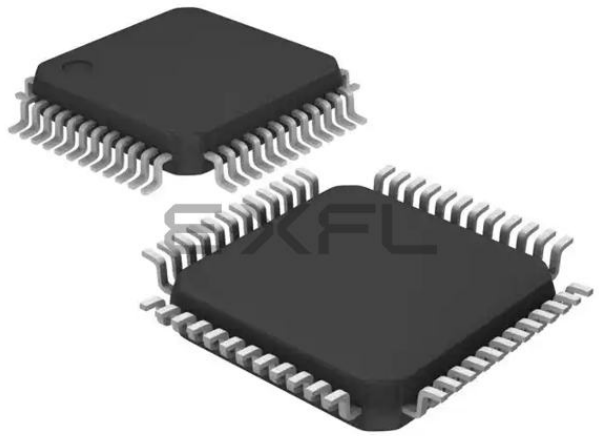

Welcome to E-XFL.COM

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	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K 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	48-LQFP
Supplier Device Package	48-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc908ap32acfae

Table 1-2. Pin Functions

PIN NAME	PIN DESCRIPTION	IN/OUT	VOLTAGE LEVEL
PTB0/SDA PTB1/SCL PTB2/TxD PTB3/RxD PTB4/T1CH0 PTB5/T1CH1 PTB6/T2CH0 PTB7/T2CH1	8-bit general purpose I/O port; PTB0–PTB3 are open drain when configured as output. PTB4–PTB7 have schmitt trigger inputs.	In/Out	V_{DD}
	PTB0 as SDA of MMIIC.	In/Out	V_{DD}
	PTB1 as SCL of MMIIC.	In/Out	V_{DD}
	PTB2 as TxD of SCI; open drain output.	Out	V_{DD}
	PTB3 as RxD of SCI.	In	V_{DD}
	PTB4 as T1CH0 of TIM1.	In/Out	V_{DD}
	PTB5 as T1CH1 of TIM1.	In/Out	V_{DD}
	PTB6 as T2CH0 of TIM2.	In/Out	V_{DD}
	PTB7 as T2CH1 of TIM2.	In/Out	V_{DD}
PTC0/$\overline{IRQ2}$ PTC1 PTC2/MISO PTC3/MOSI PTC4/$\overline{SS}$ PTC5/SPSCK PTC6/SCTxD PTC7/SCRxD	8-bit general purpose I/O port; PTC6 and PTC7 are open drain when configured as output.	In/Out	V_{DD}
	PTC0 is shared with $\overline{IRQ2}$ and has schmitt trigger input.	In	V_{DD}
	PTC2 as MISO of SPI.	In	V_{DD}
	PTC3 as MOSI of SPI.	Out	V_{DD}
	PTC4 as $\overline{SS}$ of SPI.	In	V_{DD}
	PTC5 as SPSCK of SPI.	In/Out	V_{DD}
	PTC6 as SCTxD of IRSCI; open drain output.	Out	V_{DD}
	PTC7 as SCRxD of IRSCI.	In	V_{DD}
PTD0/KBI0 : PTD7/KBI7	8-bit general purpose I/O port with schmitt trigger inputs.	In/Out	V_{DD}
	Pins as keyboard interrupts (with pullup), KBI0–KBI7.	In	V_{DD}

1. See Chapter 22 Electrical Specifications for V_{REG} tolerance.

1.6 Power Supply Bypassing (VDD, VDDA, VSS, VSSA)

V_{DD} and V_{SS} are the power supply and ground pins, the MCU operates from a single power supply together with an on chip voltage regulator.

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 Figure 1-5 shows. Place the bypass capacitors as close to the MCU power pins as possible. Use high-frequency-response ceramic capacitor for C_{BYPASS}, C_{BULK} are optional bulk current bypass capacitors for use in applications that require the port pins to source high current level.

3.4 Configuration Register 2 (CONFIG2)

Address:	$001D							
	Bit 7	6	5	4	3	2	1	Bit 0
Read:	STOP_ ICLKDIS	STOP_ RCLKEN	STOP_ XCLKEN	OSCCLK1	OSCCLK0	0	0	SCIBD- SRC
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 3-3. Configuration Register 2 (CONFIG2)

STOP_ICLKDIS — Internal Oscillator Stop Mode Disable

STOP_ICLKDIS disables the internal oscillator during stop mode. Setting the STOP_ICLKDIS bit disables the oscillator during stop mode. (See Chapter 5 Oscillator (OSC).)

Reset clears this bit.

1 = Internal oscillator disabled during stop mode

0 = Internal oscillator enabled to operate during stop mode

STOP_RCLKEN — RC Oscillator Stop Mode Enable Bit

STOP_RCLKEN enables the RC oscillator to continue operating during stop mode. Setting the STOP_RCLKEN bit allows the oscillator to operate continuously even during stop mode. This is useful for driving the timebase module to allow it to generate periodic wake up while in stop mode. (See Chapter 5 Oscillator (OSC).)

Reset clears this bit.

1 = RC oscillator enabled to operate during stop mode

0 = RC oscillator disabled during stop mode

STOP_XCLKEN — X-tal Oscillator Stop Mode Enable Bit

STOP_XCLKEN enables the crystal (x-tal) oscillator to continue operating during stop mode. Setting the STOP_XCLKEN bit allows the x-tal oscillator to operate continuously even during stop mode. This is useful for driving the timebase module to allow it to generate periodic wake up while in stop mode. (See Chapter 5 Oscillator (OSC).) Reset clears this bit.

1 = X-tal oscillator enabled to operate during stop mode

0 = X-tal oscillator disabled during stop mode

OSCCLK1, OSCCLK0 — Oscillator Output Control Bits

OSCCLK1 and OSCCLK0 select which oscillator output to be driven out as OSCCLK to the timebase module (TBM). Reset clears these two bits.

OSCCLK1	OSCCLK0	Timebase Clock Source
0	0	Internal oscillator (ICLK)
0	1	RC oscillator (RCCLK)
1	0	X-tal oscillator (XTAL)
1	1	Not used

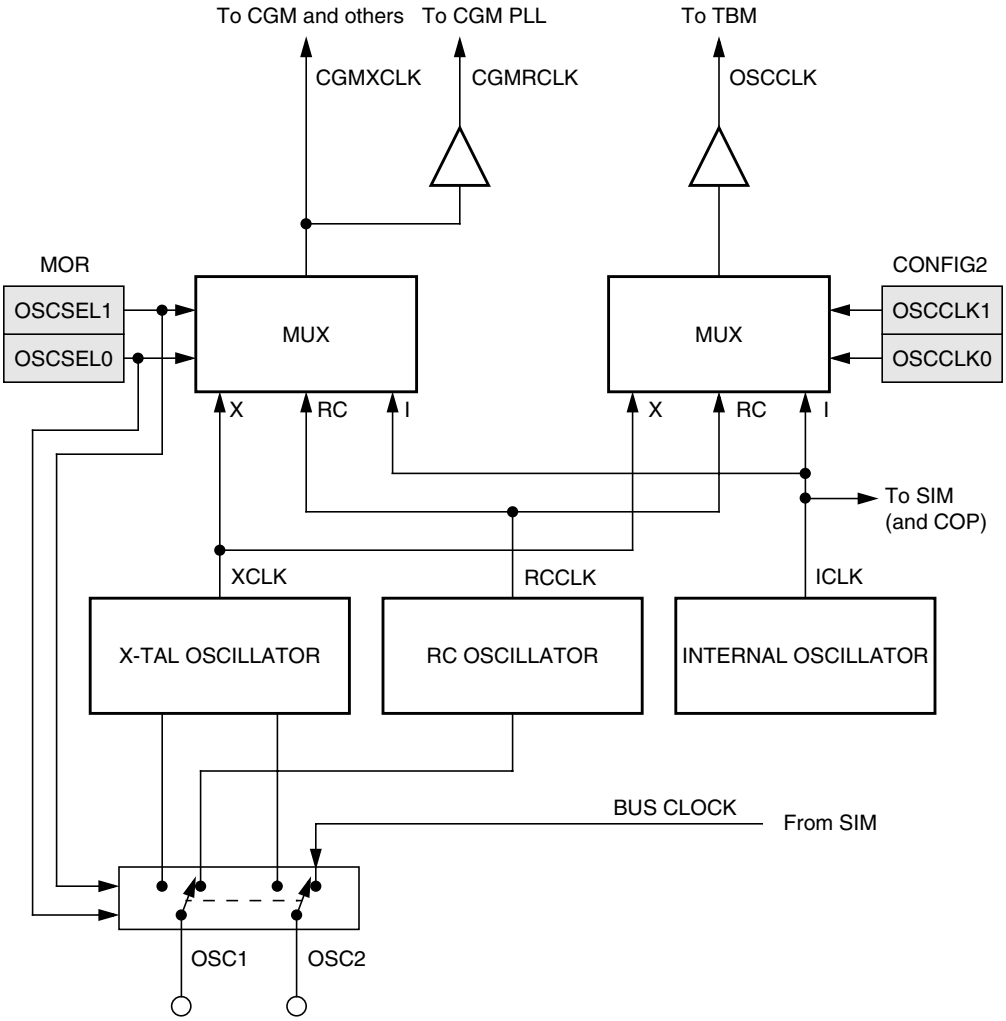

Figure 5-1. Oscillator Module Block Diagram

5.2.1 CGM Reference Clock Selection

The clock generator module (CGM) reference clock (CGMXCLK) is the reference clock input to the MCU. It is selected by programming two bits in a FLASH memory location; the mask option register (MOR), at $FFCF. See 3.5 Mask Option Register (MOR).

Address: $FFCF

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	OSCSEL1	OSCSEL0	R	R	R	R	R	R
Reset:	Unaffected by reset							
Erased:	1	1	1	1	1	1	1	1

R = Reserved

Figure 5-2. Mask Option Register (MOR)

Table 5-1. CGMXCLK Clock Selection

OSCSEL1	OSCSEL0	CGMXCLK	OSC2 Pin	Comments
0	0	—	—	Not used
0	1	ICLK	f_{BUS}	Internal oscillator generates the CGMXCLK.
1	0	RCCLK	f_{BUS}	RC oscillator generates the CGMXCLK. Internal oscillator is available after each POR or reset.
1	1	XCLK	Inverting output of X-TAL	X-tal oscillator generates the CGMXCLK. Internal oscillator is available after each POR or reset.

NOTE

The internal oscillator is a free running oscillator and is available after each POR or reset. It is turned-off in stop mode by setting the STOP_ICLKDIS bit in CONFIG2.

5.2.2 TBM Reference Clock Selection

The timebase module reference clock (OSCCLK) is selected by configuring two bits in the CONFIG2 register, at $001D. See Chapter 3 Configuration & Mask Option Registers (CONFIG & MOR).

Address: $001D

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	STOP_ ICLKDIS	STOP_ RCLKEN	STOP_ XCLKEN	OSCCLK1	OSCCLK0	0	0	SCIBD- SRC
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 5-3. Configuration Register 2 (CONFIG2)

Table 5-2. Timebase Module Reference Clock Selection

OSCCLK1	OSCCLK0	Timebase Clock Source
0	0	Internal oscillator (ICLK)
0	1	RC oscillator (RCCLK)
1	0	X-tal oscillator (XCLK)
1	1	Not used

NOTE

The RCCLK or XCLK is only available if that clock is selected as the CGM reference clock, whereas the ICLK is always available.

NOTE

PLLON and BCS have built-in protection that prevents the base clock selector circuit from selecting the VCO clock as the source of the base clock if the PLL is off. Therefore, PLLON cannot be cleared when BCS is set, and BCS cannot be set when PLLON is clear. If the PLL is off (PLLON = 0), selecting CGMPCLK requires two writes to the PLL control register. (See 6.3.8 Base Clock Selector Circuit.)

PRE1 and PRE0 — Prescaler Program Bits

These read/write bits control a prescaler that selects the prescaler power-of-two multiplier, P. (See 6.3.3 PLL Circuits and 6.3.6 Programming the PLL.) PRE1 and PRE0 cannot be written when the PLLON bit is set. Reset clears these bits.

These prescaler bits affects the relationship between the VCO clock and the final system bus clock.

Table 6-2. PRE1 and PRE0 Programming

PRE1 and PRE0	P	Prescaler Multiplier
00	0	1
01	1	2
10	2	4
11	3	8

VPR1 and VPR0 — VCO Power-of-Two Range Select Bits

These read/write bits control the VCO's hardware power-of-two range multiplier E that, in conjunction with L (See 6.3.3 PLL Circuits, 6.3.6 Programming the PLL, and 6.5.4 PLL VCO Range Select Register.) controls the hardware center-of-range frequency, f_{VRS}. VPR1:VPR0 cannot be written when the PLLON bit is set. Reset clears these bits.

Table 6-3. VPR1 and VPR0 Programming

VPR1 and VPR0	E	VCO Power-of-Two Range Multiplier
00	0	1
01	1	2
10	2	4

NOTE: Do not program E to a value of 3.

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.

Setting MS0B links channels 0 and 1 and configures them for buffered PWM operation. The TIM channel 0 registers (TCH0H:TCH0L) initially control the buffered PWM output. TIM status control register 0 (TSCR0) controls and monitors the PWM signal from the linked channels.

Clearing the toggle-on-overflow bit, TOVx, inhibits output toggles on TIM overflows. Subsequent output compares try to force the output to a state it is already in and have no effect. The result is a 0% duty cycle output.

Setting the channel x maximum duty cycle bit (CHxMAX) and setting the TOVx bit generates a 100% duty cycle output. (See 9.9.4 TIM Channel Status and Control Registers.)

9.5 Interrupts

The following TIM sources can generate interrupt requests:

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

9.6 Low-Power Modes

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

9.6.1 Wait Mode

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

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

9.6.2 Stop Mode

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

9.7 TIM During Break Interrupts

A break interrupt stops the TIM counter and inhibits input captures.

The system integration module (SIM) controls whether status bits in other modules can be cleared during the break state. The BCFE bit in the break flag control register (BFCR) enables software to clear status bits during the break state. (See 21.5.4 SIM Break Flag Control Register.)

Depending on the state of the WAKE bit in SCC1, 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

With the WAKE bit clear, setting the RWU bit after the RxD pin has been idle may cause the receiver to wake up immediately.

11.4.3.7 Receiver Interrupts

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

- SCI receiver full (SCRF) — The SCRF bit in SCS1 indicates that the receive shift register has transferred a character to the SCDR. SCRF can generate a receiver CPU interrupt request. Setting the SCI receive interrupt enable bit, SCRIE, in SCC2 enables the SCRF bit to generate receiver CPU interrupts.
- Idle input (IDLE) — The IDLE bit in SCS1 indicates that 10 or 11 consecutive logic 1s shifted in from the RxD pin. The idle line interrupt enable bit, ILIE, in SCC2 enables the IDLE bit to generate CPU interrupt requests.

11.4.3.8 Error Interrupts

The following receiver error flags in SCS1 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 SCDR. The previous character remains in the SCDR, and the new character is lost. The overrun interrupt enable bit, ORIE, in SCC3 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 SCC3 enables NF to generate SCI error CPU interrupt requests.
- Framing error (FE) — The FE bit in SCS1 is set when a logic 0 occurs where the receiver expects a stop bit. The framing error interrupt enable bit, FEIE, in SCC3 enables FE to generate SCI error CPU interrupt requests.
- Parity error (PE) — The PE bit in SCS1 is set when the SCI detects a parity error in incoming data. The parity error interrupt enable bit, PEIE, in SCC3 enables PE to generate SCI error CPU interrupt requests.

NOTE

The PTB2/TxD pin is an open-drain pin when configured as an output. Therefore, when configured as a general purpose output pin (PTB2), a pullup resistor must be connected to this pin.

11.7.2 RxD (Receive Data)

When the SCI is enabled (ENSCI=1), the PTB3/RxD pin becomes the serial data input, RxD, to the SCI receiver regardless of the state of the DDRB3 bit in data direction register B (DDRB).

NOTE

The PTB3/RxD pin is an open-drain pin when configured as an output. Therefore, when configured as a general purpose output pin (PTB3), a pullup resistor must be connected to this pin.

11.8 I/O Registers

These I/O registers control and monitor SCI operation:

- SCI control register 1 (SCC1)
- SCI control register 2 (SCC2)
- SCI control register 3 (SCC3)
- SCI status register 1 (SCS1)
- SCI status register 2 (SCS2)
- SCI data register (SCDR)
- SCI baud rate register (SCBR)

11.8.1 SCI Control Register 1

SCI control register 1:

- Enables loop mode operation
- Enables the SCI
- Controls output polarity
- Controls character length
- Controls SCI wakeup method
- Controls idle character detection
- Enables parity function
- Controls parity type

Address: $0013

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	LOOPS	ENSCI	TXINV	M	WAKE	ILTY	PEN	PTY
Reset:	0	0	0	0	0	0	0	0

Figure 11-9. SCI Control Register 1 (SCC1)

LOOPS — Loop Mode Select Bit

This read/write bit enables loop mode operation. In loop mode the RxD pin is disconnected from the SCI, and the transmitter output goes into the receiver input. Both the transmitter and the receiver must be enabled to use loop mode. Reset clears the LOOPS bit.

1 = Loop mode enabled
0 = Normal operation enabled

ENSCI — Enable SCI Bit

This read/write bit enables the SCI and the SCI baud rate generator. Clearing ENSCI sets the SCTE and TC bits in SCI status register 1 and disables transmitter interrupts. Reset clears the ENSCI bit.

1 = SCI enabled
0 = SCI disabled

TXINV — Transmit Inversion Bit

This read/write bit reverses the polarity of transmitted data. Reset clears the TXINV bit.

1 = Transmitter output inverted
0 = Transmitter output not inverted

NOTE

Setting the TXINV bit inverts all transmitted values, including idle, break, start, and stop bits.

M — Mode (Character Length) Bit

This read/write bit determines whether SCI characters are eight or nine bits long. (See Table 11-5.) The ninth bit can serve as an extra stop bit, as a receiver wakeup signal, or as a parity bit. Reset clears the M bit.

1 = 9-bit SCI characters
0 = 8-bit SCI characters

- Framing error (FE) — The FE bit in IRSCS1 is set when a logic 0 occurs where the receiver expects a stop bit. The framing error interrupt enable bit, FEIE, in IRSCC3 enables FE to generate SCI error CPU interrupt requests.
- Parity error (PE) — The PE bit in IRSCS1 is set when the SCI detects a parity error in incoming data. The parity error interrupt enable bit, PEIE, in IRSCC3 enables PE to generate SCI error CPU interrupt requests.

12.6 Low-Power Modes

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

12.6.1 Wait Mode

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

If SCI module functions are not required during wait mode, reduce power consumption by disabling the module before executing the WAIT instruction.

Refer to 7.6 Low-Power Modes for information on exiting wait mode.

12.6.2 Stop Mode

The SCI module is inactive after the execution of a STOP instruction. The STOP instruction does not affect SCI register states. SCI module operation resumes after an external interrupt.

Because the internal clock is inactive during stop mode, entering stop mode during an SCI transmission or reception results in invalid data.

Refer to 7.6 Low-Power Modes for information on exiting stop mode.

12.7 SCI During Break Module Interrupts

The system integration module (SIM) controls whether status bits in other modules can be cleared during interrupts generated by the break module. The BCFE bit in the SIM break flag control register (SBFCR) enables software to clear status bits during the break state.

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

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

12.8 I/O Signals

The two IRSCI I/O pins are:

- PTC6/SCTxD — Transmit data
- PTC7/SCRxD — Receive data

Chapter 13
Serial Peripheral Interface Module (SPI)

13.1 Introduction

This section describes the serial peripheral interface (SPI) module, which allows full-duplex, synchronous, serial communications with peripheral devices.

13.2 Features

Features of the SPI module include the following:

- Full-duplex operation
- Master and slave modes
- Double-buffered operation with separate transmit and receive registers
- Four master mode frequencies (maximum = bus frequency ÷ 2)
- Maximum slave mode frequency = bus frequency
- Serial clock with programmable polarity and phase
- Two separately enabled interrupts:
  - SPRF (SPI receiver full)
  - SPTE (SPI transmitter empty)
- Mode fault error flag with CPU interrupt capability
- Overflow error flag with CPU interrupt capability
- Programmable wired-OR mode

13.3 Pin Name Conventions and I/O Register Addresses

The text that follows describes the SPI. The SPI I/O pin names are $\overline{SS}$ (slave select), SPSCK (SPI serial clock), CGND (clock ground), MOSI (master out slave in), and MISO (master in/slave out). The SPI shares four I/O pins with four parallel I/O ports.

The full names of the SPI I/O pins are shown in Table 13-1. The generic pin names appear in the text that follows.

Table 13-1. Pin Name Conventions

SPI Generic Pin Names:		MISO	MOSI	$\overline{SS}$	SPSCK	CGND
Full SPI Pin Names:	SPI	PTC2/MISO	PTC3/MOSI	PTC4/$\overline{SS}$	PTC5/SPSCK	V_{SS}

Figure 13-1 summarizes the SPI I/O registers.

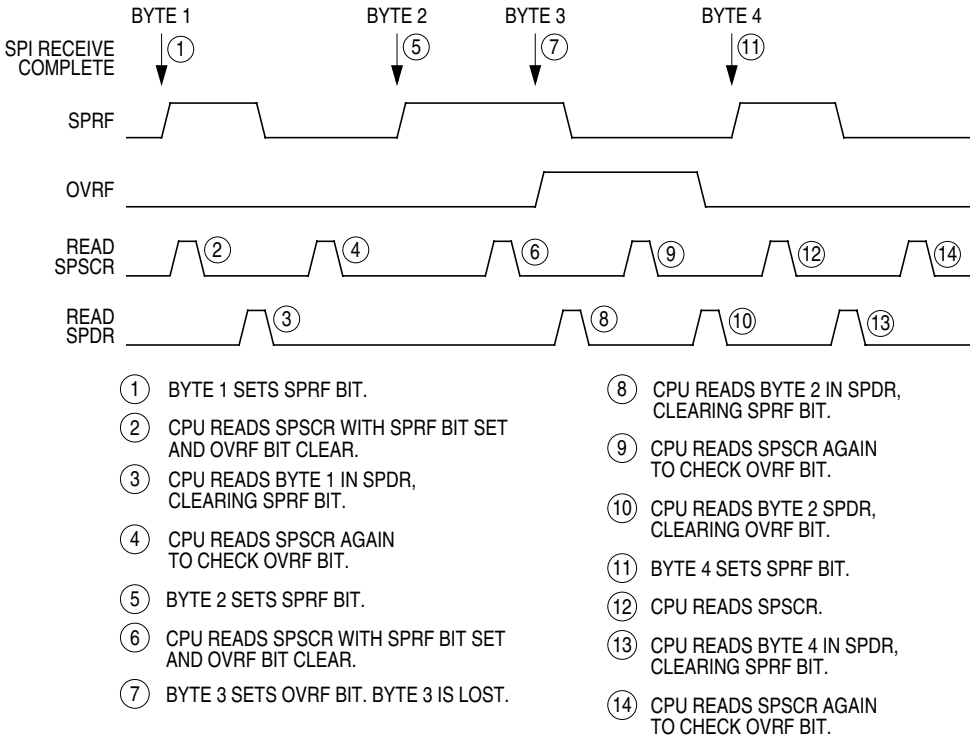

Figure 13-10. Clearing SPRF When OVRF Interrupt Is Not Enabled

13.7.2 Mode Fault Error

Setting the SPMSTR bit selects master mode and configures the SPSCK and MOSI pins as outputs and the MISO pin as an input. Clearing SPMSTR selects slave mode and configures the SPSCK and MOSI pins as inputs and the MISO pin as an output. The mode fault bit, MODF, becomes set any time the state of the slave select pin, $\overline{SS}$, is inconsistent with the mode selected by SPMSTR.

To prevent SPI pin contention and damage to the MCU, a mode fault error occurs if:

- The $\overline{SS}$ pin of a slave SPI goes high during a transmission
- The $\overline{SS}$ pin of a master SPI goes low at any time

For the MODF flag to be set, the mode fault error enable bit (MODFEN) must be set. Clearing the MODFEN bit does not clear the MODF flag but does prevent MODF from being set again after MODF is cleared.

MODF generates a receiver/error CPU interrupt request if the error interrupt enable bit (ERRIE) is also set. The SPRF, MODF, and OVRF interrupts share the same CPU interrupt vector. (See Figure 13-11.) It is not possible to enable MODF or OVRF individually to generate a receiver/error CPU interrupt request. However, leaving MODFEN low prevents MODF from being set.

In a master SPI with the mode fault enable bit (MODFEN) set, the mode fault flag (MODF) is set if $\overline{SS}$ goes to logic 0. A mode fault in a master SPI causes the following events to occur:

- If ERRIE = 1, the SPI generates an SPI receiver/error CPU interrupt request.
- The SPE bit is cleared.
- The SPTE bit is set.
- The SPI state counter is cleared.
- The data direction register of the shared I/O port regains control of port drivers.

13.12.5 CGND (Clock Ground)

CGND is the ground return for the serial clock pin, SPSCK, and the ground for the port output buffers. It is internally connected to V_{SS} as shown in Table 13-1.

13.13 I/O Registers

Three registers control and monitor SPI operation:

- SPI control register (SPCR)
- SPI status and control register (SPSCR)
- SPI data register (SPDR)

13.13.1 SPI Control Register

The SPI control register:

- Enables SPI module interrupt requests
- Configures the SPI module as master or slave
- Selects serial clock polarity and phase
- Configures the SPSCK, MOSI, and MISO pins as open-drain outputs
- Enables the SPI module

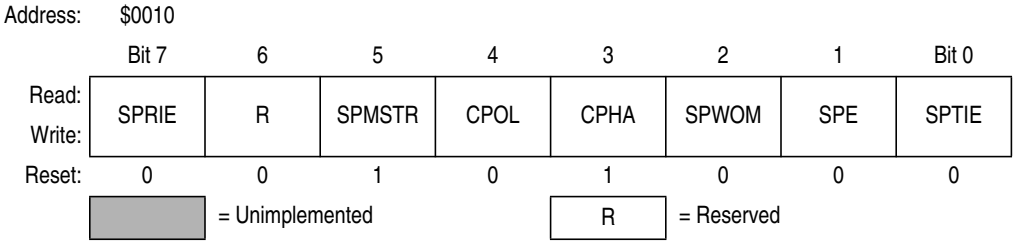

Figure 13-13. SPI Control Register (SPCR)

SPRIE — SPI Receiver Interrupt Enable Bit

This read/write bit enables CPU interrupt requests generated by the SPRF bit. The SPRF bit is set when a byte transfers from the shift register to the receive data register. Reset clears the SPRIE bit.

1 = SPRF CPU interrupt requests enabled
0 = SPRF CPU interrupt requests disabled

SPMSTR — SPI Master Bit

This read/write bit selects master mode operation or slave mode operation. Reset sets the SPMSTR bit.

1 = Master mode
0 = Slave mode

CPOL — Clock Polarity Bit

This read/write bit determines the logic state of the SPSCK pin between transmissions. (See Figure 13-4 and Figure 13-6.) To transmit data between SPI modules, the SPI modules must have identical CPOL values. Reset clears the CPOL bit.

CPHA — Clock Phase Bit

Table 13-4. SPI Master Baud Rate Selection

SPR1 and SPR0	Baud Rate Divisor (BD)
00	2
01	8
10	32
11	128

Use this formula to calculate the SPI baud rate:

$$\text{Baud rate} = \frac{\text{CGMOUT}}{2 \times \text{BD}}$$

where:

CGMOUT = base clock output of the clock generator module (CGM)
BD = baud rate divisor

13.13.3 SPI Data Register

The SPI data register consists of the read-only receive data register and the write-only transmit data register. Writing to the SPI data register writes data into the transmit data register. Reading the SPI data register reads data from the receive data register. The transmit data and receive data registers are separate registers that can contain different values. (See Figure 13-2.)

Address: $0012

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	R7	R6	R5	R4	R3	R2	R1	R0
Write:	T7	T6	T5	T4	T3	T2	T1	T0
Reset:	Unaffected by reset							

Figure 13-15. SPI Data Register (SPDR)

R7–R0/T7–T0 — Receive/Transmit Data Bits

NOTE

Do not use read-modify-write instructions on the SPI data register since the register read is not the same as the register written.

Chapter 14
Multi-Master IIC Interface (MMIIC)

14.1 Introduction

The multi-master IIC (MMIIC) interface is a two wire, bidirectional serial bus which provides a simple, efficient way for data exchange between devices. The interface is designed for internal serial communication between the MCU and other IIC devices. It has hardware generated START and STOP signals; and byte by byte interrupt driven software algorithm.

This bus is suitable for applications which need frequent communications over a short distance between a number of devices. It also provides a flexibility that allows additional devices to be connected to the bus. The maximum data rate is 100k-bps, and the maximum communication distance and number of devices that can be connected is limited by a maximum bus capacitance of 400pF.

This MMIIC interface is also SMBus (System Management Bus) version 1.0 and 1.1 compatible, with hardware cyclic redundancy code (CRC) generation, making it suitable for smart battery applications.

14.2 Features

Features of the MMIC module include:

- Full SMBus version 1.0/1.1 compliance
- Multi-master IIC bus standard
- Software programmable for one of eight different serial clock frequencies
- Software controllable acknowledge bit generation
- Interrupt driven byte by byte data transfer
- Calling address identification interrupt
- Arbitration loss detection and no-ACK awareness in master mode and automatic mode switching from master to slave
- Auto detection of R/W bit and switching of transmit or receive mode accordingly
- Detection of START, repeated START, and STOP signals
- Auto generation of START and STOP condition in master mode
- Repeated start generation
- Master clock generator with eight selectable baud rates
- Automatic recognition of the received acknowledge bit
- Busy detection
- Software enabled 8-bit CRC generation/decoding

14.6.2 MMIIC Control Register 1 (MMCR1)

Address: $0049

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	MMEN	MMIEN	0	0	MMTXAK	REPSEN	MMCRCBYTE	0
Write:			MMCLRBB					
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 14-5. MMIIC Control Register 1 (MMCR1)

MMEN — MMIIC Enable

This bit is set to enable the Multi-master IIC module. When MMEN = 0, module is disabled and all flags will restore to its power-on default states. Reset clears this bit.

1 = MMIIC module enabled
0 = MMIIC module disabled

MMIEN — MMIIC Interrupt Enable

When this bit is set, the MMTXIF, MMRXIF, MMALIF, and MMNAKIF flags are enabled to generate an interrupt request to the CPU. When MMIEN is cleared, the these flags are prevented from generating an interrupt request. Reset clears this bit.

1 = MMTXIF, MMRXIF, MMALIF, and/or MMNAKIF bit set will generate interrupt request to CPU
0 = MMTXIF, MMRXIF, MMALIF, and/or MMNAKIF bit set will not generate interrupt request to CPU

MMCLRBB — MMIIC Clear Busy Flag

Writing a logic 1 to this write-only bit clears the MMBB flag. MMCLRBB always reads as a logic 0. Reset clears this bit.

1 = Clear MMBB flag
0 = No affect on MMBB flag

MMTXAK — MMIIC Transmit Acknowledge Enable

This bit is set to disable the MMIIC from sending out an acknowledge signal to the bus at the 9th clock bit after receiving 8 data bits. When MMTXAK is cleared, an acknowledge signal will be sent at the 9th clock bit. Reset clears this bit.

1 = MMIIC does not send acknowledge signals at 9th clock bit
0 = MMIIC sends acknowledge signal at 9th clock bit

REPSEN — Repeated Start Enable

This bit is set to enable repeated START signal to be generated when in master mode transfer (MMAST = 1). The REPSEN bit is cleared by hardware after the completion of repeated START signal or when the MMAST bit is cleared. Reset clears this bit.

1 = Repeated START signal will be generated if MMAST bit is set
0 = No repeated START signal will be generated

MMCRCBYTE — MMIIC CRC Byte

In receive mode, this bit is set by software to indicate that the next receiving byte will be the packet error checking (PEC) data.

In master receive mode, after completion of CRC generation on the received PEC data, an acknowledge signal is sent if MMTXAK = 0; no acknowledge is sent If MMTXAK = 1.

In slave receive mode, no acknowledge signal is sent if a CRC error is detected on the received PEC data. If no CRC error is detected, an acknowledge signal is sent if MMTXAK = 0; no acknowledge is sent If MMTXAK = 1.

15.7 I/O Registers

These I/O registers control and monitor ADC operation:

- ADC status and control register (ADSCR) — $0057
- ADC clock control register (ADICLK) — $0058
- ADC data register high:low 0 (ADRH0:ADRL0) — $0059:$005A
- ADC data register low 1–3 (ADRL1–ADRL3) — $005B–$005D
- ADC auto-scan control register (ADASCR) — $005E

15.7.1 ADC Status and Control Register

Function of the ADC status and control register is described here.

Address:	$0057							
Read:	COCO	AIEN	ADCO	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0
Write:								
Reset:	0	0	0	1	1	1	1	1

Figure 15-3. ADC Status and Control Register (ADSCR)

COCO — Conversions Complete Bit

In non-interrupt mode (AIEN = 0), COCO is a read-only bit that is set at the end of each conversion. COCO will stay set until cleared by a read of the ADC data register. Reset clears this bit.

In interrupt mode (AIEN = 1), COCO is a read-only bit that is not set at the end of a conversion. It always reads as a 0.

1 = Conversion completed (AIEN = 0)
0 = Conversion not completed (AIEN = 0) or CPU interrupt enabled (AIEN = 1)

NOTE

The write function of the COCO bit is reserved. When writing to the ADSCR register, always have a 0 in the COCO bit position.

AIEN — ADC Interrupt Enable Bit

When this bit is set, an interrupt is generated at the end of an ADC conversion. The interrupt signal is cleared when the data register, ADR0, is read or the ADSCR is written. Reset clears the AIEN bit.

1 = ADC interrupt enabled
0 = ADC interrupt disabled

ADCO — ADC Continuous Conversion Bit

When set, the ADC will convert samples continuously and update the ADC data register at the end of each conversion. Only one conversion is allowed when this bit is cleared. Reset clears the ADCO bit.

1 = Continuous ADC conversion
0 = One ADC conversion

This bit should not be set when auto-scan mode is enabled; i.e. when ASCAN=1.

ADCH[4:0] — ADC Channel Select Bits

ADCH[4:0] form a 5-bit field which is used to select one of the ADC channels when not in auto-scan mode. The five channel select bits are detailed in Table 15-1.

NOTE

Care should be taken when using a port pin as both an analog and a digital input simultaneously to prevent switching noise from corrupting the analog

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0007	Data Direction Register D (DDRD)	Read: Write:	DDRD7	DDRD6	DDRD5	DDRD4	DDRD3	DDRD2	DDRD1	DDRD0
		Reset:	0	0	0	0	0	0	0	0
$000C	Port-A LED Control Register (LEDA)	Read: Write:	LEDA7	LEDA6	LEDA5	LEDA4	LEDA3	LEDA2	LEDA1	LEDA0
		Reset:	0	0	0	0	0	0	0	0

Figure 16-1. I/O Port Register Summary (Continued)