TMS470R1x Serial Communication Interface (SCI) Reference Guide

Literature Number: SPNU196A
September 2002

Texas Instruments

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of that third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Mailing Address:

Texas Instruments
Post Office Box 655303
Dallas, Texas 75265

REVISION HISTORY

REVISION	DATE	AUTHOR	NOTES
A	9/02	R. Haley	Converted to a stand-alone book
*	10/98	R. Haley	*Initial version*

Contents

Figures

Tables

Serial Communication Interface (SCI)

This reference guide contains a general description of the serial communication interface (SCI) module. The SCI is a universal asynchronous receiver-transmitter (UART) that implements the standard nonreturn to zero (NRZ) format.

Topic **Page**

1 Overview

This section provides an overview of the SCI module. Table 1 contains a brief description of the SCI, lists its significant pins, described interrupts, and defines the low-power mode of operation.

Table 1. SCI Module Overview

Description	The SCI module is a universal asynchronous receiver-transmitter that implements the standard nonreturn to zero format. The SCI can be used to communicate, for example, through an RS-232 port or over a K-line.	
Pins	SCIRx	SCI receive pin
	SCITx	SCI transmit pin
	SCICLK	SCI clock pin used for isosynchronous mode
Interrupts	The SCI has three interrupts: transmit, receive, and error. Each interrupt can be individually enabled.	
Sleep Mode	The SCI can be placed in low-power (sleep) mode and then wake up to receive an incoming message.	
Features	Standard universal asynchronous receiver-transmitter (UART) communication Supports full- or half-duplex operation Standard nonreturn to zero (NRZ) format Double-buffered receive and transmit functions Configurable frame format of 3 to 13 bits per character based on the following: Data word length programmable from one to eight bits Additional address bit in address-bit mode Parity programmable for zero or one parity bit, odd or even parity Stop programmable for one or two stop bits Asynchronous or isosynchronous communication modes Two multiprocessor communication formats allow communication between more than two devices. Sleep mode is available to free CPU resources during multiprocessor communication. The 24-bit programmable baud rate supports 2^{24} different baud rates provide high accuracy baud rate selection. Four error flags and six status flags provide detailed information regarding SCI events. Capability to use DMA for transmit and receive data (requires on-chip DMA controller). See the device-specific data sheet for information on DMA controller availability.	

2 Functional Description of SCI Module

This programmable serial communication interface (SCI) module supports digital communications between the CPU and other asynchronous peripherals that use the standard nonreturn to zero (NRZ) format. The SCI receiver and transmitter are double-buffered, and each has its own separate enable and interrupt bits. The receiver and transmitter may each be operated independently or simultaneously in full duplex mode.

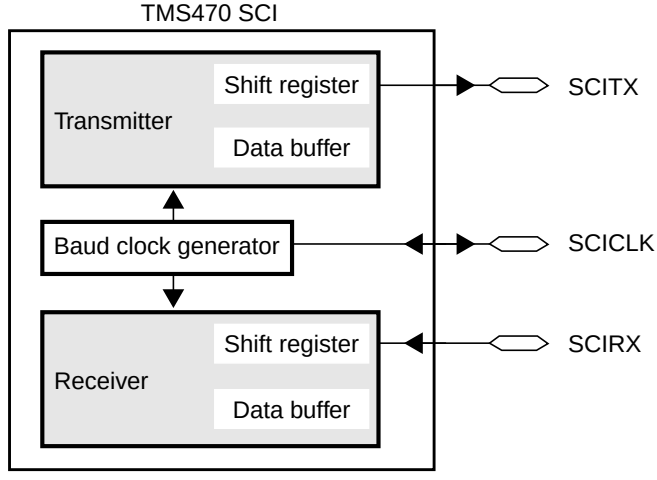
To ensure data integrity, the SCI checks the data it receives for breaks, parity, overrun, and framing errors. The bit rate (baud) is programmable to over 16 million different rates through a 24-bit baud-select register.

2.1 Operation of the SCI Module

As shown in Figure 1, the major components of the SCI include:

- A transmitter (TX) contains two major registers to perform the double-buffering:
  - The transmitter data buffer register (SCITXBUF) contains data loaded by the CPU to be transferred to the shift register for transmission.
  - The transmitter shift register (SCITXSHF) loads data from the data buffer (SCITXBUF) and shifts data onto the SCITX pin, one bit at a time.

Figure 1. Simple SCI Block Diagram



- A receiver (RX) contains two major registers to perform the double-buffering:
  - The receiver shift register (SCIRXSHF) shifts data in from the SCIRX pin one bit at a time and transfers completed data into the receive data buffer.
  - The receiver data buffer register (SCIRXBUF) contains received data transferred from the receiver shift register.
- A programmable baud generator produces either a baud clock scaled from ICLK (interface clock generated by the system module) or allows input of an external clock source to clock the SCI.

The SCI receiver and transmitter can operate independently in a half-duplex configuration (as a receiver only or transmitter only) or simultaneously in a full-duplex configuration. A third port line (SCICLK) is available for an optional external clock input.

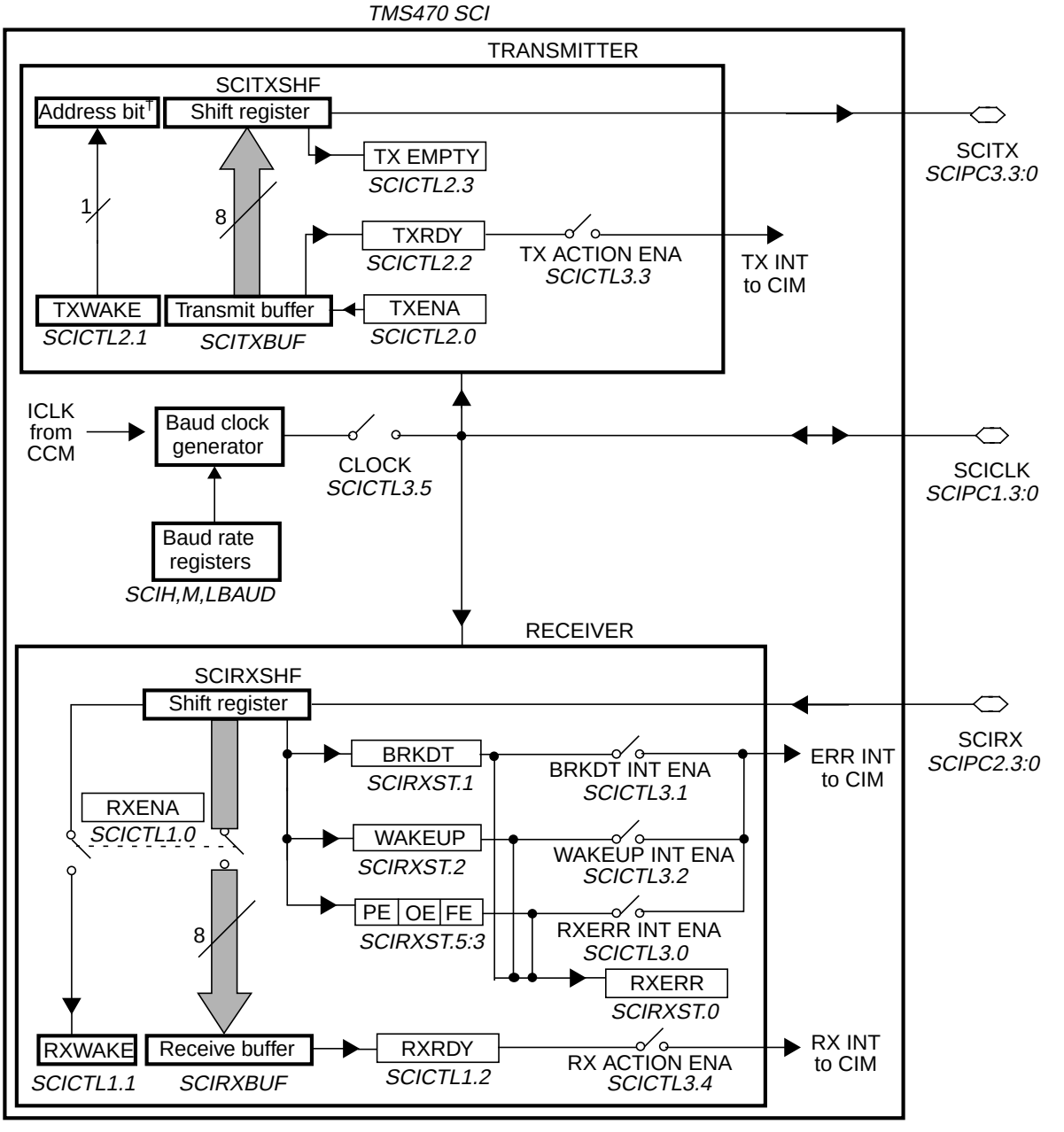
Note: Non-Addressable Internal Registers

In addition to communicating through its addressable registers, the SCI uses two non-addressable internal registers: the transmit shift register (SCITXSHF) and the receive shift register (SCIRXSHF). Throughout this document these registers are referenced only by their mnemonic. For more details, see Section 6.7, *SCI Data Buffers (SCIRXEMU, SCIRXBUF, SCITXBUF)*, on page 47.

2.2 Detailed SCI Block Diagram

Figure 2 provides a detailed view of the SCI module.

Figure 2. Expanded SCI Block Diagram



† Address bit = The SCI saves the value of TXWAKE when it transfers data from SCITXBUF to SCITXSHF.

Note: Italicized mnemonics (for example, *SCIRXST* and *SCILBAUD*) represent SCI control registers. The number following the mnemonic is the associated register bit number.

2.3 SCI Internal Registers

The SCI control registers are located at the SCI base address. The name, mnemonic, and location of each register are shown in Table 2. This table is a general representation of the SCI internal registers. For a more detailed description of each register, see Section 6, *SCI Registers*, on page 31.

Table 2. SCI Internal Registers

Offset Address †	Mnemonic	Name	Description	Page
00h	SCICCR	Communication control register	Defines the frame format, protocol, and communication mode used by the SCI	33
04h	SCICTL1	Control register 1	Controls SCI receive functionality and contains receive flags	35
08h	SCICTL2	Control register 2	Controls SCI transmit functionality and contains transmit flags	37
0Ch	SCICTL3	Control register 3	Controls SCI reset, power mode, and clock, and contains all module interrupt enable bits	39
10h	SCIRXST	Receiver status control register	Contains receiver status flags and updates flags as needed	42
14h	SCIHBAUD	Baud register, HIGH bits	The 3 baud registers that provide 24 bits used to set the SCI baud rate	44
18h	SCIMBAUD	Baud register MIDDLE bits		
1Ch	SCILBAUD	Baud register, LOW bits		
20h	SCIRXEMU	Emulation buffer register	Addressable register used to read the data buffer without affecting the RXRDY flag	47
24h	SCIRXBUF	Receiver data buffer register	Holds data that has been shifted from the receiver shift register SCIRXSHF	47
28h	SCITXBUF	Transmit data buffer register	Holds data that is to be transmitted before it is moved to the transmit shift register SCITXSHF	48
2Ch	SCIPC1	Pin control register 1	Holds current value on the SCICLK clock pin and defines the function of the pin as general purpose I/O or SCICLK	49
30h	SCIPC2	Pin control register 2	Holds current value on the SCIRX receive pin and defines the function of the pin as general purpose I/O or SCIRX	50
34h	SCIPC3	Pin control register 3	Holds current value on the SCITX transmit pin and defines the function of the pin as general purpose I/O or SCITX	51

† The absolute address of these registers is device specific. Consult the specific device data sheet to verify the SCI register addresses.

3 SCI Communication Formats

The SCI module can be configured to meet the requirements of many applications. Because communication formats vary depending on the specific application, many attributes of the TMS470 SCI are user configurable. The list below describes these configuration options.

- ❏ Frame format

  The SCI uses a standard UART nonreturn to zero format with the option to configure parity, number of data bits, and number of stop bits.

- ❏ Timing modes

  The SCI offers two timing modes: asynchronous and isosynchronous. The availability of these modes allows you to choose whether or not a synchronizing clock signal is used during data transmission.

- ❏ Baud rate

  The SCI baud rate is configurable to one of 2^{24} possible data transmission rates.

- ❏ Multiprocessor modes

  The SCI permits networked communication between more than two UART devices by providing two multiprocessor modes: idle-line and address-bit.

Each of the topics above is discussed in detail in this section.

3.1 SCI Frame Format

The SCI uses a programmable frame format. All frames consist of the following:

- ❏ One start bit
- ❏ One to eight data bits
- ❏ Zero or one address bit
- ❏ Zero or one parity bit
- ❏ One or two stop bits

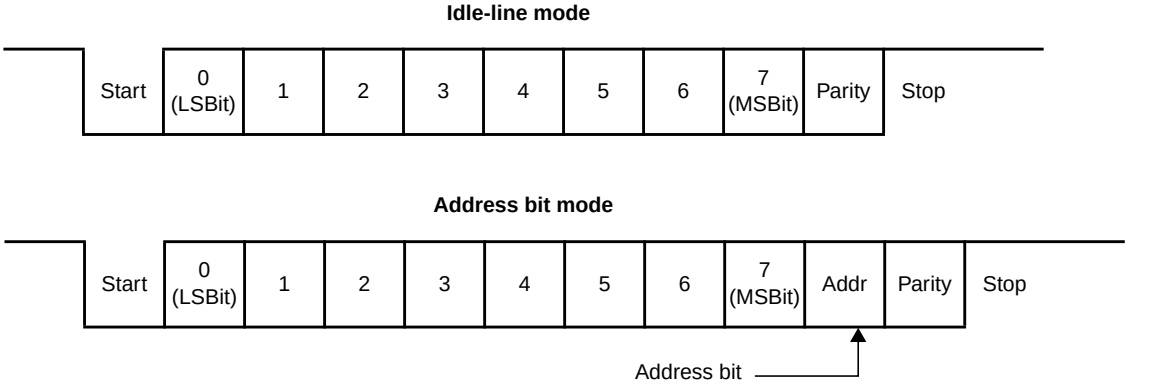
The frame format for both the transmitter and receiver is programmable through the bits in the SCICCR register (see Section 6.1, *Communication Control Register (SCICCR)*, on page 33). Both receive and transmit data is in nonreturn to zero (NRZ) format, which means that the transmit and receive lines are at logic high when idle. Each frame transmission begins with a start bit, in which the transmitter pulls the SCI line low (logic low). Following the start bit, the frame data is sent and received least significant bit first (LSB).

An address bit is present in each frame if the SCI is configured to be in address-bit mode but is not present in any frame if the SCI is configured for idle-line mode. The format of frames with and without the address bit is illustrated in Figure 3.

A parity bit is present in every frame when the PARITY ENA bit (SCICCR.5) is set. The value of the parity bit depends on the number of one bits in the frame and whether odd or even parity has been selected via the PARITY bit (SCICCR.6). Both examples in Figure 3 have parity enabled.

All frames include one stop bit, which is always a high level. This high level at the end of each frame is used to indicate the end of a frame to ensure synchronization between communicating devices. Two stop bits are transmitted if the STOP bit (SCICCR.7) is set. The examples shown in Figure 3 use one stop bit per frame.

Figure 3. Typical SCI Data Frame Formats



3.2 SCI Timing Modes

The SCI can be configured to use asynchronous or isosynchronous timing. The asynchronous timing mode uses only the receive and transmit data lines to interface with devices using the standard universal asynchronous receiver-transmitter (UART) protocol. The isosynchronous timing configuration requires the use of a clock line in addition to the receive and transmit data lines to synchronize communication with other peripherals. Both timing modes can be used with either the address-bit or idle-line multiprocessor communication modes.

3.2.1 Asynchronous Timing Mode

In the asynchronous timing mode, each bit in a frame has a duration of 8 SCI baud clock periods. Each bit therefore consists of 8 samples (one for each

clock period). When the SCI is using asynchronous mode, the baud rates of all communicating devices must match as closely as possible. Receive errors result from devices communicating at different baud rates.

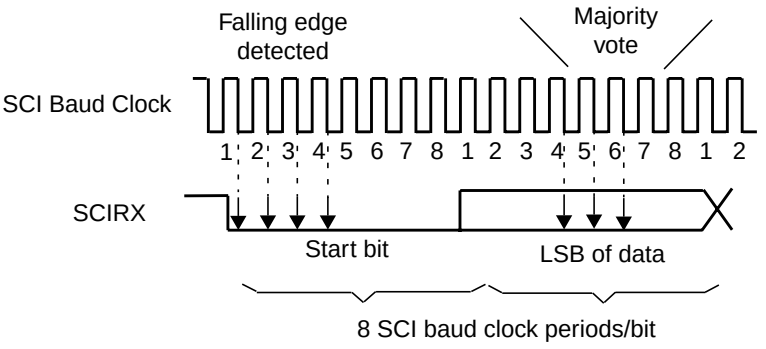
With this timing configuration, the transmitter and receiver are not required to make use of the SCICLK pin. The CLOCK bit (SCICTL3.5) must be set to 1 in order to use the internal SCICLK.

With the receiver in the asynchronous timing mode, the SCI detects a valid start bit if the first four samples after a falling edge on the SCIRX pin are of logic level 0. As soon as a falling edge is detected on SCIRX, the SCI assumes that a frame is being received and synchronizes itself to the bus.

The SCI module has been designed to provide some protection from noise causing unintended start bits or incorrect data. Without protection, a noise spike that brings an idle receive line low may be interpreted as a start bit. The SCI prevents this by requiring a start bit to bring the SCIRX line low for at least four contiguous SCI baud clock periods. If any of the receive samples during the first four SCI baud clock periods is not a logic low, then the SCI does not consider this a start bit and considers the receive line idle. When another falling edge is detected, the SCI checks for a valid, noise-free start bit. When a valid start bit is detected, the SCI determines the value of each bit by sampling the SCI RX line value during the fourth, fifth, and sixth SCI baud clock periods. A majority vote of these samples is used to determine the value stored in the SCI receiver shift register. By sampling in the middle of the bit, the SCI reduces errors caused by propagation delays and rise and fall times. By taking a majority vote, the SCI reduces the likelihood of data corruption caused by data line noise. Figure 4 illustrates how the receiver samples a start bit and a data bit in asynchronous timing mode.

The transmitter transmits each bit for a duration of 8 SCI baud clock periods. During the first clock period for a bit, the transmitter shifts the value of that bit onto the SCITX pin. The transmitter then holds the current bit value on SCITX. for 8 SCI baud clock periods.

Figure 4. Asynchronous Communication Bit Timing



3.2.2 Isosynchronous Timing Mode

In isosynchronous timing mode, each bit in a frame has a duration of exactly 1 baud clock period and therefore consists of a single sample.

With this timing configuration, the transmitter and receiver are required to make use of the SCICLK pin (see Table 3). That is, the CLK FUNC bit (SCIPC1.1) must be set to a 1. The CLOCK bit (SCICTL3.5) can be set to choose whether the baud clock is provided internally or externally. If an internal baud clock is selected (CLOCK bit, SCICTL3.5 = 1), then the SCI outputs its baud clock on the SCICLK pin if the CLK FUNC bit (SCIPC1.1) and the CLK DATA DIR bit (SCIPC1.0) are set.

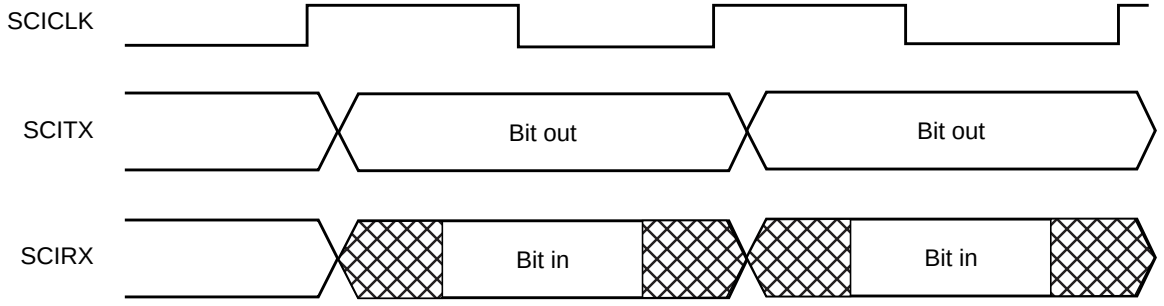
If an external clock is selected (CLOCK bit, SCICTL3.5 = 0), the SCI expects to use a clock signal on the SCICLK pin as its baud clock. This clock is not scaled by the baud rate registers. To configure the SCI to use an external clock source, the CLK FUNC bit (SCIPC1.1) must be set and the CLK DATA DIR bit (SCIPC1.0) must be cleared to 0.

Table 3. SCICLK Configuration (Isosynchronous Mode)

Configuration	CLOCK (SCICTL3.5)	CLK FUNC (SCIPC1.1)	CLK DATA DIR (SCIPC1.0)
Clock produced internally, output on SCICLK pin	1	1	1
Clock produced externally, input to SCICLK pin	0	1	0

Figure 5 illustrates the timing for the communication bits in isosynchronous mode.

Figure 5. Isosynchronous Communication Bit Timing



If the SCI is not currently receiving data, then a low level on the SCIRX pin is interpreted as a start bit and a high level on the SCIRX bit is interpreted as an idle bit. If the SCI is synchronized to a frame (that is, a valid start bit has been received and data is being shifted into the receiver), then the value sampled on SCIRX is taken as the value of the current bit.

Because each bit in this timing mode requires one baud clock, the SCI neither checks for noise on the start bit nor does any majority voting to determine the bit value.

The transmitter shifts data onto SCITX with the rising edge of each baud clock. In isosynchronous timing mode, the transmitter holds the value of the current bit on SCITX for exactly 1 baud clock. Data is latched into the SCI receiver on the falling edge of each SCI baud clock.

3.3 SCI Baud Rate

The SCI can use either an internal or external clock source to generate a baud clock for the SCI. If an internal clock is selected via the CLOCK bit (SCICTL3.5), then the SCI uses the value in the three 8-bit baud registers (SCIHBAUD, SCIMBAUD, and SCILBAUD) to generate an internal baud clock for the SCI. The 24-bit value derived from concatenating SCIHBAUD, SCIMBAUD, and SCILBAUD is referred to as BAUD. This BAUD value is used to scale the ICLK (interface clock) signal fed to the SCI from the TMS470 system module.

In asynchronous timing mode, the SCI generates a baud clock according to the following formula:

$$\textit{Asynchronous baud value} = \left(\frac{\text{ICLK Frequency}}{8(BAUD + 1)}\right) \quad \text{(EQ 1)}$$

For BAUD = 0,

$$\textit{Asynchronous baud value} = \left(\frac{\text{ICLK Frequency}}{16}\right) \quad \text{(EQ 2)}$$

In isosynchronous timing mode, the SCI generates a baud clock according to the following formula:

$$\textit{Isosynchronous baud value} = \left(\frac{\text{ICLK Frequency}}{BAUD + 1}\right) \quad \text{(EQ 3)}$$

For BAUD = 0,

$$\textit{Isosynchronous baud value} = \left(\frac{\text{ICLK Frequency}}{2}\right) \quad \text{(EQ 4)}$$

If an external clock is selected via the CLOCK bit, then the SCI disregards the values in the three 8-bit baud registers. In this case, the clock signal input to the SCICLK pin is used as the baud clock for the SCI. This external clock signal must not exceed ICLK/8.

Note: Transmit on Rising Edge; Receive on Falling Edge

The receiver samples data from SCIRX on the *falling* edge of the baud clock, and the transmitter shifts data onto SCITX on the *rising* edge of the baud clock.

3.4 SCI Multiprocessor Communication Modes

In some applications, the TMS470 SCI may be connected to more than one serial communication device. In such a multiprocessor configuration, several frames of data may be sent to all connected devices or to an individual device. In the case of data sent to an individual device, the receiving devices must determine when they are being addressed. When a message is not intended for them, the devices can ignore the following data which frees up the processor to carry out actions other than reading unimportant data from the SCI bus. When only two devices make up the SCI network, addressing is not needed, so multiprocessor communication schemes are not required.

To interact with multiprocessor bus configurations, the TMS470 SCI includes the option for the use of either idle-line (SCICCR.3 = 0) or address-bit (SCICCR.3 = 1) multiprocessor communication mode. Both of these communication modes can function with either asynchronous or isosynchronous timing.

In a multiprocessor configuration, processors send an address character followed by one or more data characters. The SCI must be able to distinguish address characters from data characters. The idle-line and address-bit communication modes offer two methods of differentiating address and data information:

- Idle-line mode leaves a space of 10 or more idle bits before each frame containing an address and 9 or fewer idle bits before each frame containing data.
- Address-bit mode adds an extra bit (the address bit) into every frame. Frames containing an address have the address bit set to 1, and frames containing data have the address bit set to 0. In this mode, the idle space between frames is irrelevant.

When the SCI is not used in a multiprocessor environment, software can consider all frames as data frames. In this case, the only distinction between the idle-line and address-bit modes is the presence of an extra bit (the address bit) in each frame sent with the address-bit protocol.

The SCI allows full-duplex communication where data can be sent and received via the transmit and receive pins simultaneously. However, the protocol used by the SCI assumes that only one device transmits data on the same bus line at any one time. No arbitration is done by the SCI.

Note: Avoid Transmitting Simultaneously on the Same Serial Bus

The system designer must ensure that devices connected to the same serial bus line do not attempt to transmit simultaneously. If two devices are transmitting different data, the resulting bus conflict could damage the device.

3.4.1 Multiprocessor Definitions

Table 4 defines some terms used in the discussion of idle-line mode and address-bit mode in a multiprocessor environment.

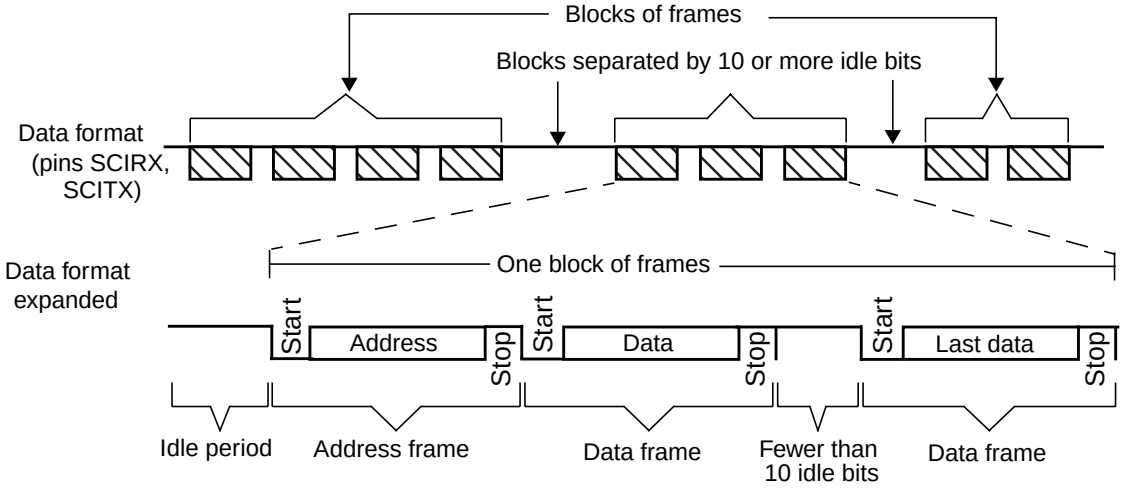
Table 4. Definition of Multiprocessor Terms

Term	Definition
Bit	A single value to be transmitted or received. As discussed in sections 3.2.1 and 3.2.2, each bit sent or received with asynchronous timing consists of 8 SCI baud clock periods, and each bit sent or received with isosynchronous timing consists of a single clock period.
Idle bit	A bit of logic level 1 occurring between frames
Idle period	A group of 10 or more idle bits
Frame	The group of bits to be transmitted or received in order to communicate a piece of data or an address. A frame always contains a start bit, a stop bit, and 1 to 8 data bits and may contain a parity bit, an address bit, and an extra stop bit. Each frame has a format as defined by the settings of the SCICCR register (see Section 6.1, *Communication Control Register (SCICCR)*, on page 33.)
Address frame	A frame whose contents should be interpreted as an address
Data frame	A frame whose contents should be interpreted as data
Block	An address frame followed by zero or more data frames. A new block is begun at every address frame.

3.4.2 Idle-Line Multiprocessor Mode

In idle-line multiprocessor mode, a frame that is preceded by an idle period (10 or more idle bits) is an address frame. A frame that is preceded by fewer than 10 idle bits is a data frame. Figure 6 illustrates the format of several blocks and frames with idle-line mode.

Figure 6. Idle-Line Multiprocessor Communication Format



There are two ways to transmit an address frame using idle-line mode:

- **Method 1:** In software, deliberately leave an idle period between the transmission of the last data frame of the previous block and the address frame of the new block.
- **Method 2:** Configure the SCI to automatically send an idle period between the last data frame of the previous block and the address frame of the new block.

Although Method 1 is only accomplished by a delay loop in software, Method 2 can be implemented by using the transmit buffer and the TXWAKE bit (SCICTL2.1) in the following manner:

1) Write a 1 to the TXWAKE bit.
2) Write a dummy data value to the SCITXBUF register. This triggers the SCI to begin the idle period as soon as the transmitter shift register is empty.
3) Wait for the SCI to clear the TXWAKE flag.
4) Write the address value to SCITXBUF.

As indicated by Step 3, software should wait for the SCI to clear the TXWAKE bit. However, the SCI clears the TXWAKE bit at the same time it sets TXRDY (that is, transfers data from SCITXBUF into SCITXSHF). Therefore, if the TX ACTION ENA bit is set, the transfer of data from SCITXBUF to SCITXSHF causes an interrupt to be generated at the same time that the SCI clears the TXWAKE bit. If this interrupt method is used, software is not required to poll the TXWAKE bit waiting for the SCI to clear it.

When idle-line multiprocessor communications are used, software must ensure that the idle time exceeds 10 bit periods before *addresses* (using one of the methods mentioned above), and software must also ensure that *data* frames are written to the transmitter quickly enough to be sent without a delay of 10 bit periods between frames. Failure to comply with these conditions will result in data interpretation errors by other devices receiving the transmission.

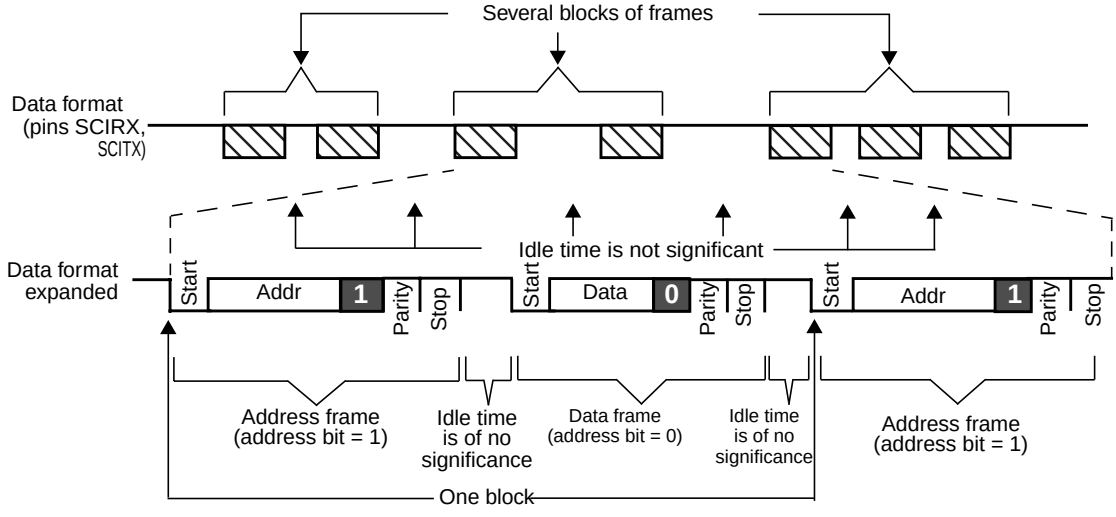
3.4.3 Address-Bit Multiprocessor Mode

In the address-bit protocol, each frame has an extra bit immediately following the data field called an address bit. A frame with the address bit set to 1 is an address frame; a frame with the address bit set to 0 is a data frame. The idle period timing is irrelevant in this mode. Figure 7 illustrates the format of several blocks and frames with the address-bit mode.

When address-bit mode is used, the value of the TXWAKE bit is the value sent as the address bit. To send an address frame, software must set the TXWAKE bit (SCICTL2.1 = 1). This bit is cleared as the contents of the SCITXBUF are shifted from the TXWAKE register so that all frames sent are data except when the TXWAKE bit is written as a 1.

No dummy write to SCITXBUF is required before an address frame is sent in address-bit mode. The first byte written to SCITXBUF after the TXWAKE bit is written to 1 is transmitted with the address bit set when address-bit mode is used.

Figure 7. Address-Bit Multiprocessor Communication Format



3.4.4 Sleep Mode for Multiprocessor Communication

When the SCI receives data and transfers that data from SCIRXSHF to SCIRXBUF, the RXRDY bit is set and if RX ACTION ENA is set, the SCI also generates an interrupt or a DMA request. The interrupt triggers the CPU to read the newly received frame before another one is received. In multiprocessor communication modes, this default behavior may be enhanced to provide selective indication of new data. When the TMS470 SCI receives an address frame that does not match its address, the device can ignore the data following this non-matching address until the next address frame by using sleep mode. Sleep mode can be used with both idle-line and address-bit multiprocessor modes.

If sleep mode is enabled by the SLEEP bit (SCICTL1.3 = 1), then the SCI transfers data from SCIRXSHF to SCIRXBUF only for address frames. Therefore, in sleep mode, all data frames are assembled in the SCIRXSHF register without being shifted into the SCIRXBUF and without initiating a receive interrupt or DMA request. Upon reception of an address frame, the contents of the SCIRXSHF are moved into SCIRXBUF, and the software must read SCIRXBUF and determine if the SCI is being addressed by comparing the received address against the address previously set in the software and stored somewhere in memory (the SCI does not have hardware available for address comparison). If the SCI is being addressed, the software must clear the SLEEP bit so that the SCI will load SCIRXBUF with the data of the data frames that follow the address frame.

When the SCI has been addressed and sleep mode has been disabled (in software) to allow the receipt of data, the SCI should check the RXWAKE bit (SCICTL1.1) to determine when the next address has been received. This bit is set to 1 if the current value in SCIRXBUF is an address and set to 0 if SCIRXBUF contains data. If the RXWAKE bit is set, then software should check the address in SCIRXBUF against its own address. If it is still being addressed, then sleep mode should remain disabled. Otherwise, the SLEEP bit should be set again.

Following is a sequence of events typical of sleep mode operation:

- ❏ The SCI is configured and both sleep mode and receive actions are enabled.
- ❏ An address frame is received and a receive interrupt is generated.
- ❏ Software compares the received address frame against that set by software and determines that the SCI is not being addressed, so the value of the SLEEP bit is not changed.
- ❏ Several data frames are shifted into SCIRXSHF, but no data is moved to SCIRXBUF and no receive interrupts are generated.

- ❏ A new address frame is received and a receive interrupt is generated.
- ❏ Software compares the received address frame against that set by software and determines that the SCI is being addressed and clears the SLEEP bit.
- ❏ Data shifted into SCIRXSHF is transferred to SCIRXBUF, and a receive interrupt is generated after each data frame is received.
- ❏ In each interrupt routine, software checks RXWAKE to determine if the current frame is an address frame.
- ❏ Another address frame is received, RXWAKE is set, software determines that the SCI is **not** being addressed and sets the SLEEP bit back to 1. No receive interrupts are generated for the data frames following this address frame.

By ignoring data frames that are not intended for the device, fewer interrupts are generated. These interrupts would otherwise require CPU intervention to read data that is of no significance to this specific device. Using sleep mode can help free some CPU resources.

Except for the RXRDY flag, the SCI continues to update the receiver status flags (see Table 7, on page 40) while sleep mode is active. In this way, if an error occurs on the receive line, an application can immediately respond to the error and take the appropriate corrective action.

Because the RXRDY bit is not updated for data frames when sleep mode is enabled, the SCI can enable sleep mode and use a polling algorithm if desired. In this case, when RXRDY is set, software knows that a new address has been received. If the SCI is not being addressed, then the software should not change the value of the SLEEP bit and should continue to poll RXRDY.

4 SCI Data Transfer

Data transmitted and received by the SCI must be moved to and from the SCI buffer registers from some location in device memory. Two transfer options are available:

- ❏ CPU instructions driven by interrupts or polling
- ❏ SCI DMA requests

This section discusses the configuration of the SCI for interrupt-driven operation and for DMA data transfer.

4.1 SCI Interrupts

The SCI receiver and transmitter can be controlled by interrupts. The receive and transmit interrupts allow efficient operation of the SCI by reading and writing character information to and from the SCI as new data arrives and when old data has just been sent. The RXRDY flag (SCICTL1.2) used by the receiver indicates that new data is available to be read. The receiver also has various error interrupts that indicate when a particular error condition is active. An active error interrupt condition is indicated by the RXERR flag in SCIRXST. However, the exact source of an error interrupt can be determined by checking the parity error (PE), frame error (FE), overrrun error (OE), break-detect (BRKDT), and wake-up (WAKEUP) flags also located in SCIRXST. Additionally, the transmitter uses the TXRDY flag (SCICTL2.2) to indicate that the transmitter is ready for new data to be written that will be sent to the bus.

Transmit, receive, and error interrupts are enabled or disabled through separate interrupt-enable bits. When not enabled, the interrupts are not asserted; however, polled operation of the SCI is still possible because the interrupt flags continue to indicate module events.

The SCI module generates three interrupt requests to the TMS470 system module: one each for transmitter, receiver, and error interrupts. Each of these interrupts must also be configured in the TMS470 system module before operation. Normally, the error interrupt has the highest priority, the receiver interrupt has the next highest priority, and the transmitter generally has the lowest priority. This prioritizing scheme reduces the possibility of missed error conditions and receiver overrun. For interrupt priority levels on a specific device, consult the device data sheet.

4.1.1 Transmit Interrupt

To use transmit interrupt functionality, the TX DMA ENA bit (SCICTL2.5) must be cleared. The following paragraphs describe how a CPU interrupt can be initiated by a transmit ready condition.

The transmit ready (TXRDY) flag is set when the SCI transfers the contents of SCITXBUF to the shift register, SCITXSHF. The TXRDY flag indicates that SCITXBUF is ready to be loaded with more data. In addition, the SCI sets the TX EMPTY bit if both the SCITXBUF and SCITXSHF registers are empty. Transmit interrupts are enabled by the TX ACTION ENA (SCICTL3.3) bit. If the TX ACTION ENA bit (SCICTL3.3) is set, then a transmit interrupt is generated when the TXRDY flag goes high.

Writing data to the SCITXBUF register clears the TXRDY bit. When this data has been moved to the SCITXSHF register, the TXRDY bit is set again. The interrupt request can be suspended by clearing the TX ACTION ENA bit; however, when the TX ACTION ENA bit is again set to 1, the TXRDY interrupt is asserted again. The transmit interrupt request can be eliminated until the next series of values is written to SCITXBUF by disabling the transmitter via the TXENA bit (SCICTRL2.0 = 0), an SCI software reset, or by a device hardware reset.

4.1.2 Receive Interrupt

The receive ready (RXRDY) flag is set when the SCI transfers newly received data from SCIRXSHF to SCIRXBUF. The RXRDY flag therefore indicates that the SCI has new data to be read. Receive interrupts are enabled by the RX ACTION ENA bit. If the RX ACTION ENA bit (SCICTL3.4) is set when the SCI sets the RXRDY flag, then a receive interrupt is generated.

On a device with both SCI and a DMA controller, the bits RX DMA ALL (SCICTL1.6) and RX DMA ENA (SCICTL1.5) must be cleared to select interrupt functionality.

4.1.3 Error Interrupts

The TMS470 SCI provides hardware indication of error conditions to provide you with information about the status of module operation. According to the data being assembled by the receiver, the SCI monitors received data for errors and sets the parity error (PE), framing error (FE), and/or the break-detect (BRKDT) flag when these conditions are detected. In addition, the SCI sets the overrun error (OE) flag if a transfer of new data from SCIRXSHF to SCIRXBUF overwrites unread data in SCIRXBUF. (If both overrun and parity errors occur, only the overrun error flag is set.) The SCI sets the wake-up flag (WAKEUP) if bus activity on the RX line either prevents power-down mode from being entered, or RX line activity causes an exit from power-down mode. Each of these flags is located in the receiver status (SCIRXST) register.

The RXERR flag, also in the SCIRXST register, is the logical OR of the parity error, framing error, overrun error, wake-up, and break-detect flags. This

feature allows software that is polling for receiver errors to check only one bit, which will indicate if any or all of the five error conditions are active.

Error interrupts are controlled by three separate enable bits: RXERR INT ENA, BRKDT INT ENA, and WAKEUP INT ENA.

- If RXERR INT ENA (SCICTL3.0) is set, an error interrupt is generated when the receiver detects either a parity, framing, or overrun error. The break-detect interrupt is enabled separately.
- If BRKDT INT ENA (SCICTL3.1) is set, an error interrupt is generated if the receiver detects a break condition. A break condition occurs when the SCIRX line remains continuously low (active) for at least 10 bits immediately following a missed stop bit. This is further explained in the BRKDT bit description in Section 6.5, *Receiver Status Register (SCIRXST)*, on page 42.
- If WAKEUP INT ENA (SCICTL3.2) is set, an error interrupt is generated when bus activity on the RX line either prevents power-down mode from being entered or RX line activity causes an exit from power-down mode.

4.2 SCI DMA Requests

Byte transfers from the SCI receiver and to the SCI transmitter can be controlled by a DMA (direct memory access) controller on microcontrollers that include the DMA controller module (see your device data sheet for information on DMA availability). The DMA controller moves data directly from memory to the SCI transmitter, or the DMA can store received data directly to memory. In both cases, the chief benefit of SCI DMA requests is to minimize CPU intervention. The DMA allows the CPU resources to be used on other tasks and transfers large data blocks more efficiently.

To enable DMA requests from the SCI, the DMA module must be configured to work with the SCI. The SCI uses two DMA request lines: one for the receiver and one for the transmitter, each of which must be configured separately. Refer to the DMA module documentation for more information on properly configuring the DMA controller module.

4.2.1 Transmit DMA Request

The TXRDY flag is set when the SCI transfers the contents of SCITXBUF to SCITXSHF. The TXRDY flag indicates that SCITXBUF is ready to be loaded with more data. In addition, the SCI sets the TX EMPTY bit if both the SCITXBUF and SCITXSHF registers are empty.

Transmit DMA requests are enabled by the setting TX DMA ENA and TX ACTION ENA bits.

- ❏ If the TX DMA ENA bit (SCICTL2.5) and TX ACTION ENA bit (SCICTL3.3) are set, then a TX DMA request is sent to the DMA when data is written to SCITXBUF and TXRDY is set.
- ❏ The DMA will write the first byte to the transmit buffer.

4.2.2 Receive DMA Request

The receiver DMA request is set when a frame is received successfully and DMA functionality has been previously enabled. The RXRDY flag is set when the SCI transfers newly received data from the SCIRXSHF register to the SCIRXBUF buffer. The RXRDY flag therefore indicates that the SCI has new data to be read. Receive DMA requests are enabled by the RX ACTION ENA bit.

Parity, overrun, break detect, wake-up, and framing errors generate an error interrupt request immediately upon detection, if enabled, even if the device is in the process of a DMA data transfer. The DMA transfer is postponed until the error interrupt is served. The error interrupt can delete this particular DMA request by reading the receive buffer.

In multiprocessor mode, the SCI can generate receiver interrupts for address frames and DMA requests for data frames. This is controlled by an extra select bit in the SCICTL1 register (RX DMA ALL, SCICTL1.6).

- ❏ If the RX DMA ALL bit (SCICTL1.6) is set and the RX DMA ENA bit (SCICTL1.5) and the RX ACTION ENA bit (SCICTL3.4) are set when the SCI sets the RXRDY flag, then a receive DMA request is generated for address and data frames.
- ❏ If the RX DMA ALL bit (SCICTL1.6) is cleared and the RX DMA ENA bit (SCICTL1.5) and the RX ACTION ENA bit (SCICTL3.4) are set when the SCI sets the RXRDY flag upon receipt of a data frame, then a receive DMA request is generated. Receive interrupt requests are generated for address frames.
- ❏ In multiprocessor mode with the SLEEP bit (SCICTL1.3) set, no DMA is generated for received data frames. The software must clear the SLEEP bit before data frames can be received.

Table 5 specifies the bit values for DMA requests in multiprocessor modes.

Table 5. DMA Requests in Multiprocessor Modes

	RX DMA ALL (SCICTL1.6)	RX DMA ENA (SCICTL1.5)	RX ACTION ENA (SCICTL3.4)
DMA request generated for address and data frames	1	1	1
Interrupt request generated for address frames and DMA request generated for data frames	0	1	1

5 Operating the SCI

Before the SCI sends or receives data, its registers should be properly configured. Upon power-up or a system-level reset, each bit in the SCI registers is set to a default state. Of particular importance is the SW nRESET bit (SCICTL3.7). This active-low bit is initialized to 0 and keeps the SCI in a reset state until it is programmed to 1. Therefore, all SCI configuration should be completed before a 1 (one) is written to the SW nRESET bit.

The following sections describe general configuration requirements for the SCI as well as the configuration process for the SCI receiver and SCI transmitter.

5.1 Configuration Requirements

The SCI contains some registers and bits that affect operation of both the transmitter and receiver. The control registers and bits listed below must be configured to suit the application whenever the SCI is used.

- ❏ See Section 6.1, *Communication Control Register (SCICCR)*, on page 33.
- ❏ See Section 6.6, *Baud Rate Selection Registers*, on page 44.
- ❏ Continue on suspend bit (CONT); see SCICTL2.7 on page 37.
- ❏ Loop back bit (LOOP BACK); see SCICTL2.6 on page 37.
- ❏ SCI internal clock enable bit (CLOCK), see SCICTL3.5 on page 40.
- ❏ Power-down bit (POWERDOWN); see SCICTL3.6 on page 40.
- ❏ Software reset bit (SW nRESET); see SCICTL3.7 on page 40.

The following list details the configuration steps that software should perform prior to the transmission or reception of data. As long as SW nRESET is held low the entire time that the SCI is being configured, the order in which the registers are programmed is not important.

- ❏ Clear SW nRESET to 0 before configuring the SCI.
- ❏ Select the desired frame format by programming SCICCR.
- ❏ Select the baud rate to be used for communication by programming SCIHBAUD, SCIMBAUD, and SCILBAUD.
- ❏ Select whether an internal or external clock should be used by programming the CLOCK bit.
- ❏ Set the CONT bit if you do not want the SCI to halt for an emulation breakpoint until its current reception or transmission is complete. (This bit is used only in an emulation environment).

- ❏ Set LOOP BACK if you want to connect the transmitter to the receiver internally. (This feature can be used to perform a self-test.)
- ❏ Configure the receiver if data is to be received (see Section 5.2, *Receiving Data*, on page 25).
- ❏ Configure the transmitter if data is to be transmitted (see Section 5.3, *Transmitting Data*, on page 27).
- ❏ Configure the SCIRX and SCITX pins for SCI functionality by setting the RX FUNC bit (SCIPC2.1) and the TX FUNC bit (SCIPC3.1) to 1.
- ❏ Configure the clock pin if the SCI is to output its clock or accept an external clock (see Section 6.8, *Pin Control Register 1 (SCIPC1)*, on page 49).
- ❏ Set SW nRESET to 1 after the SCI is configured.

5.2 Receiving Data

The following paragraphs describe the configuration and signal timing requirements to prepare the device to receive data.

5.2.1 Receiver Configuration

The receiver is configured by setting or clearing the following bits:

- ❏ RX enable (SCICTL1.0)
- ❏ Sleep (SCICTL1.3)
- ❏ RX error interrupt enable (SCICTL3.0)
- ❏ Break-detect interrupt enable (SCICTL3.1)
- ❏ Wake-up interrupt enable (SCICTL3.2)
- ❏ RX action enable (SCICTL3.4)
- ❏ RX DMA enable (SCICTL1.5)
- ❏ RX DMA all (SCICTL1.6)
- ❏ RX pin control register (SCIPC2.3:0)

The SCI receiver is enabled to receive messages if SW nRESET is inactive, and if the RX FUNC bit (SCIPC2.1) and the RXENA bit (SCICTL1.0) are set to 1. If the RX FUNC bit is not set, the SCIRX pin functions as a general purpose I/O pin rather than as an SCI function pin. No data is transferred into the receive buffer register while the RXENA bit is cleared to 0. The receiver assembles data into the receiver shift register (SCIRXSHF), but nothing is moved into the receiver buffer register (SCIRXBUF). Both of these control bits allow the SCI receiver to be held inactive independently of the transmitter. Receive interrupts or DMA requests must also be configured if they are to be used. By default, the interrupts and DMA requests are disabled. For more information on receive interrupts and DMA, see the Section 4.1, *SCI Interrupts*, on page 19 and Section 4.2, *SCI DMA Requests*, on page 21.

When the receiver has been configured and enabled and SW nRESET has been set to 1, the receiver looks for an idle period (approximately 11 bit times). Until the receiver finds this idle period, the SCI regards any high bit sampled as being part of the idle period. If the detection of an idle period is interrupted by a low bit, then the SCI begins searching again for an idle period. When it is searching for an idle period, the SCI does not distinguish between high bits that are part of data being sent and high bits that are part of a *true* idle period.

An idle period must be detected before data can be received regardless of the mode of operation. This requirement is in place so that the SCI can safely connect to the receive bus after power up and wake up from low-power mode. The idle period is required so that the SCI does not begin monitoring the bus in the middle of communication with other peripherals and receive invalid frame data.

After the idle period is found, data is automatically received as it arrives on the SCIRX pin. Because the SCI sets the RXRDY bit when it transfers newly received data from SCIRXSHF to SCIRXBUF, new data should be read each time RXRDY is set. The SCI clears the RXRDY bit after the new data in SCIRXBUF has been read. Also, as data is transferred from SCIRXSHF to SCIRXBUF, the SCI sets FE, OE, or PE if any of these error conditions were detected in the received data. The wake-up and break-detect status bits are also set if one of these errors occurs, but they do not necessarily occur at the same time that new data is being loaded into SCIRXBUF.

It is not necessary to poll the RXRDY bit to wait for it to be set high. If RX ACTION ENA is set, the SCI generates a receive interrupt or DMA request while it is transferring data from SCIRXSHF to SCIRXBUF. Also, if the RXERR INT ENA bit is set, then the SCI generates an error interrupt if any of the corresponding error conditions (PE, OE, or FE) are detected in the received data.

5.2.2 Receiver Signal Timing

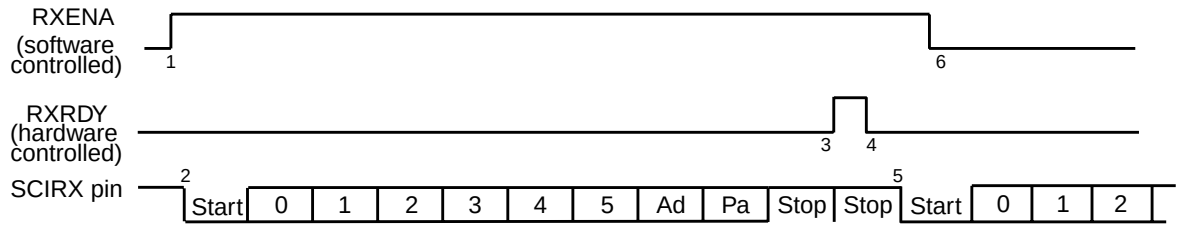
Figure 8 is an example of the timing relationship of RXENA and RXRDY bits and received data. In the figure, the SCI is assumed to be in the following configuration:

- ❏ Address-bit mode
- ❏ Six data bits per frame
- ❏ Parity enabled
- ❏ Two stop bits

The notes following Figure 8 correspond to the numbered steps in the diagram and explain the receive process. Note that when RXRDY is set, the receiver flags PE, OE, FE, RXERR, and RXWAKE become valid for the data

being transferred into SCIRXBUF. Also, the BUS BUSY flag is cleared when the RXRDY flag is set because this condition indicates that the receiver has completed the reception of a frame.

Figure 8. SCI RX Signals in Communication Modes



Notes:
1) RXENA is set high to enable the receiver to load SCIRXBUF with new data.
2) A start bit is detected (for this example, an idle period has already been detected).
3) Data is transferred from SCIRXSHF to SCIRXBUF. An interrupt or DMA is requested if RX ACTION ENA is set.
4) SCIRXBUF is read.
5) Another start bit is detected.
6) RXENA is set low to disable the receiver. Data is shifted into SCIRXSHF but is not transferred to SCIRXBUF.

5.3 Transmitting Data

The following paragraphs describe the configuration and signal timing requirements to prepare the device to transmit data.

5.3.1 Transmitter Configuration

The transmitter is configured by setting or clearing the following bits:

- ❏ TX enable (SCICTL2.0)
- ❏ TX wake-up method select (SCICTL2.1)
- ❏ TX action enable (SCICTL3.3)
- ❏ TX DMA enable (SCICTL2.5)
- ❏ TX pin control register (SCIPC3.3:0)

The SCI transmitter is enabled to transmit messages if SW nRESET is inactive (SCICTL3.7 = 1), and if the TX FUNC bit (SCIPC3.1) and the TXENA bit (SCICTL2.0) are set to 1. If the TX FUNC bit is not set, the SCITX pin functions as a general purpose I/O pin rather than an SCI function pin. No data is transferred out of the shift register. The TXENA bit allows the transfer of data from the transmit buffer (SCITXBUF) to the transmit shift register (SCITXSHF). While the TXENA bit is cleared to 0, software can write data to the transmit buffer, but it is not moved into the transmit shift register. Any value written to the SCITXBUF before TXENA is set to 1 is not transmitted. If the TXENA bit is cleared while the transmitter is sending a byte, then both bytes in the SCITXBUF and SCITXSHF are transmitted before the transmitter is

disabled. Both of these control bits allow for the SCI transmitter to be held inactive independently of the receiver. The transmit interrupt or DMA must also be configured if it is to be used. By default, the interrupt and DMA are disabled. For more information on transmit interrupts and DMA, see Section 4.1, *SCI Interrupts*, on page 19 and Section 4.2, *SCI DMA Requests*, on page 21.

When the transmitter has been configured, enabled, and the SW nRESET has been set to 1, the SCI waits for data to be written to SCITXBUF, transfers it to SCITXSHF, and transmits it. The flags TXRDY and TX EMPTY indicate the status of the transmit buffers. That is, when the transmitter is ready for data to be written to SCITXBUF, the TXRDY bit is set. Additionally, if both SCITXBUF and SCITXSHF are empty, then the TX EMPTY bit is also set.

Because the SCI sets the TXRDY bit each time it transfers data from SCITXBUF to SCITXSHF, new data can be written to SCITXBUF each time TXRDY is set. The SCI clears the TXRDY bit only when data is currently in SCITXBUF. It is not necessary to poll the TXRDY bit to wait for it to be set high. If TX ACTION ENA is set, the SCI generates an interrupt or DMA request while it is transferring data from SCITXBUF to SCITXSHF and setting the TXRDY bit.

When the SCI has completed transmission of all pending frames, the SCITXSHF register and SCITXBUF are empty, the TXRDY bit is set, and an interrupt is generated, if enabled. Because all data has been transmitted, the interrupt request should be halted by either disabling the transmit interrupt via the TX ACTION ENA bit (SCICTL3.3 = 0) or by disabling the transmitter (SCICTL2.0 = 0). Note that disabling the transmit interrupt halts the interrupt request only until the transmit interrupt is re-enabled.

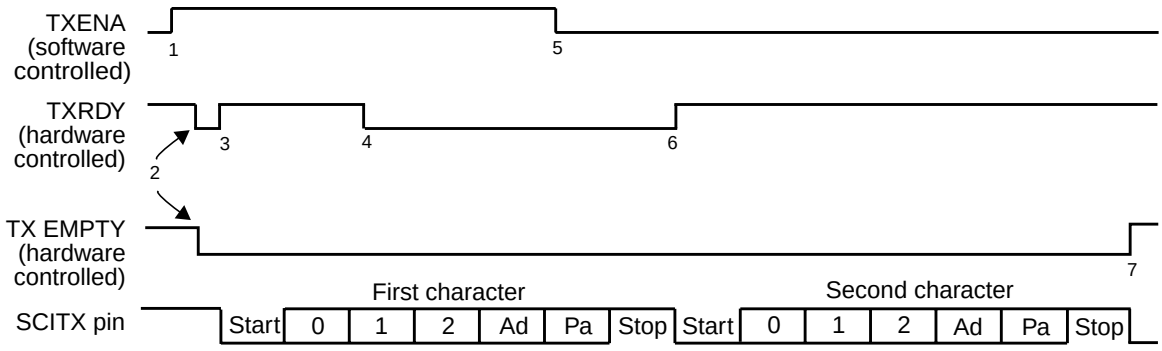
5.3.2 Transmitter Signal Timing

Figure 9 is an example of the timing relationship of TXENA, TX EMPTY, and TXRDY, and transmitted data. In the figure, the SCI is assumed to be in the following configuration:

- ❏ Address-bit mode
- ❏ Three data bits per frame
- ❏ Parity enabled
- ❏ One stop bit

The notes following Figure 9 correspond to the numbered steps in the diagram and explain the transmit process.

Figure 9. SCI TX Signals in Communication Modes



Notes:
1) TXENA is set high to enable the transmitter to send data.
2) A write occurs to SCITXBUF.
3) The SCI transfers data from SCITXBUF to SCITXSHF and begins transmitting. The transmitter requests an interrupt or DMA if TX ACTION ENA is set.
4) A second write occurs to SCITXBUF.
5) TXENA is set low to disable the transmitter.
6) The first character is completed, so the SCI transfers data from SCITXBUF to SCITXSHF and continues transmitting. The transmitter does not request an interrupt because the transmitter is disabled.
7) Transmission is complete and both SCITXBUF and SCITXSHF are empty.

5.4 Power-Down Mode

The SCI can be put in either local or global low-power mode. Global low-power mode is asserted by the system and is not controlled by the SCI. During global low-power mode, all clocks to the SCI are turned off so the module is completely inactive.

Local low-power mode is asserted by setting the POWERDOWN (SCICTL3.6) bit; setting this bit stops the clocks to the SCI internal logic, but the SCI registers continue to be clocked. Setting the PPWNOVR bit found in the CLKCTRL register of the TMS470 system module disables the clocks to all modules in which the local POWERDOWN bit is set. Because setting the POWERDOWN bit causes the SCI to enter local low-power mode, as soon as it exits from local low-power mode, the SCI automatically clears the POWERDOWN bit.

The wake-up interrupt is used to allow the SCI to automatically exit low-power mode when a low level is detected on the SCIRX pin. If this interrupt is disabled (WAKEUP INT ENA is cleared), then the SCI immediately enters low-power mode when it is requested. Also, activity on the SCIRX pin does not cause the SCI to exit low-power mode when the wake-up interrupt is disabled.

Note: Enabling Local Low-Power Mode During Receive and Transmit

If the wake-up interrupt is enabled and low-power mode is requested while the receiver is receiving data, then the SCI immediately generates a wake-up interrupt and prevents the SCI from entering low-power mode. Otherwise, if the wake-up interrupt is disabled, then the SCI cannot prevent entrance into low-power mode while the receiver is active. The BUS BUSY bit can be read to determine if the SCI is in the process of receiving data. If BUS BUSY is active, then the SCI is currently receiving data and the low-power mode should not be requested. The SCI does not exit low-power mode to continue a transmission if it enters low-power mode during that transmission.

6 SCI Registers

The SCI is controlled and accessed through the registers listed Figure 10. Among the features that can be programmed are the communication and timing modes, baud value, frame format, SCI DMA requests, and interrupt enables. The SCI registers are eight bits wide and are word aligned. These registers are accessible in 8-, 16-, and 32-bit reads or writes.

Figure 10. SCI Control Registers

Offset Address †	Register	7	6	5	4	3	2	1	0	Name
00h	SCICCR	STOP	PARITY	PARITY ENA	TIMING MODE	COMM MODE	CHAR2	CHAR1	CHAR0	Communica-tion Control
04h	SCICTL1	Reserved	RX DMA ALL	RX DMA ENA	IDLE	SLEEP	RXRDY	RXWAKE	RXENA	SCI Control Register 1
08h	SCICTL2	CONT	LOOP BACK	TX DMA ENA	Reserved	TX EMPTY	TXRDY	TXWAKE	TXENA	SCI Control Register 2
0Ch	SCICTL3	SW nRE-SET	POWER DOWN	CLOCK	RX ACTION ENA	TX ACTION ENA	WAKEUP INT ENA	BRKDT INT ENA	RXERR INT ENA	SCI Control Register 3
10h	SCIRXST	BUS BUSY	Reserved	FE	OE	PE	WAKEUP	BRKDT	RXERR	Receiver Status
14h	SCIH-BAUD	BAUD23 (*MSBit*)	BAUD22	BAUD21	BAUD20	BAUD19	BAUD18	BAUD17	BAUD16	Baud Rate (MSB)
18h	SCIM-BAUD	BAUD15	BAUD14	BAUD13	BAUD12	BAUD11	BAUD10	BAUD9	BAUD8	Baud Rate (middle)
1Ch	SCILBAUD	BAUD7	BAUD6	BAUD5	BAUD4	BAUD3	BAUD2	BAUD1	BAUD0 (*LSBit*)	Baud Rate (LSB)
20h	SCIRX-EMU	ED7	ED6	ED5	ED4	ED3	ED2	ED1	ED0	EMU Data Buffer
24h	SCIRXBUF	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	Receiver Data Buffer
28h	SCITXBUF	TD7	TD6	TD5	TD4	TD3	TD2	TD1	TD0	Transmit Data Buffer
2Ch	SCIPC1	Reserved				CLK DATA IN	CLK DATA OUT	CLK FUNC	CLK DATA DIR	Pin Control Register 1
30h	SCIPC2	Reserved				RX DATA IN	RX DATA OUT	RX FUNC	RX DATA DIR	Pin Control Register 2
34h	SCIPC3	Reserved				TX DATA IN	TX DATA OUT	TX FUNC	TX DATA DIR	Pin Control Register 3

† The absolute address of these registers is device specific. Consult the specific device data sheet to verify the SCI register addresses.

6.1 Communication Control Register (SCICCR)

The SCICCR register defines the frame format, protocol, and communication mode used by the SCI.

Bits	7	6	5	4	3	2	1	0
00h	STOP	PARITY	PARITY ENA	TIMING MODE	COMM MODE	CHAR2	CHAR1	CHAR0
	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0

R = Read; W = Write; *-n* = Value after reset

Bit 7 **STOP. SCI number of stop bits.** Specifies the number of stop bits per frame.

0 = One stop bit
1 = Two stop bits

The receiver checks for only one stop bit. However in idle-line mode, the receiver waits until the end of the second stop bit (if STOP = 1) to begin checking for an idle period.

Bit 6 **PARITY. SCI parity odd/even selection.** If the PARITY ENA bit (SCICCR.5) is set, PARITY (SCICCR.6) designates odd or even parity.

0 = Odd parity
1 = Even parity

The parity bit is calculated based on the data bits in each frame and the address bit (in address-bit mode). The start and stop fields in the frame are not included in the parity calculation.

For odd parity, the SCI transmits and expects to receive a value in the parity bit that makes odd the total number of bits in the frame with the value of 1.

For even parity, the SCI transmits and expects to receive a value in the parity bit that makes even the total number of bits in the frame with the value of 1.

Bit 5 **PARITY ENA. SCI parity enable.** Enables or disables the parity function.

0 = Parity disabled; no parity bit is generated during transmission or is expected during reception
1 = Parity enabled

Bit 4 **TIMING MODE. SCI timing mode bit.** Selects the SCI timing mode.

0 = Isosynchronous timing
1 = Asynchronous timing

See Section 3.2, *SCI Timing Modes*, on page 8 for more information on the isosynchronous and asynchronous timing modes.

Bit 3 **COMM MODE. SCI communication mode bit.** Selects the SCI communication mode.

0 = Idle-line mode
1 = Address-bit mode

See Section 3.4, *SCI Multiprocessor Communication Modes*, on page 12 for more information on the two available communication modes.

Bits 2:0 **CHAR2:0. Character length control bits.** Sets the SCI data length from 1 to 8 bits. Bit values and the corresponding data lengths are listed in Table 6.

Table 6. CHAR.2:0 Bit Values and Data Lengths

CHAR2:0 Bit Values (Binary)			Data Length (Bits)
CHAR2	CHAR1	CHAR0	
0	0	0	1
0	0	1	2
0	1	0	3
0	1	1	4
1	0	0	5
1	0	1	6
1	1	0	7
1	1	1	8

When data of fewer than eight bits in length is received, it is left justified in SCIRXBUF and padded with trailing zeros. Data read from the SCIRXBUF should be shifted by software to make the received data right justified.

Data written to the SCITXBUF should be right justified but does not need to be padded with leading zeros.

6.2 SCI Control Register 1 (SCICTL1)

The SCICTL1 register controls whether the receiver loads data into SCIRXBUF and whether the receive responds to the data frames differently than address frames. In addition, this register contains two receiver status bits RXRDY and RXWAKE.

Bits	7	6	5	4	3	2	1	0
04h	Reserved	RX DMA ALL	RX DMA ENA	IDLE	SLEEP	RXRDY	RXWAKE	RXENA
	R-0	RW-0	RW-0	R-1	RW-0	RC-0	R-0	RW-0

R = Read; W = Write; C = Clear; *-n* = Value after reset

Bit 7 **Reserved.**

This bit is always read as 0. Writes have no effect.

Bit 6 **RX DMA ALL.** Determines if a separate interrupt is generated for the address frames sent in multiprocessor communications.

0 = RX interrupt request generated for address frames and DMA requests generated for data frames
1 = RX DMA request generated for address and data frames

Bit 5 **RX DMA ENA.** To select DMA requests, this bit must be set. If it is cleared, interrupt requests are generated. When selected, the DMA or INT requests must be enabled. For more information, see bit SCICTL3.4.

0 = RX interrupt request selected
1 = RX DMA request selected

Bit 4 **IDLE. SCI receiver in idle state.** While this bit is set, the SCI looks for an idle period to resynchronize itself with the bit stream. The receiver does not receive any data while the bit is set. The bus must be idle for 11 bit periods to clear this bit. The SCI enters this state if one of the following events occurs: a system reset, an SCI software reset, power down, and if the RX pin is configured as a general I/O pin.

0 = Idle period detected, the SCI is ready to receive.
1 = Idle period not detected, the SCI will not receive any data.

Bit 3 **SLEEP. SCI sleep.** In a multiprocessor configuration, this bit controls the receive sleep function. Clearing this bit brings the SCI out of sleep mode.

0 = Sleep mode disabled
1 = Sleep mode enabled

The receiver still operates when the SLEEP bit is set; however, RXRDY is updated and SCIRXBUF is loaded with new data only when an address frame is detected. The remaining receiver status flags (see Table 7, on page 40) are updated and an error interrupt is requested if the corresponding interrupt enable bit is set, regardless of the value of the SLEEP bit. In this way, if an error is detected on the receive data line while the SCI is asleep, software can promptly deal with the error condition.

The SLEEP bit is **not** automatically cleared when an address byte is detected.

See Section 3.4.4, *Sleep Mode for Multiprocessor Communication*, on page 17 for more information on using the SLEEP bit for multiprocessor communication.

Bit 2 **RXRDY. SCI receiver ready flag.** The receiver sets this bit to indicate that the SCIRXBUF contains new data and is ready to be read by the CPU or DMA. The SCI generates a receive interrupt when it sets this bit if the interrupt-enable bit RX ACTION ENA is set. RXRDY is cleared by an active SW nRESET, a system reset, writing a 1 to this bit, or by reading SCIRXBUF.

Read:
0 = No new data in SCIRXBUF
1 = New data ready to be read from SCIRXBUF

Write:
0 = No effect
1 = Clears this bit to 0

Bit 1 **RXWAKE. Receiver wake-up detect flag.** The SCI sets this bit to indicate that the data currently in SCIRXBUF is an address. RXWAKE is cleared by an active SW nRESET, a system reset, or by the SCI upon receipt of a data frame.

0 = The data in SCIRXBUF is not an address.
1 = The data in SCIRXBUF is an address.

See Section 3.4.4, *Sleep Mode for Multiprocessor Communication*, on page 17 for more information on using the RXWAKE bit with sleep mode.

Bit 0 **RXENA. SCI receiver enable.** RXENA allows or prevents the transfer of data from SCIRXSHF to SCIRXBUF.

0 = Prevents the receiver from transferring data from the shift buffer to the receive buffer
1 = Allows the receiver to transfer data from the shift buffer to the receive buffer

Clearing RXENA stops received characters from being transferred into the receive buffer, prevents the RX status flags (see Table 7, on page 40) from being updated by receive data, and inhibits both receive and error interrupts. However, the shift register continues to assemble data regardless of the state of RXENA.

If RXENA is cleared before the time the reception of a frame is complete, the data from the frame is not transferred into the receive buffer.

If RXENA is set before the time the reception of a frame is complete, the data from the frame is transferred into the receive buffer. If RXENA is set while SCIRXSHF is in the process of assembling a frame, the status flags are not guaranteed to be accurate for that frame. To ensure that the status flags correctly reflect what was detected on the bus during a particular frame, RXENA should be set before the detection of that frame.

6.3 SCI Control Register 2 (SCICTL2)

Bits	7	6	5	4	3	2	1	0
08h	CONT	LOOP BACK	TX DMA ENA	Reserved	TX EMPTY	TXRDY	TXWAKE	TXENA
	RW-0	RW-0	RW-0	R-0	R-1	R-1	RW-0	RW-0

R = Read; W = Write; *-n* = Value after reset

Bit 7 **CONT. Continue on suspend.** This bit has an effect only when a program is being debugged with an emulator, and it determines how the SCI operates when the program is suspended.

0 = When debug mode is entered, the SCI continues to operate until the current transmit and receive functions are complete.
1 = When debug mode is entered, the SCI state machine is frozen. Transmissions are halted and resume when debug mode is exited.

Bit 6 **LOOP BACK.** The self-checking option for the SCI can be selected with this bit. If the SCITX and SCIRX pins are configured with SCI functionality, then the SCITX pin is internally connected to the SCIRX pin. Externally, during loop back operation, the SCITX pin outputs a high value and the SCIRX pin is in a high-impedance state. If this bit value is changed while the SCI is transmitting or receiving data, errors may result.

0 = Loop back mode disabled
1 = Loop back mode enabled

Bit 5 **TX DMA ENA.** To select DMA requests for the transmitter, this bit must be set. If it is cleared, interrupt requests are generated. To enable DMA or INT requests, see the TX ACTION ENA bit description (SCICTL3.3)**.**

0 = TX interrupt request selected
1 = TX DMA request selected

Bit 4 **Reserved.**

This bit is always read as 0. Writes have no effect.

Bit 3 **TX EMPTY. Transmitter empty flag.** The value of this flag indicates the contents of the transmitter's buffer register (SCITXBUF) and shift register (SCITXSHF). An active SW nRESET (SCICTL3.7) or a system reset sets this bit. This bit does not cause an interrupt request.

0 = Transmitter buffer or shift register (or both) are loaded with data.
1 = Transmitter buffer and shift registers are both empty.

Bit 2 **TXRDY. Transmitter buffer register ready flag.** When set, this bit indicates that the transmit buffer register (SCITXBUF) is ready to receive another character. Writing data to SCITXBUF automatically clears this bit. The SCI asserts a transmit interrupt or DMA request after data is written to the SCITXBUF, when it sets this flag if the interrupt/DMA enable bit, TX ACTION ENA (SCICTL3.3), is set. TXRDY is also set to 1 either by enabling SW nRESET (SCICTL3.7) or by a system reset.

0 = SCITXBUF is full.
1 = SCITXBUF is ready to receive the next character.

For more information on transmit interrupt handling, see Section 4.1.1, *Transmit Interrupt*, on page 19.

Bit 1 **TXWAKE. SCI transmitter wake-up method select.** The TXWAKE bit controls whether the data in SCITXBUF should be sent as an address or data frame using multiprocessor communication format. This bit is set to 1 or 0 by software before a byte is written to SCITXBUF and is cleared by the SCI when data is transferred from SCITXBUF to SCITXSHF or by a system reset. TXWAKE is not cleared by the SW nRESET bit (SCICTL3.7).

Address-bit mode:
0 = Frame to be transmitted will be data (address bit = 0).
1 = Frame to be transmitted will be an address (address bit = 1).

Idle-line mode:
0 = Frame to be transmitted will be data.
1 = Following frame to be transmitted will be an address (writing a 1 to this

bit followed by writing dummy data to the SCITXBUF will result in a idle period of 11 bit periods before the next frame is transmitted).

See Section 3.4, *SCI Multiprocessor Communication Modes*, on page 12 for more information on using the TXWAKE bit in the available communication modes.

Bit 0 **TXENA. SCI transmitter enable.** Data is transferred from SCITXBUF to SCITXSHF only when the TXENA bit is set.

0 = Disable transfers from SCITXBUF to SCITXSHF
1 = Enable SCI to transfer data from SCITXBUF to SCITXSHF

Data written to SCITXBUF before TXENA is set is not transmitted. If TXENA is cleared while transmission is ongoing, the data previously written to SCITXBUF is sent.

6.4 SCI Control Register 3 (SCICTL3)

Bits	7	6	5	4	3	2	1	0
0Ch	SW nRESET	POWER DOWN	CLOCK	RX ACTION ENA	TX ACTION ENA	WAKEUP INT ENA	BRKDT INT ENA	RXERR ENA
	RW-0	RWP-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0

R = Read; W = Write; P = Write in privilege modes only; *-n* = Value after reset

Bit 7 **SW nRESET. Software reset (active LOW).** Writing a 0 to this bit initializes the SCI state machines and operating flags as defined in Table 7 and Table 8. All affected logic is held in the reset state until a 1 is written to SW nRESET. After SW nRESET is set to 1, the configuration of the module should not change.

0 = The SCI is in its reset state; no data will be transmitted or received.
1 = The SCI is in its ready state; transmission and reception can be done.

Table 7. SCI Receiver Status Flags

SCI Flag	Register	Bit	Value After SW nRESET†
RXWAKE	SCICTL1	1	0
RXRDY	SCICTL1	2	0
FE	SCIRXST	5	0
OE	SCIRXST	4	0
PE	SCIRXST	3	0
BRKDT	SCIRXST	1	0
RXERR	SCIRXST	0	0

† The flags are frozen with their reset value while SW nRESET = 0.

Table 8. SCI Transmitter Status Flags

SCI Flag	Register	Bit	Value After SW nRESET†
TX EMPTY	SCICTL2	3	1
TXRDY	SCICTL2	2	1

† The flags are frozen with their reset value while SW nRESET = 0.

Note: Configure SCI While Module Is In Reset

The SCI should only be configured while SW nRESET = 0.

Bit 6 **POWERDOWN: Power down.** When the POWERDOWN bit is set, the SCI attempts to enter local low-power mode. If the POWERDOWN bit is set while the receiver is actively receiving data and the wake-up interrupt is enabled, then the SCI immediately asserts an error interrupt to prevent low-power mode from being entered. See Section 5.4, *Power-Down Mode*, on page 29 for more information on low-power mode.

User mode writes:
0 = No effect
1 = No effect

Privilege mode writes:
0 = Normal operation
1 = Request local low power modes

Bit 5 **CLOCK: SCI internal clock enable.** The CLOCK bit determines the source of the module clock on the SCICLK pin.

0 = External SCICLK
1 = Internal SCICLK

If an external clock is selected, then the internal baud rate generator and baud rate registers are bypassed. The maximum frequency allowed for an externally sourced SCI clock is ICLK/8.

Bit 4 **RX ACTION ENA: Receiver DMA or interrupt enable.** Setting this bit enables the SCI to generate a receive interrupt or DMA request after a frame has been completely received and the data is being transferred from SCIRXSHF to SCIRXBUF.

0 = Receive DMA/interrupt disabled
1 = Receive DMA/interrupt enabled

See the RX DMA ENA bit (SCICTL1.5) for more details on selecting DMA or interrupt requests.

Bit 3 **TX ACTION ENA. Transmitter DMA or interrupt enable.** Setting this bit enables the SCI to generate a transmit interrupt as data is being transferred from SCITXBUF to SCITXSHF and the TXRDY bit is being set.

0 = Transmit DMA/interrupt disabled
1 = Transmit DMA/interrupt enabled

See the TX DMA ENA bit (SCICTL1.5) for more details on selecting DMA or interrupt requests.

Bit 2 **WAKEUP INT ENA. Wake-up interrupt enable.** Setting this bit enables the SCI to generate a wake-up interrupt and thereby exit low-power mode. If enabled, the wake-up interrupt is asserted when local low-power mode is requested while the receiver is busy or if a low level is detected on the SCIRX pin during low-power mode. This interrupt is requested on the SCI error interrupt request line.

0 = Wake-up interrupt disabled
1 = Wake-up interrupt enabled

Bit 1 **BRKDT INT ENA. Break-detect interrupt enable.** Setting this bit enables the SCI to generate an error interrupt if a break condition is detected on the SCIRX pin.

0 = Break-detect interrupt disabled
1 = Break-detect interrupt enabled

Bit 0 **ERR INT ENA. SCI receiver enable.** Setting this bit enables the SCI to generate an error interrupt if any of the following conditions are detected on the SCIRX pin.

- ❏ Framing error (FE)
- ❏ Overrun error (OE)
- ❏ Parity error (PE)

Note that break-detect and wake-up interrupts are enabled separately.

0 = Error interrupt disabled
1 = Error interrupt enabled

6.5 Receiver Status Register (SCIRXST)

The SCIRXST register contains receiver status flags. Each time a complete character is transferred from SCIRXSHF to SCIRXBUF, the status flags are updated. Each of the bits except RXERR and BUS BUSY can be cleared by writing a 1 to the bit.

Bits	7	6	5	4	3	2	1	0
10h	BUS BUSY	Reserved	FE	OE	PE	WAKEUP	BRKDT	RXERR
	R-0	R-0	RC-0	RC-0	RC-0	RC-0	RC-0	R-0

R = Read; W = Write; *-n* = Value after reset

Bit 7 **Bus busy flag.** This bit indicates whether the receiver is in the process of receiving a frame. As soon as the receiver detects the beginning of a start bit, the BUS BUSY bit is set to 1. When the reception of a frame is complete, the SCI clears the BUS BUSY bit. If WAKEUP INT ENA is set and power down is requested while this bit is set, the SCI automatically prevents low-power mode from being entered and generates an error interrupt. The BUS BUSY bit is controlled directly by the SCI receiver but is cleared by an active SW nRESET or by a system reset.

0 = Receiver is not currently receiving a frame.
1 = Receiver is currently receiving a frame.

Bit 6 **Reserved.**

This bit is always read as 0. Writes have no effect.

Bit 5 **FE. SCI framing error flag.** This bit is set when an expected stop bit is not found. Only the first stop bit is checked. The missing stop bit indicates that synchronization with the start bit has been lost and that the character is incorrectly framed. Detection of a framing error causes the SCI to generate

an error interrupt if the RXERR INT ENA bit is set. The framing error flag is reset by clearing the SW nRESET bit, by a system reset, or by writing a 1 to this bit.

Read:
0 = No framing error detected
1 = Framing error detected

Write:
0 = No effect
1 = Clears this bit to 0

Bit 4 **OE. SCI overrun error flag.** This bit is set when the transfer of data from SCIRXSHF to SCIRXBUF overwrites unread data already in SCIRXBUF. Detection of an overrun error causes the SCI to generate an error interrupt if the RXERR INT ENA bit is set. The OE flag is reset by an active SW nRESET, a system reset, or by writing a 1 to this bit.

Read:
0 = No overrun error detected
1 = Overrun error detected

Write:
0 = No effect
1 = Clears this bit to 0

Bit 3 **PE. SCI parity error flag.** This bit is set when a parity error is detected in the received data. In address-bit mode, the parity is calculated on the data and address bit fields of the received frame. In idle-line mode, only the data is used to calculate parity. An error is generated when a character is received with a mismatch between the number of 1s and its parity bit. For more information on parity checking, see Section 6.1, *Communication Control Register (SCICCR)*, on page 33. If the parity function is disabled (that is, SCICCR.5 = 0), the PE flag is disabled and read as 0. Detection of a parity error causes the SCI to generate an error interrupt if the RXERR INT ENA bit is set. The PE bit is reset by an active SW nRESET, a system reset, or by writing a 1 to this bit.

Read:
0 = No parity error or parity disabled
1 = Parity error detected

Write:
0 = No effect
1 = Clears this bit to 0

Bit 2 **WAKEUP. Wake-up flag.** This bit is set by the SCI when receiver or transmitter activity has taken the module out of power-down mode. An

interrupt is generated if the WAKEUP INT ENA bit (SCICTL3.2) is set. It is cleared by an active SW nRESET, a system reset, or by writing a 1 to this bit. See Section 5.4, *Power-Down Mode*, on page 29 for more information on low-power mode.

Read:
0 = Do not wake up from power-down mode
1 = Wake up from power-down mode

Write:
0 = No effect
1 = Clears this bit to zero

Bit 1 **BRKDT. SCI break-detect flag.** This bit is set when the SCI detects a break condition on the SCIRX pin. A break condition occurs when the SCIRX pin remains continuously low for at least 10 bits after a missing first stop bit, that is, after a framing error. Detection of a break condition causes the SCI to generate an error interrupt if the BRKDT INT ENA bit is set. The BRKDT bit is reset by an active SW nRESET, a system reset, or by writing a 1 to this bit.

Read:
0 = No break condition detected
1 = Break condition detected

Write:
0 = No effect
1 = Clears this bit to 0

Bit 0 **RXERR. SCI receiver error flag.** The RXERR bit is the logical OR of the framing error, overrun error, parity error, wake-up, and break-detect flags (bits FE, OE, PE, WAKEUP, and BRKDT bits 5, 4, 3, 2, and 1). This bit can be used for fast error-condition checking and is cleared by an active SW nRESET, by a system reset, or by clearing FE, OE, PE, WAKEUP, and BRKDT if any of these bits are set.

Read:
0 = No error flags set
1 = Error flag(s) set

Write:
0 = No effect
1 = Clears this bit to 0

6.6 Baud Rate Selection Registers

This section describes the baud rate selection registers.

Baud Select (MSB) Register (SCIHBAUD)

Bits	7	6	5	4	3	2	1	0
14h	BAUD23 (*MSBit*)	BAUD22	BAUD21	BAUD20	BAUD19	BAUD18	BAUD17	BAUD16
	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0

R = Read; W = Write; *-n* = Value after reset

Baud Select (Middle) Register (SCIMBAUD)

Bits	7	6	5	4	3	2	1	0
18h	BAUD15	BAUD14	BAUD13	BAUD12	BAUD11	BAUD10	BAUD9	BAUD8
	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0

R = Read; W = Write; *-n* = Value after reset

Baud Select (LSB) Register (SCILBAUD)

Bits	7	6	5	4	3	2	1	0
1Ch	BAUD7	BAUD6	BAUD5	BAUD4	BAUD3	BAUD2	BAUD1	BAUD0 (*LSBit*)
	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0

R = Read; W = Write; *-n* = Value after reset

Bits 23–0 **BAUD.23:0. SCI 24-bit baud selection.** Registers SCIHBAUD *(MSB)*, SCIMBAUD (middle), and SCILBAUD *(LSB)* are concatenated to form a 24-bit baud value (this value is referred to as BAUD in baud rate calculations).

If an external clock source is selected (via the CLOCK bit, SCICTL3.5), then the SCI accepts an external clock on the SCICLK pin. If an internal clock source is selected and both the CLK FUNC and CLK DATA DIR bits are set, the SCICLK pin becomes the serial clock output pin. If an internal clock is selected and CLK FUNC is set but CLK DATA DIR is cleared, then the SCI uses an internally generated clock but does not output this clock signal on the SCICLK pin.

The internally generated serial clock is determined by the ICLK and the three baud select registers. The SCI uses the 24-bit value of these registers to select 1 of over 16,700,000 available baud rates. The baud rate can be calculated using the following formulas:

$$\textit{Asynchronous baud value} = \left(\frac{\text{ICLK Frequency}}{8(BAUD + 1)}\right) \quad \text{(EQ 5)}$$

For BAUD = 0,

$$\textit{Asynchronous baud value} = \left(\frac{\text{ICLK Frequency}}{16}\right) \quad \text{(EQ 6)}$$

$$\textit{Isosynchronous baud value} = \left(\frac{\text{ICLK Frequency}}{BAUD + 1}\right) \quad \text{(EQ 7)}$$

For BAUD = 0,

$$\textit{Isosynchronous baud value} = \left(\frac{\text{ICLK Frequency}}{2}\right) \quad \text{(EQ 8)}$$

The SCI receives data on the falling clock edges and transmits data on the rising clock edges. Table 9 contains comparative baud values for different register values (with ICLK = 25 MHz) for asynchronous mode.

Table 9. SCI Baud Selected for Different Register Values—Asynchronous Mode

24-Bit Register Value		Baud Selected		Percent Error
Decimal	Hex	Ideal	Actual	
26	00001A	115200	115740	0.47
53	000035	57600	57870	0.47
80	000050	38400	38580	0.47
162	0000A2	19200	19172	-0.15
299	00012B	10400	10417	0.16
325	000145	9600	9586	-0.15
399	00018F	7812.5	7812.5	0.00
650	00028A	4800	4800	0.00
15624	003BA0	200	200	0.00
624999	098967	5	5	0.00

Note: For this example, ICLK = 25 MHz. Values are in decimal except for column 2.

Table 10 contains comparative baud values for different baud register values (with ICLK = 25 MHz) for the isosynchronous mode.

Table 10. SCI Baud Selected for Different Register Values—Isosynchronous Mode

24-Bit Register Value		Baud Selected		Percent
Decimal	Hex	Ideal	Actual	Error
216	0000D8	115200	115207	0.01
433	0001B1	57600	57604	0.01
650	00028A	38400	38402	0.01
1301	000515	19200	19201	0.01
2403	000963	10400	10399	0.01
2603	000A2B	9600	9601	0.01
3199	000C7F	7812.5	7812.5	0.00
5207	001457	4800	4800	0.00
124999	01E847	200	200	0.00
4999999	4C4B3F	5	5	0.00

Note: For this example, CLK = 25 MHz. Values are in decimal except for column 2. An external SCI clock should not exceed ICLK/8

6.7 SCI Data Buffers (SCIRXEMU, SCIRXBUF, SCITXBUF)

The SCI has three addressable registers in which transmit and receive data is stored.

6.7.1 Receiver Emulation Data Buffer (SCIRXEMU)

The SCIRXEMU register is addressed at a location different from SCIRXBUF but is physically the same register. Unlike SCIRXBUF, however, reading SCIRXEMU does not clear the RXRDY flag. This register should be used only by an emulator which must continually read the data buffer without affecting the RXRDY flag.

Bits	7	6	5	4	3	2	1	0
20h	ED7	ED6	ED5	ED4	ED3	ED2	ED1	ED0
	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

R = Read; *-n* = Value after reset

6.7.2 Receiver Data Buffer (SCIRXBUF)

When a frame has been completely received, the data in the frame is transferred from the receiver shift register SCIRXSHF to this register. As this

transfer occurs, the RXRDY flag is set and a receive interrupt or DMA request is generated if RX ACTION ENA (SCICTL3.4) is set. When the data is read from SCIRXBUF, the RXRDY flag is automatically cleared.

Bits	7	6	5	4	3	2	1	0
24h	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0
	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

R = Read; *-n* = Value after reset

Note: Receive Buffer is Left Justified

For received data that is fewer than 8 bits in length, the SCI loads the data into this register in a left-justified format padded with trailing zeros. Therefore, software should perform a logical shift on the data by the correct number of positions to make it right justified.

6.7.3 Transmit Data Buffer Register (SCITXBUF)

Data to be transmitted is written to the SCITXBUF register. The transfer of data from this register to the transmit shift register SCITXSHF sets the TXRDY flag (register SCICTL2.2), which indicates that SCITXBUF is ready to be loaded with another byte of data. If TX ACTION ENA (register SCICTL3.3) is set, this data transfer also causes an interrupt or DMA request.

Bits	7	6	5	4	3	2	1	0
28h	TD7	TD6	TD5	TD4	TD3	TD2	TD1	TD0
	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0

R = Read; W = Write; *-n* = Value after reset is indeterminate

Note: Receive Buffer Must Be Right Justified

Data written to SCITXBUF register that is fewer than 8 bits long must be right justified but is not required to be padded with leading zeros.

6.8 Pin Control Register 1 (SCIPC1)

Bits	7	6	5	4	3	2	1	0
2Ch	Reserved				CLK DATA IN	CLK DATA OUT	CLK FUNC	CLK DATA DIR
	R-0				R-X	RW-0	RW-0	RW-0

R = Read; W = Write; *-n* = Value after reset; *-X* = Value after reset is indeterminate

Bits 7:4 **Reserved.**

These bits are always read as 0. Writes have no effect.

Bit 3 **CLK DATA IN.** Contains the current value on pin SCICLK.

0 = Pin SCICLK value is logic low.
1 = Pin SCICLK value is logic high.

Bit 2 **CLK DATA OUT.** Contains the data to be output on pin SCICLK if the following conditions are met:

CLK FUNC = 0 (SCICLK pin is a general-purpose I/O.)
CLK DATA DIR = 1 (SCICLK pin is a general-purpose output.)

0 = Output value on SCICLK is a 0 (logic low).
1 = Output value on SCICLK is a 1 (logic high).

Bit 1 **CLK FUNC. Clock function.** Defines the function of pin SCICLK.

0 = SCICLK is a general-purpose digital I/O pin.
1 = SCICLK is the SCI serial clock pin.

Bit 0 **CLK DATA DIR. Clock data direction.** Determines the data direction on the SCICLK pin. The direction is defined differently depending upon the value of the CLK FUNC bit.

If CLK FUNC = 0 (SCICLK = general-purpose I/O), then the CLK DATA DIR bit has the following function:

0 = SCICLK pin is a general-purpose input pin.
1 = SCICLK pin is a general-purpose output pin.

If CLK FUNC = 1 (SCICLK pin has SCI clock functionality) and an internal clock has been selected (SCICTL3.5 = 1), then the CLK DATA DIR bit has the following functionality:

0 = Internal SCI clock is not output on SCICLK pin.
1 = Internal SCI clock is output on SCICLK pin.

If CLK FUNC = 1 (SCICLK pin has SCI clock functionality) and an external clock has been selected (SCICTL3.5 = 0), then the SCI expects an external clock signal not to exceed ICLK/8 on the SCICLK pin (see Table 11).

Table 11. SCICLK Pin Control

Function	CLK DATA IN†	CLK DATA OUT	CLK FUNC	CLK DATA DIR
SCICLK (output)	X	X	1	1
SCICLK (input)	X	X	1	0
General purpose input	X	X	0	0
General purpose output, high	X	1	0	1
General purpose output, low	X	0	0	1

6.9 Pin Control Register 2 (SCIPC2)

Bits	7	6	5	4	3	2	1	0
30h	Reserved				RX DATA IN	RX DATA OUT	RX FUNC	RX DATA DIR
	R-0				R-X	RW-0	RW-0	RW-0

R = Read; W = Write; *-n* = Value after reset; *-X* = Value after reset is indeterminate

Bits 7:4 **Reserved.**

These bits are always read as 0. Writes have no effect.

Bit 3 **RX DATA IN.** Contains current value on the SCIRX pin.

0 = SCIRX value is logic low.
1 = SCIRX value is logic high.

Bit 2 **RX DATA OUT.** Contains the data to be output on pin SCIRX if the following conditions are met:

RX FUNC = 0 (SCIRX pin is a general-purpose I/O.)
RX DATA DIR = 1 (SCIRX pin is a general-purpose output.)

0 = Output value on SCIRX is 0 (logic low).
1 = Output value on SCIRX is 1 (logic high).

Bit 1 **RX FUNC.** Defines the function of pin SCIRX.

0 = SCIRX is a general-purpose digital I/O pin.
1 = SCIRX is the SCI receive pin.

Bit 0 **RX DATA DIR.** Determines the data direction on the SCIRX pin if it is configured with general-purpose I/O functionality (RX FUNC = 0). See Table 12 for bit values.

0 = SCIRX is a general-purpose input pin.
1 = SCIRX is a general-purpose output pin.

Table 12. SCIRX Pin Control

Function	RX DATA IN†	RX DATA OUT	RX FUNC	RX DATA DIR
SCIRX	X	X	1	X
General purpose input	X	X	0	0
General purpose output, high	X	1	0	1
General purpose output, low	X	0	0	1

† RX DATA IN is a read-only bit. Its value always reflects the level of the SCIRX pin.

6.10 Pin Control Register 3 (SCIPC3)

Bits	7	6	5	4	3	2	1	0
34h	Reserved				TX DATA IN	TX DATA OUT	TX FUNC	TX DATA DIR
	R-0				R-X	RW-0	RW-0	RW-0

R = Read; W = Write; *-n* = Value after reset; *-X* = Value after reset is indeterminate

Bits 7:4 **Reserved.**

These bits are always read as 0. Writes have no effect.

Bit 3 **TX DATA IN.** Contains current value on the SCITX pin.

0 = SCITX value is logic low.
1 = SCITX value is logic high.

Bit 2 **TX DATA OUT.** Contains the data to be output on pin SCITX if the following conditions are met:

TX FUNC = 0 (SCITX pin is a general-purpose I/O.)

TX DATA DIR = 1 (SCITX pin is a general-purpose output.)
0 = Output value on SCITX is a 0 (logic low).
1 = Output value on SCITX is a 1 (logic high).

Bit 1 **TX FUNC.** Defines the function of pin SCITX.

0 = SCITX is a general-purpose digital I/O pin.
1 = SCITX is the SCI transmit pin.

Bit 0 **TX DATA DIR.** Determines the data direction on the SCITX pin if it is configured with general-purpose I/O functionality (TX FUNC = 0). See Table 13 for bit values.

0 = SCITX is a general-purpose input pin.
1 = SCITX is a general-purpose output pin.

Table 13. SCITX Pin Control

Function	TX DATA IN†	TX DATA OUT	TX FUNC	TX DATA DIR
SCITX	X	X	1	X
General purpose input	X	X	0	0
General purpose output, high	X	1	0	1
General purpose output, low	X	0	0	1

† TX DATA IN is a read-only bit. Its value always reflects the level of the SCIRX pin.