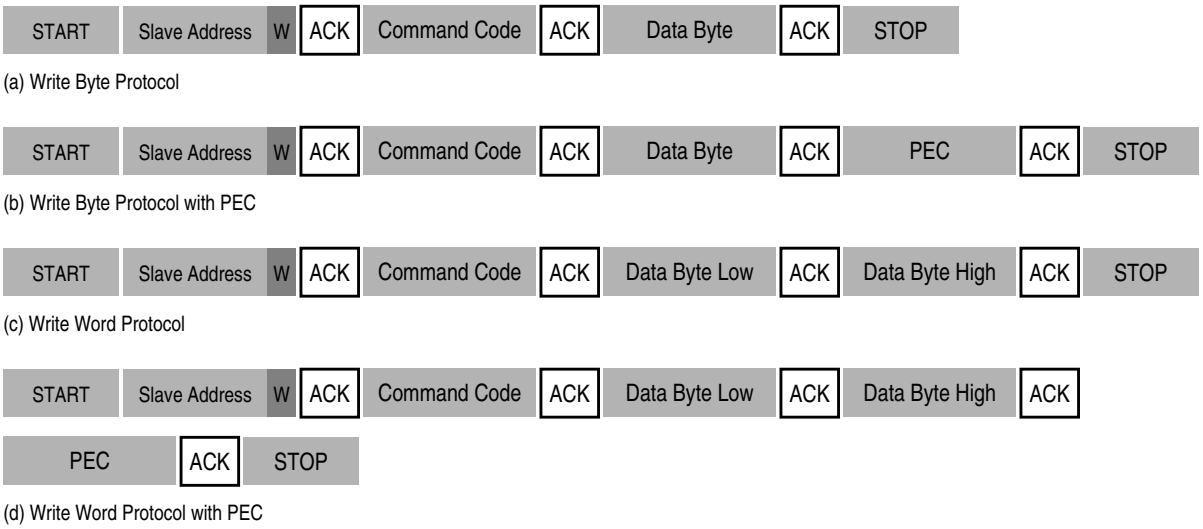


Figure 14-16. Write Byte/Word

14.8.5 Read Byte/Word

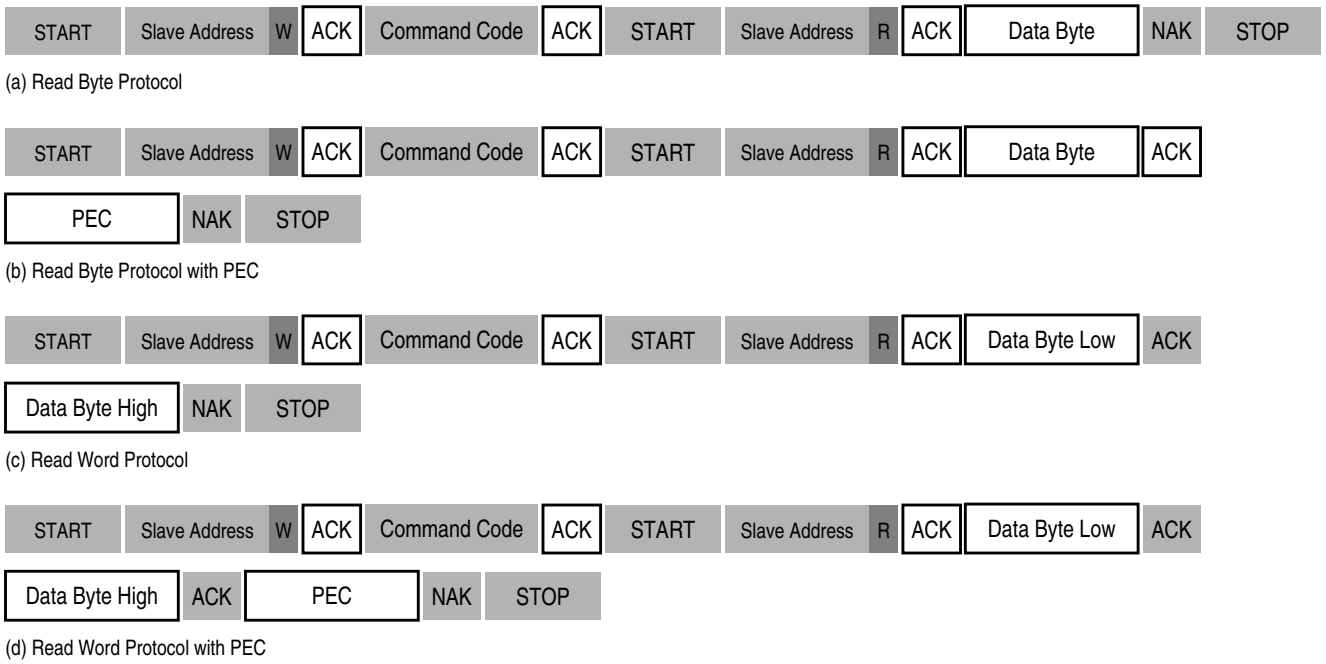


Figure 14-17. Read Byte/Word

Chapter 17
External Interrupt (IRQ)

17.1 Introduction

The external interrupt (IRQ) module provides two maskable interrupt inputs: $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$.

17.2 Features

Features of the IRQ module include:

- A dedicated external interrupt pin, $\overline{\text{IRQ1}}$
- An external interrupt pin shared with a port pin, PTC0/$\overline{\text{IRQ2}}$
- Separate IRQ interrupt control bits for $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$
- Programmable edge-only or edge and level interrupt sensitivity
- Automatic interrupt acknowledge
- Internal pullup resistor, with disable option on $\overline{\text{IRQ2}}$

NOTE

References to either IRQ1 or IRQ2 may be made in the following text by omitting the IRQ number. For example, IRQF may refer generically to IRQ1F and IRQ2F, and IMASK may refer to IMASK1 and IMASK2.

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$001C	IRQ2 Status and Control Register (INTSCR2)	Read:	0	PUC0ENB	0	0	IRQ2F	0	IMASK2	MODE2
		Write:						ACK2		
		Reset:	0	0	0	0	0	0	0	0
$001E	IRQ1 Status and Control Register (INTSCR1)	Read:	0	0	0	0	IRQ1F	0	IMASK1	MODE1
		Write:						ACK1		
		Reset:	0	0	0	0	0	0	0	0
			= Unimplemented							

Figure 17-1. External Interrupt I/O Register Summary

17.3 Functional Description

A logic 0 applied to the external interrupt pin can latch a CPU interrupt request. Figure 17-2 and Figure 17-3 shows the structure of the IRQ module.

Interrupt signals on the $\overline{\text{IRQ}}$ pin are latched into the IRQ latch. An interrupt latch remains set until one of the following actions occurs:

- Vector fetch — A vector fetch automatically generates an interrupt acknowledge signal that clears the latch that caused the vector fetch.

NOTE

When using the level-sensitive interrupt trigger, avoid false interrupts by masking interrupt requests in the interrupt routine.

The $\overline{\text{IRQ1}}$ pin has a permanent internal pullup device connected, while the $\overline{\text{IRQ2}}$ pin has an optional pullup device that can be enabled or disabled by the PUC0ENB bit in the INTSCR2 register.

17.5 IRQ Module During Break Interrupts

The BCFE bit in the SIM break flag control register (SBFCR) enables software to clear the latch during the break state. (See Chapter 21 Break Module (BRK).)

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

To protect CPU interrupt flags during the break state, write a logic 0 to the BCFE bit. With BCFE at logic 0 (its default state), writing to the ACK bit in the IRQ status and control register during the break state has no effect on the IRQ interrupt flags.

17.6 IRQ Registers

Each IRQ is controlled and monitored by an status and control register.

- IRQ1 Status and Control Register — $001E
- IRQ2 Status and Control Register — $001C

17.6.1 IRQ1 Status and Control Register

The IRQ1 status and control register (INTSCR1) controls and monitors operation of IRQ1. The INTSCR1 has the following functions:

- Shows the state of the IRQ1 flag
- Clears the IRQ1 latch
- Masks IRQ1 interrupt request
- Controls triggering sensitivity of the $\overline{\text{IRQ1}}$ interrupt pin

Address: $001E

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	IRQ1F	0	IMASK1	MODE1
Write:						ACK1		
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 17-4. IRQ1 Status and Control Register (INTSCR1)

IRQ1F — IRQ1 Flag Bit

This read-only status bit is high when the IRQ1 interrupt is pending.

1 = $\overline{\text{IRQ1}}$ interrupt pending
0 = $\overline{\text{IRQ1}}$ interrupt not pending

ACK1 — IRQ1 Interrupt Request Acknowledge Bit

Writing a logic 1 to this write-only bit clears the IRQ1 latch. ACK1 always reads as logic 0. Reset clears ACK1.

Table 22-4. DC Electrical Characteristics (5V)

Characteristic[1]	Symbol	Min	Typ[2]	Max	Unit
Low-voltage inhibit, trip voltage2	V_{TRIPF2}	2.25	2.45	2.65	V
V_{REG}[10]	V_{REG}	2.25	2.50	2.75	V

1. V_{DD} = 4.5 to 5.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H, unless otherwise noted.
2. Typical values reflect average measurements at midpoint of voltage range, 25 °C only.
3. Run (operating) I_{DD} measured using external 32MHz clock to OSC1; all inputs 0.2 V from rail; no dc loads; less than 100pF on all outputs; C_L = 20 pF on OSC2; all ports configured as inputs; OSC2 capacitance linearly affects run I_{DD}; measured with all modules enabled.
4. Wait I_{DD} measured using external 32MHz to OSC1; all inputs 0.2 V from rail; no dc loads; less than 100 pF on all outputs. C_L = 20 pF on OSC2; all ports configured as inputs; OSC2 capacitance linearly affects wait I_{DD}.
5. STOP I_{DD} measured using external 8MHz clock to OSC1; no port pins sourcing current.
6. STOP I_{DD} measured with OSC1 grounded; no port pins sourcing current.
7. Maximum is highest voltage that POR is guaranteed. The rearm voltage is triggered by V_{REG}.
8. If minimum V_{DD} is not reached before the internal POR reset is released, $\overline{RST}$ must be driven low externally until minimum V_{DD} is reached.
9. R_{PU1} and R_{PU2} are measured at V_{DD} = 5.0V
10. Measured from $V_{DD} = V_{TRIPF1}$ (Min) to 5.5 V.

22.6 5V Control Timing

Table 22-5. Control Timing (5V)

Characteristic[1]	Symbol	Min	Max	Unit
Internal operating frequency[2]	f_{OP}	—	8	MHz
$\overline{RST}$ input pulse width low[3]	t_{RL}	100	—	ns

1. V_{DD} = 4.5 to 5.5 Vdc, V_{SS} = 0 Vdc; timing shown with respect to 20% V_{DD} and 70% V_{DD}, unless otherwise noted.
2. Some modules may require a minimum frequency greater than dc for proper operation; see appropriate table for this information.
3. Minimum pulse width reset is guaranteed to be recognized. It is possible for a smaller pulse width to cause a reset.

Chapter 23
Mechanical Specifications

23.1 Introduction

This section gives the dimensions for:

- 48-pin plastic low-profile quad flat pack (case #932)
- 44-pin plastic quad flat pack (case #824A)
- 42-pin shrink dual in-line package (case #858)