TMS320x280x, 281x DSP Serial Communication Interface (SCI) Reference Guide

Literature Number: SPRU051A
June 2003

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 the 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.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265

Contents

Figures

Tables

Chapter 1

Overview

The serial communications interface is a two–wire asynchronous serial port, commonly known as a UART. The SCI modules support digital communications between the CPU and other asynchronous peripherals that use the standard non-return-to-zero (NRZ) format. The SCI receiver and transmitter each have a 16-level deep FIFO for reducing servicing overhead, and each has its own separate enable and interrupt bits. Both can be operated independently for half-duplex communication, or simultaneously for full-duplex communication.

To ensure data integrity, the SCI checks received data for break detection, parity, overrun, and framing errors. The bit rate is programmable to different speeds through a 16-bit baud-select register.

Topic **Page**

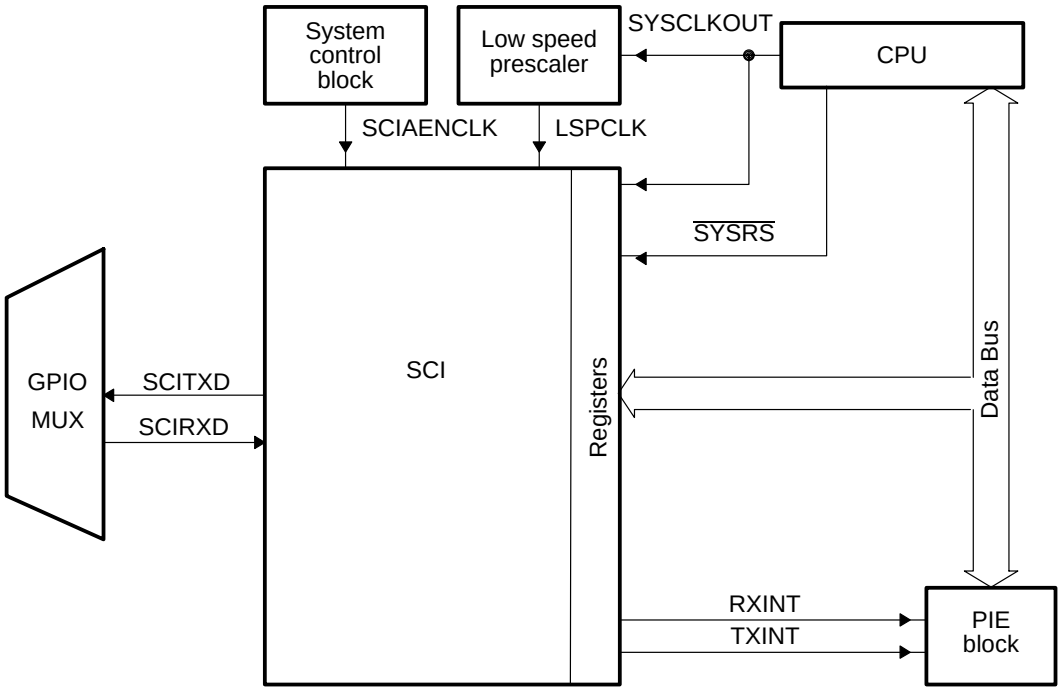
Note: 28x Enhanced Features

The 28x SCI features several enhancements compared to the 240xA SCI. See section 1.2.10 for a description of these features.

1.1 Enhanced SCI Module Overview

The SCI interfaces are shown in Figure 1–1.

Figure 1–1. SCI CPU Interface



Features of the SCI module include:

- Two external pins:
  - SCITXD: SCI transmit-output pin
  - SCIRXD: SCI receive-input pin

  Both pins can be used as GPIO if not used for SCI.
- Baud rate programmable to 64K different rates
- Data-word format
  - One start bit
  - Data-word length programmable from one to eight bits
  - Optional even/odd/no parity bit
  - One or two stop bits
- Four error-detection flags: parity, overrun, framing, and break detection

- Two wake-up multiprocessor modes: idle-line and address bit
- Half- or full-duplex operation
- Double-buffered receive and transmit functions
- Transmitter and receiver operations can be accomplished through interrupt-driven or polled algorithms with status flags.
- Separate enable bits for transmitter and receiver interrupts (except BRKDT)
- NRZ (non-return-to-zero) format
- 13 SCI module control registers located in the control register frame beginning at address 7050h

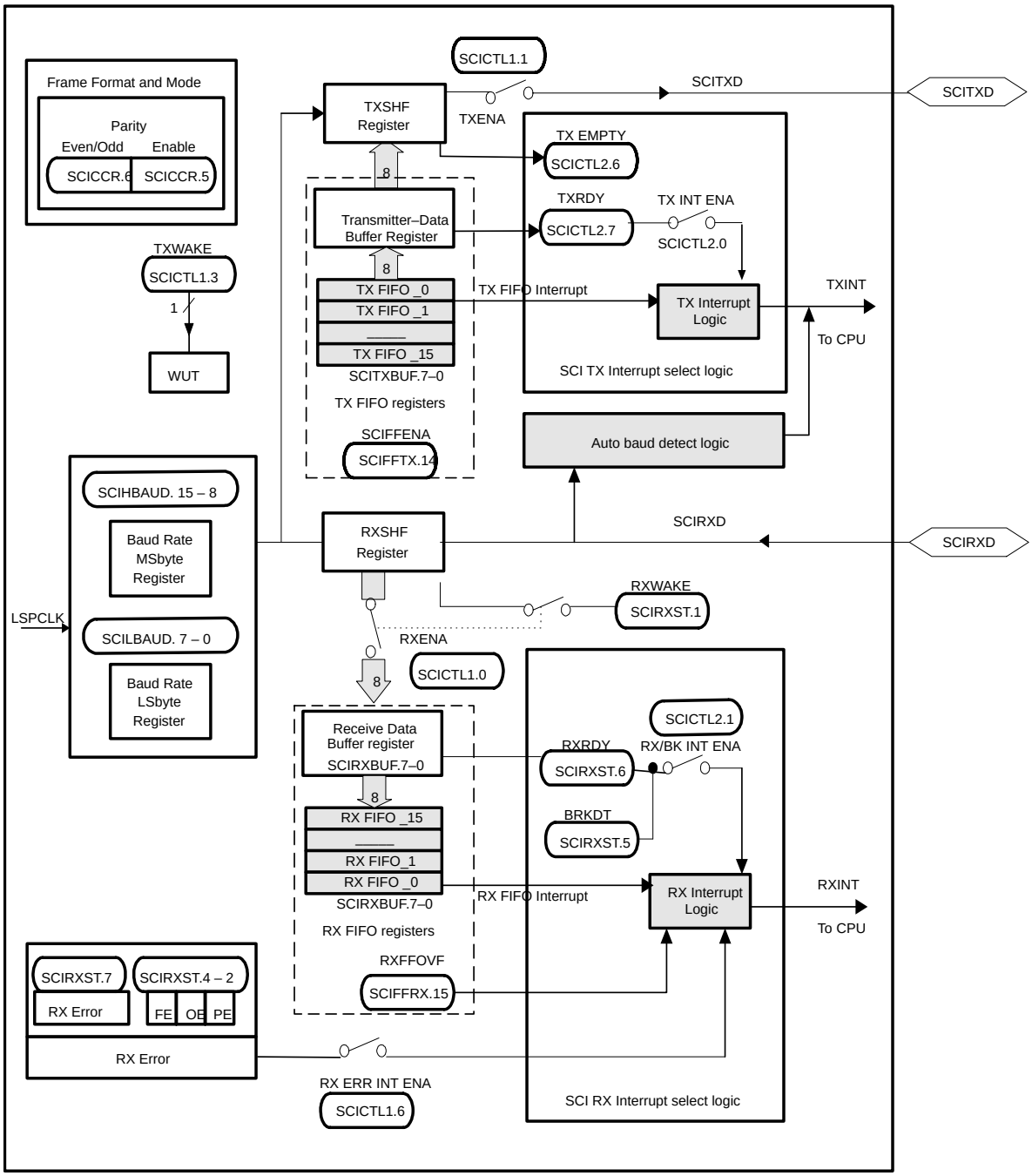
  All registers in this module are 8-bit registers that are connected to Peripheral Frame 2. When a register is accessed, the register data is in the lower byte (7–0), and the upper byte (15–8) is read as zeros. Writing to the upper byte has no effect.

Enhanced features:

- Auto-baud-detect hardware logic
- 16-level transmit/receive FIFO

Figure 1–2 shows the SCI module block diagram.

Figure 1–2. Serial Communications Interface (SCI) Module Block Diagram



The SCI port operation is configured and controlled by the registers listed in Table 1–1 and Table 1–2.

Table 1–1. SCI-A Registers

Name	Address Range	Size (x16)	Description
SCICCR	0x0000–7050	1	SCI-A Communications Control Register
SCICTL1	0x0000–7051	1	SCI-A Control Register 1
SCIHBAUD	0x0000–7052	1	SCI-A Baud Register, High Bits
SCILBAUD	0x0000–7053	1	SCI-A Baud Register, Low Bits
SCICTL2	0x0000–7054	1	SCI-A Control Register 2
SCIRXST	0x0000–7055	1	SCI-A Receive Status Register
SCIRXEMU	0x0000–7056	1	SCI-A Receive Emulation Data Buffer Register
SCIRXBUF	0x0000–7057	1	SCI-A Receive Data Buffer Register
SCITXBUF	0x0000–7059	1	SCI-A Transmit Data Buffer Register
SCIFFTX	0x0000–705A	1	SCI-A FIFO Transmit Register
SCIFFRX	0x0000–705B	1	SCI-A FIFO Receive Register
SCIFFCT	0x0000–705C	1	SCI-A FIFO Control Register
SCIPRI	0x0000–705F	1	SCI-A Priority Control Register

Table 1–2. SCI-B Registers

Name	Address Range	Size (x16)	Description
SCICCR	0x0000–7750	1	SCI-B Communications Control Register
SCICTL1	0x0000–7751	1	SCI-B Control Register 1
SCIHBAUD	0x0000–7752	1	SCI-B Baud Register, High Bits
SCILBAUD	0x0000–7753	1	SCI-B Baud Register, Low Bits
SCICTL2	0x0000–7754	1	SCI-B Control Register 2
SCIRXST	0x0000–7755	1	SCI-B Receive Status Register
SCIRXEMU	0x0000–7756	1	SCI-B Receive Emulation Data Buffer Register
SCIRXBUF	0x0000–7757	1	SCI-B Receive Data Buffer Register

Notes: 1) The registers are mapped to peripheral frame 2. This frame allows only 16-bit accesses. Using 32-bit accesses will produce undefined results.

2) SCIB is an optional peripheral. In some devices this may not be present. See the device-specific data sheet for peripheral availability.

Table 1–2. SCI-B Registers (Continued)

Name	Address Range	Size (x16)	Description
SCITXBUF	0x0000–7759	1	SCI-B Transmit Data Buffer Register
SCIFFTX	0x0000–775A	1	SCI-B FIFO Transmit Register
SCIFFRX	0x0000–775B	1	SCI-B FIFO Receive Register
SCIFFCT	0x0000–775C	1	SCI-B FIFO Control Register
SCIPRI	0x0000–775F	1	SCI-B Priority Control Register

Notes: 1) The registers are mapped to peripheral frame 2. This frame allows only 16-bit accesses. Using 32-bit accesses will produce undefined results.

2) SCIB is an optional peripheral. In some devices this may not be present. See the device-specific data sheet for peripheral availability.

1.2 Architecture

The major elements used in full-duplex operation are shown in Figure 1–2 and include:

- A transmitter (TX) and its major registers (upper half of Figure 1–2)
  - SCITXBUF — transmitter data buffer register. Contains data (loaded by the CPU) to be transmitted
  - TXSHF register — transmitter shift register. Accepts data from register SCITXBUF and shifts data onto the SCITXD pin, one bit at a time
- A receiver (RX) and its major registers (lower half of Figure 1–2)
  - RXSHF register — receiver shift register. Shifts data in from SCIRXD pin, one bit at a time
  - SCIRXBUF — receiver data buffer register. Contains data to be read by the CPU. Data from a remote processor is loaded into register RXSHF and then into registers SCIRXBUF and SCIRXEMU
- A programmable baud generator
- Data-memory-mapped control and status registers

The SCI receiver and transmitter can operate either independently or simultaneously.

1.2.1 SCI Module Signal Summary

Signal Name	Description
External signals	
SCIRXD	SCI Asynchronous Serial Port receive data
SCITXD	SCI Asynchronous Serial Port transmit data
Control	
Baud clock	LSPCLK Prescaled clock
Interrupt signals	
TXINT	Transmit interrupt
RXINT	Receive Interrupt

1.2.2 Multiprocessor and Asynchronous Communication Modes

The SCI has two multiprocessor protocols, the *idle-line* multiprocessor mode (see section 1.2.5 on page 1-10) and the *address-bit* multiprocessor mode (see section 1.2.6 on page 1-13). These protocols allow efficient data transfer between multiple processors.

The SCI offers the universal asynchronous receiver/transmitter (UART) communications mode for interfacing with many popular peripherals. The asynchronous mode (see section 1.2.7 on page 1-14) requires two lines to interface with many standard devices such as terminals and printers that use RS-232-C formats. Data transmission characteristics include:

- One start bit
- One to eight data bits
- An even/odd parity bit or no parity bit
- One or two stop bits

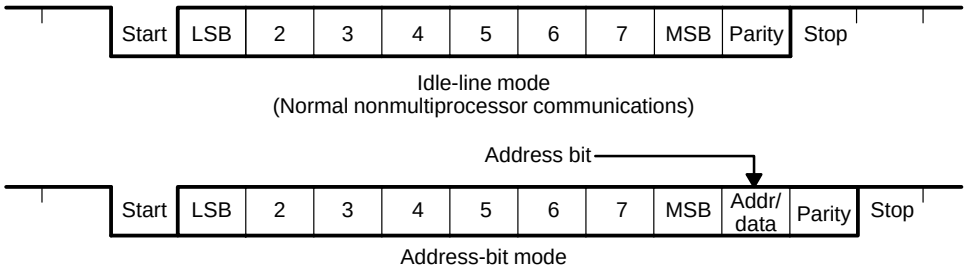
1.2.3 SCI Programmable Data Format

SCI data, both receive and transmit, is in NRZ (non-return-to-zero) format. The NRZ data format, shown in Figure 1–3, consists of:

- One start bit
- One to eight data bits
- An even/odd parity bit (optional)
- One or two stop bits
- An extra bit to distinguish addresses from data (address-bit mode only)

The basic unit of data is called a character and is one to eight bits in length. Each character of data is formatted with a start bit, one or two stop bits, and optional parity and address bits. A character of data with its formatting information is called a frame and is shown in Figure 1–3.

Figure 1–3. Typical SCI Data Frame Formats



To program the data format, use the SCICCR register. The bits used to program the data format are shown in Table 1–3.

Table 1–3. Programming the Data Format Using SCICCR

Bit(s)	Bit Name	Designation	Functions
2–0	SCI CHAR2–0	SCICCR.2:0	Select the character (data) length (one to eight bits).
5	PARITY ENABLE	SCICCR.5	Enables the parity function if set to 1, or disables the parity function if cleared to 0.
6	EVEN/ODD PARITY	SCICCR.6	If parity is enabled, selects odd parity if cleared to 0 or even parity if set to 1.
7	STOP BITS	SCICCR.7	Determines the number of stop bits transmitted—one stop bit if cleared to 0 or two stop bits if set to 1.

1.2.4 SCI Multiprocessor Communication

The multiprocessor communication format allows one processor to efficiently send blocks of data to other processors on the same serial link. On one serial line, there should be only one transfer at a time. In other words, there can be only one talker on a serial line at a time.

Address Byte

The *first byte* of a block of information that the talker sends contains an *address byte* that is read by all listeners. Only listeners with the correct address can be interrupted by the data bytes that follow the address byte. The listeners with an incorrect address remain uninterrupted until the next address byte.

Sleep Bit

All processors on the serial link set the SCI SLEEP bit (bit 2 of SCICTL1) to 1 so that they are interrupted only when the address byte is detected. When a processor reads a block address that corresponds to the CPU device address as set by your application software, your program must clear the SLEEP bit to enable the SCI to generate an interrupt on receipt of each data byte.

Although the receiver still operates when the SLEEP bit is 1, it does not set RXRDY, RXINT, or any of the receiver error status bits to 1 unless the address byte is detected and the address bit in the received frame is a 1 (applicable to address-bit mode). The SCI does not alter the SLEEP bit; your software must alter the SLEEP bit.

1.2.4.1 Recognizing the Address Byte

A processor recognizes an address byte differently, depending on the multiprocessor mode used. For example:

- The *idle-line mode* (section 1.2.5 on page 1-10) leaves a quiet space before the address byte. This mode does not have an extra address/data bit and is more efficient than the address-bit mode for handling blocks that contain more than ten bytes of data. The idle-line mode should be used for typical non-multiprocessor SCI communication.
- The *address-bit mode* (section 1.2.6 on page 1-13) adds an extra bit (that is, an address bit) into every byte to distinguish addresses from data. This mode is more efficient in handling many small blocks of data because, unlike the idle mode, it does not have to wait between blocks of data. However, at a high transmit speed, the program is not fast enough to avoid a 10-bit idle in the transmission stream.

1.2.4.2 Controlling the SCI TX and RX Features

The multiprocessor mode is software selectable via the ADDR/IDLE MODE bit (SCICCR, bit 3). Both modes use the TXWAKE flag bit (SCICTL1, bit 3), RXWAKE flag bit (SCIRXST, bit1), and the SLEEP flag bit (SCICTL1, bit 2) to control the SCI transmitter and receiver features of these modes.

1.2.4.3 Receipt Sequence

In both multiprocessor modes, the receive sequence is:

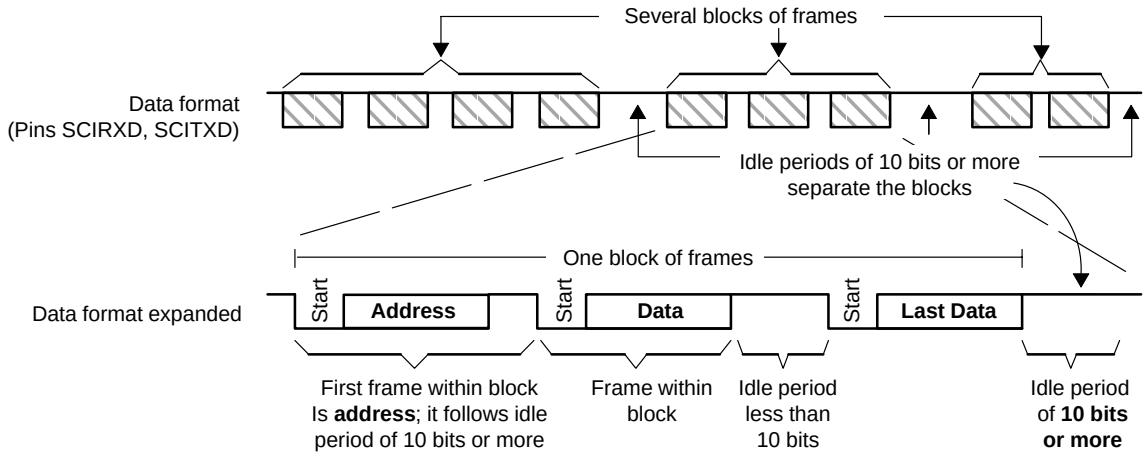
1) At the receipt of an address block, the SCI port wakes up and requests an interrupt (bit number 1 RX/BK INT ENA-of SCICTL2 must be enabled to request an interrupt). It reads the first frame of the block, which contains the destination address.
2) A software routine is entered through the interrupt and checks the incoming address. This address byte is checked against its device address byte stored in memory.
3) If the check shows that the block is addressed to the device CPU, the CPU clears the SLEEP bit and reads the rest of the block; if not, the software routine exits with the SLEEP bit still set and does not receive interrupts until the next block start.

1.2.5 Idle-Line Multiprocessor Mode

In the idle-line multiprocessor protocol (ADDR/IDLE MODE bit=0), blocks are separated by having a longer idle time between the blocks than between frames in the blocks. An idle time of ten or more high-level bits after a frame indicates the start of a new block. The time of a single bit is calculated directly from the baud value (bits per second). The idle-line multiprocessor commu-

nication format is shown in Figure 1–4 (ADDR/IDLE MODE bit is bit 3 of SCICCR).

Figure 1–4. Idle-Line Multiprocessor Communication Format



1.2.5.1 Idle-Line Mode Steps

The steps followed by the idle-line mode:

1) SCI wakes up after receipt of the block-start signal.

2) The processor recognizes the next SCI interrupt.

3) The interrupt service routine compares the received address (sent by a remote transmitter) to its own.

4) If the CPU *is being addressed*, the service routine clears the SLEEP bit and receives the rest of the data block.

5) If the CPU *is not being addressed*, the SLEEP bit remains set. This lets the CPU continue to execute its main program without being interrupted by the SCI port until the next detection of a block start.

1.2.5.2 Block Start Signal

There are two ways to send a block-start signal:

- ❏ Method 1: Deliberately leave an idle time of ten bits or more by delaying the time between the transmission of the last frame of data in the previous block and the transmission of the address frame of the new block.

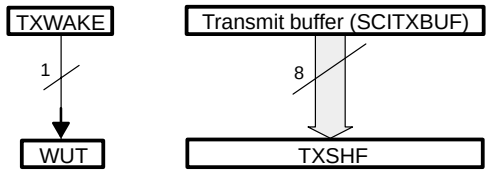
- ❏ Method 2: The SCI port first sets the TXWAKE bit (SCICTL1, bit 3) to 1 before writing to the SCITXBUF register. This sends an idle time of exactly

11 bits. In this method, the serial communications line is not idle any longer than necessary. (A don't care byte has to be written to SCITXBUF after setting TXWAKE, and before sending the address, so as to transmit the idle time.)

1.2.5.3 Wake-UP Temporary (WUT) Flag

Associated with the TXWAKE bit is the wake-up temporary (WUT) flag. WUT is an internal flag, double-buffered with TXWAKE. When TXSHF is loaded from SCITXBUF, WUT is loaded from TXWAKE, and the TXWAKE bit is cleared to 0. This arrangement is shown in Figure 1–5.

Figure 1–5. Double-Buffered WUT and TXSHF



Note: WUT = wake-up temporary

Sending a Block Start Signal

To send out a block-start signal of exactly one frame time during a sequence of block transmissions:

1) Write a 1 to the TXWAKE bit.

2) Write a data word (content not important: a *don't care*) to the SCITXBUF register (transmit data buffer) to send a block-start signal. (The first data word written is suppressed while the block-start signal is sent out and ignored after that.) When the TXSHF (transmit shift register) is free again, SCITXBUF contents are shifted to TXSHF, the TXWAKE value is shifted to WUT, and then TXWAKE is cleared.

   Because TXWAKE was set to a 1, the start, data, and parity bits are replaced by an idle period of 11 bits transmitted following the last stop bit of the previous frame.

3) Write a new address value to SCITXBUF.

A *don't-care* data word must first be written to register SCITXBUF so that the TXWAKE bit value can be shifted to WUT. After the don't-care data word is shifted to the TXSHF register, the SCITXBUF (and TXWAKE if necessary) can be written to again because TXSHF and WUT are both double-buffered.

1.2.5.4 Receiver Operation

The receiver operates regardless of the SLEEP bit. However, the receiver neither sets RXRDY nor the error status bits, nor does it request a receive interrupt *until an address frame is detected*.

1.2.6 Address-Bit Multiprocessor Mode

In the address-bit protocol (ADDR/IDLE MODE bit=1), frames have an extra bit called an address bit that immediately follows the last data bit. The address bit is set to 1 in the first frame of the block and to 0 in all other frames. The idle period timing is irrelevant (see Figure 1–6, ADDR/IDLE MODE bit in SCICCR, bit 3).

1.2.6.1 Sending an Address

The TXWAKE bit value is placed in the address bit. During transmission, when the SCITXBUF register and TXWAKE are loaded into the TXSHF register and WUT respectively, TXWAKE is reset to 0 and WUT becomes the value of the address bit of the current frame. Thus, to send an address:

1) Set the TXWAKE bit to 1 and write the appropriate address value to the SCITXBUF register.

   When this address value is transferred to the TXSHF register and shifted out, its address bit is sent as a 1. This flags the other processors on the serial link to read the address.

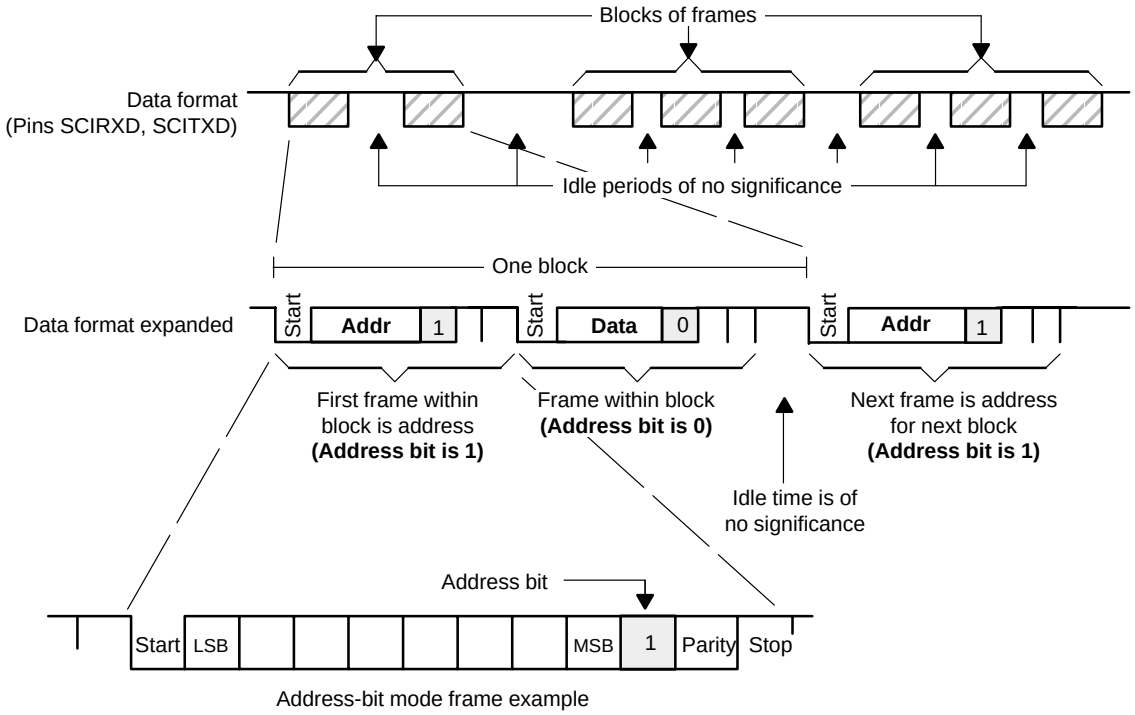
2) Write to SCITXBUF and TXWAKE after TXSHF and WUT are loaded. (Can be written to immediately since both TXSHF and WUT are both double-buffered.

3) Leave the TXWAKE bit set to 0 to transmit non-address frames in the block.

Note: The Address-bit format is for transfers of 11 bytes or less

As a general rule, the address-bit format is typically used for data frames of 11 bytes or less. This format adds one bit value (1 for an address frame, 0 for a data frame) to all data bytes transmitted. The idle-line format is typically used for data frames of 12 bytes or more.

Figure 1–6. Address-Bit Multiprocessor Communication Format



1.2.7 SCI Communication Format

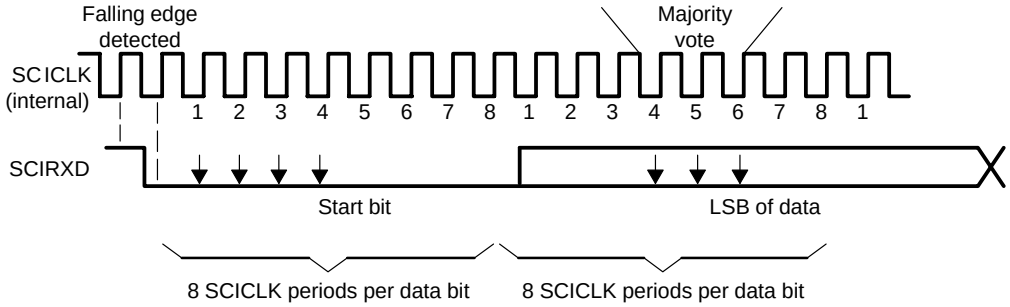
The SCI asynchronous communication format uses either single line (one way) or two line (two way) communications. In this mode, the frame consists of a start bit, one to eight data bits, an optional even/odd parity bit, and one or two stop bits (shown in Figure 1–7). There are *eight SCICLK periods* per data bit.

The receiver begins operation on receipt of a valid start bit. A valid start bit is identified by four consecutive internal SCICLK periods of zero bits as shown in Figure 1–7. If any bit is not zero, then the processor starts over and begins looking for another start bit.

For the bits following the start bit, the processor determines the bit value by making three samples in the middle of the bits. These samples occur on the fourth, fifth, and sixth SCICLK periods, and bit-value determination is on a majority (two out of three) basis. Figure 1–7 illustrates the asynchronous communication format for this with a start bit showing how edges are found and where a majority vote is taken.

Since the receiver synchronizes itself to frames, the external transmitting and receiving devices do not have to use a synchronized serial clock. The clock can be generated locally.

Figure 1–7. SCI Asynchronous Communications Format

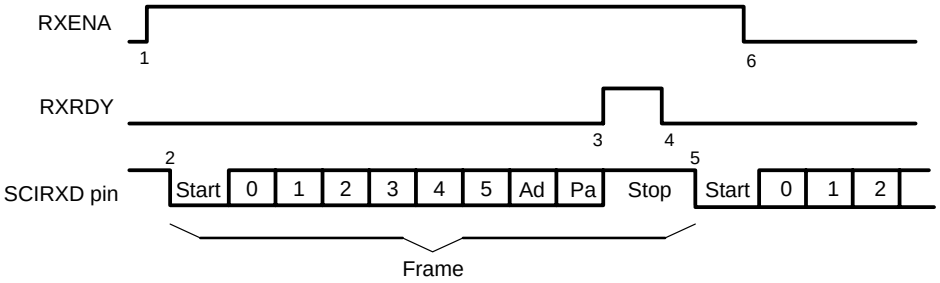


1.2.7.1 Receiver Signals in Communication Modes

Figure 1–8 illustrates an example of receiver signal timing that assumes the following conditions:

- ❑ Address-bit wake-up mode (address bit does not appear in idle-line mode)
- ❑ Six bits per character

Figure 1–8. SCI RX Signals in Communication Modes



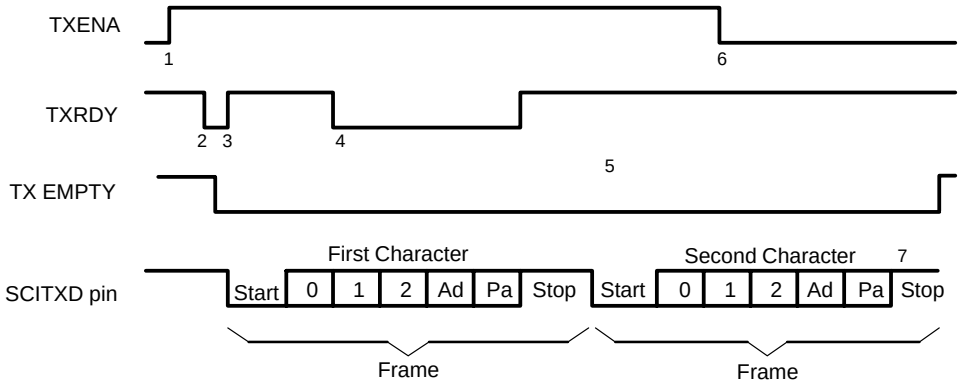
Notes: 1) Flag bit RXENA (SCICTL1, bit 0) goes high to enable the receiver.
2) Data arrives on the SCIRXD pin, start bit detected.
3) Data is shifted from RXSHF to the receiver buffer register (SCIRXBUF); an interrupt is requested. Flag bit RXRDY (SCIRXST, bit 6) goes high to signal that a new character has been received.
4) The program reads SCIRXBUF; flag RXRDY is automatically cleared.
5) The next byte of data arrives on the SCIRXD pin; the start bit is detected, then cleared.
6) Bit RXENA is brought low to disable the receiver. Data continues to be assembled in RXSHF but is not transferred to the receiver buffer register.

1.2.7.2 Transmitter Signals in Communication Modes

Figure 1–9 illustrates an example of transmitter signal timing that assumes the following conditions:

- ❑ Address-bit wake-up mode (address bit does not appear in idle-line mode)
- ❑ Three bits per character

Figure 1–9. SCI TX Signals in Communications Mode



Notes: 1) Bit TXENA (SCICTL1, bit 1) goes high, enabling the transmitter to send data.

2) SCITXBUF is written to; thus, (1) the transmitter is no longer empty, and (2) TXRDY goes low.

3) The SCI transfers data to the shift register (TXSHF). The transmitter is ready for a second character (TXRDY goes high), and it requests an interrupt (to enable an interrupt, bit TX INT ENA — SCICTL2, bit 0 — must be set).

4) The program writes a second character to SCITXBUF after TXRDY goes high (item 3). (TXRDY goes low again after the second character is written to SCITXBUF.)

5) Transmission of the first character is complete. Transfer of the second character to shift register TXSHF begins.

6) Bit TXENA goes low to disable the transmitter; the SCI finishes transmitting the current character.

7) Transmission of the second character is complete; transmitter is empty and ready for new character.

1.2.8 SCI Port Interrupts

The SCI receiver and transmitter can be interrupt controlled. The SCICTL2 register has one flag bit (TXRDY) that indicates active interrupt conditions, and the SCIRXST register has two interrupt flag bits (RXRDY and BRKDT), plus the RX ERROR interrupt flag which is a logical OR of the FE, OE and PE conditions. The transmitter and receiver have separate interrupt-enable bits. When not enabled, the interrupts are not asserted; however, the condition flags remain active, reflecting transmission and receipt status.

The SCI has independent peripheral interrupt vectors for the receiver and transmitter. Peripheral interrupt requests can be either high priority or low priority. This is indicated by the priority bits which are output from the peripheral to the PIE controller. When both RX and TX interrupt requests are made at the same priority level, the receiver always has higher priority than the transmitter, reducing the possibility of receiver overrun.

The operation of peripheral interrupts is described in the peripheral interrupt expansion controller chapter of the *TMS320F28x System Control and Interrupts Peripheral Reference Guide* (literature number SPRU078).

- If the RX/BK INT ENA bit (SCICTL2, bit 1) is set, the receiver peripheral interrupt request is asserted when one of the following events occurs:
  - The SCI receives a complete frame and transfers the data in the RXSHF register to the SCIRXBUF register. This action sets the RXRDY flag (SCIRXST, bit 6) and initiates an interrupt.
  - A break detect condition occurs (the SCIRXD is low for ten bit periods following a missing stop bit). This action sets the BRKDT flag bit (SCIRXST, bit 5) and initiates an interrupt.
- If the TX INT ENA bit (SCICTL2.0) is set, the transmitter peripheral interrupt request is asserted whenever the data in the SCITXBUF register is transferred to the TXSHF register, indicating that the CPU can write to SCITXBUF; this action sets the TXRDY flag bit (SCICTL2, bit 7) and initiates an interrupt.

Note:

Interrupt generation due to the RXRDY and BRKDT bits is controlled by the RX/BK INT ENA bit (SCICTL2, bit 1). Interrupt generation due to the RX ERROR bit is controlled by the RX ERR INT ENA bit (SCICTL1, bit 6).

1.2.9 SCI Baud Rate Calculations

The internally generated serial clock is determined by the low-speed peripheral clock LSPCLK) and the baud-select registers. The SCI uses the 16-bit value of the baud-select registers to select one of the 64K different serial clock rates possible for a given LSPCLK.

See the bit descriptions in section 2.4, *Baud-Select Registers*, on page 2-8 for the formula to use when calculating the SCI asynchronous baud. Table 1–4 shows the baud-select values for common SCI bit rates.

Table 1–4. Asynchronous Baud Register Values for Common SCI Bit Rates

	LSPCLK Clock Frequency, 37.5 MHz		
Ideal Baud	**BRR**	**Actual Baud**	**% Error**
2400	1952 (7A0h)	2400	0
4800	976 (3D0h)	4798	–0.04
9600	487 (1E7h)	9606	0.06
19200	243 (F3h)	19211	0.06
38400	121 (79h)	38422	0.06

1.2.10 SCI Enhanced Features

The 28x SCI features autobaud detection and transmit/receive FIFO. The following section explains the FIFO operation.

1.2.10.1 SCI FIFO Description

The following steps explain the FIFO features and help with programming the SCI with FIFOs.

1) Reset. At reset the SCI powers up in standard SCI mode and the FIFO function is disabled. The FIFO registers SCIFFTX, SCIFFRX, and SCIFFCT remain inactive.

2) Standard SCI. The standard F24x SCI modes will work normally with TXINT/RXINT interrupts as the interrupt source for the module.

3) FIFO enable. FIFO mode is enabled by setting the SCIFFEN bit in the SCIFFTX register. SCIRST can reset the FIFO mode at any stage of its operation.

4) Active registers. All the SCI registers and SCI FIFO registers (SCIFFTX, SCIFFRX, and SCIFFCT) are active.

5) Interrupts. FIFO mode has two interrupts; one for transmit FIFO, TXINT and one for receive FIFO,RXINT. RXINT is the common interrupt for SCI FIFO receive, receive error, and receive FIFO overflow conditions. The TXINT of the standard SCI will be disabled and this interrupt will service as SCI transmit FIFO interrupt.

6) Buffers. Transmit and receive buffers are supplemented with two 16 level FIFOs. The transmit FIFO registers are 8 bits wide and receive FIFO registers are 10 bits wide. The one word transmit buffer of the standard SCI functions as a transition buffer between the transmit FIFO and shift register. The one word transmit buffer is loaded from transmit FIFO only after the last bit of the shift register is shifted out. With the FIFO enabled, TXSHF is directly loaded after an optional delay value (SCIFFCT), TXBUF is not used.

7) Delayed transfer. The rate at which words in the FIFO are transferred to the transmit shift register is programmable. The SCIFFCT register bits (7–0) FFTXDLY7–FFTXDLY0 define the delay between the word transfer. The delay is defined in the number SCI baud clock cycles. The 8 bit register can define a minimum delay of 0 baud clock cycles and a maximum of 256-baud clock cycles. With zero delay, the SCI module can transmit data in continuous mode with the FIFO words shifting out back to back. With

the 256 clock delay the SCI module can transmit data in a maximum delayed mode with the FIFO words shifting out with a delay of 256 baud clocks between each words. The programmable delay facilitates communication with slow SCI/UARTs with little CPU intervention.

8) FIFO status bits. Both the transmit and receive FIFOs have status bits TXFFST or RXFFST (bits 12– 0) that define the number of words available in the FIFOs at any time. The transmit FIFO reset bit TXFIFO and receive reset bit RXFIFO reset the FIFO pointers to zero when these bits *are cleared to 0*. The FIFOs resumes operation from start once these bits are *set to one.*

9) Programmable interrupt levels. Both transmit and receive FIFO can generate CPU interrupts. The interrupt trigger is generated whenever the transmit FIFO status bits TXFFST (bits 12–8) match (less than or equal to) the interrupt trigger level bits TXFFIL (bits 4–0). This provides a programmable interrupt trigger for transmit and receive sections of the SCI. Default value for these trigger level bits will be 0x11111 for receive FIFO and 0x00000 for transmit FIFO respectively.

   Figure 1–10 and Table 1–5 explain the operation/configuration of SCI interrupts in nonFIFO/FFO mode.

Figure 1–10. SCI FIFO Interrupt Flags and Enable Logic

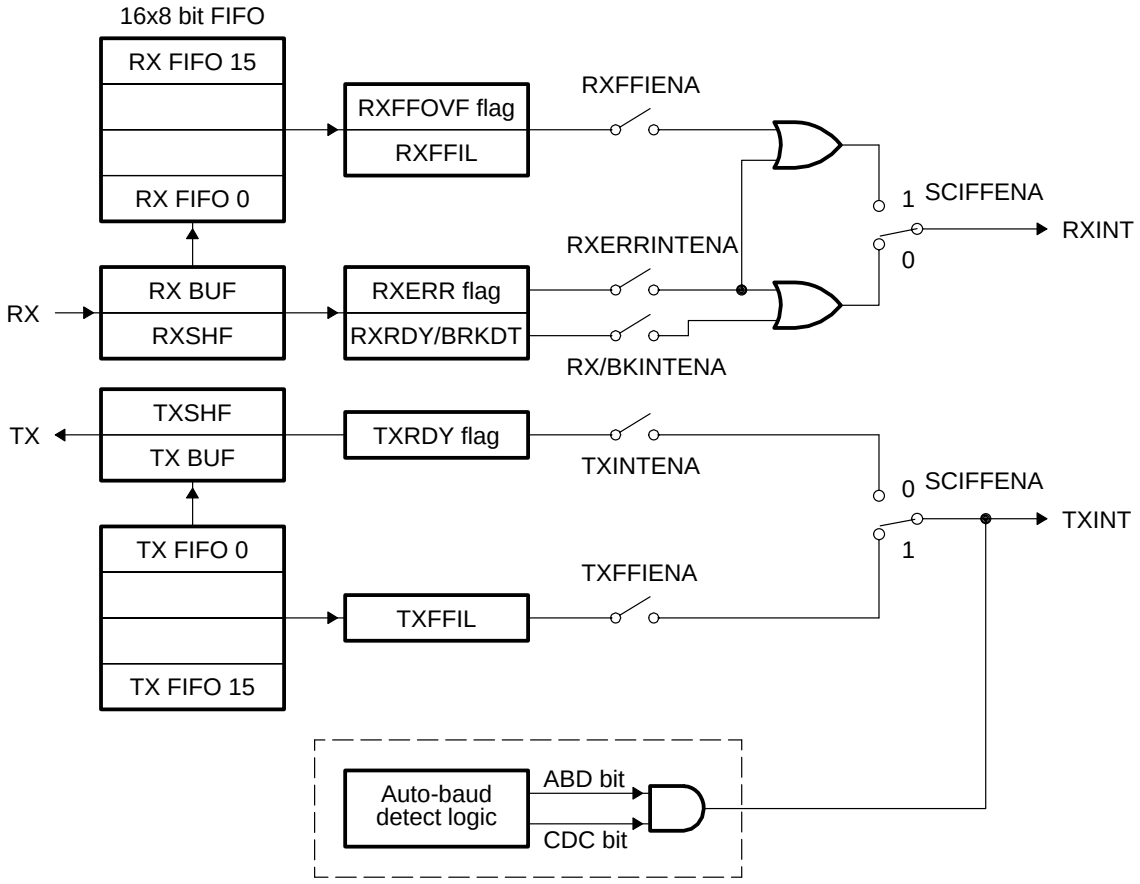


Table 1–5. SCI Interrupt Flags

FIFO Options	SCI Interrupt Source	Interrupt Flags	Interrupt Enables	FIFO Enable SCIFFENA	Interrupt line
SCI without FIFO	Receive error	RXERR	RXERRINTENA	0	RXINT
	Receive break	BRKDT	RX/BKINTENA	0	RXINT
	Data receive	RXRDY	RX/BKINTENA	0	RXINT
	Transmit empty	TXRDY	TXINTENA	0	TXINT
SCI with FIFO	Receive error and receive break	RXERR	RXERRINTENA	1	RXINT
	FIFO receive	RXFFIL	RXFFIENA	1	RXINT
	Transmit empty	TXFFIL	TXFFIENA	1	TXINT
Auto-baud	Auto-baud detected	ABD	Don't care	x	TXINT

Notes: 1) RXERR can be set by BRKDT, FE, OE, PE flags. In FIFO mode, BRKDT interrupt is only through RXERR flag
2) FIFO mode TXSHF is directly loaded after delay value, TXBUF is not used.

1.2.10.2 SCI Auto-Baud

Most SCI modules do not have an auto-baud detect logic built-in hardware. These SCI modules are integrated with embedded controllers whose clock rates are dependent on PLL reset values. Often embedded controller clocks change after final design. In the enhanced feature set this module supports an autobaud-detect logic in hardware. The following section explains the enabling sequence for autobaud-detect feature.

1.2.10.3 Autobaud-Detect Sequence

Bits ABD and CDC in SCIFFCT control the autobaud logic. The SCIRST bit should be enabled to make autobaud logic work.

If ABD is set while CDC is 1, which indicates auto-baud alignment, SCI transmit FIFO interrupt will occur (TXINT). After the interrupt service CDC bit has to be cleared by software. If CDC remains set even after interrupt service, there should be no repeat interrupts.

Step 1: Enable autobaud-detect mode for the SCI by setting the CDC bit (bit 13) in SCIFFCT and clearing the ABD bit (Bit 15) by writing a 1 to ABDCLR bit (bit 14).

Step 2: Initialize the baud register to be 1 or less than a baud rate limit of 500 Kbps.

Step 3: Allow SCI to receive either character ”A” or ”a” from a host at the desired baud rate. If the first character is either ”A” or ”a”. the autobaud-detect hardware will detect the incoming baud rate and set the ABD bit.

Step 4: The auto-detect hardware will update the baud rate register with the equivalent baud value hex. The logic will also generate an interrupt to the CPU.

Step 5: Respond to the interrupt clear ADB bit by writing a 1 to ABD CLR (bit 14) of SCIFFCT register and disable further autobaud locking by clearing CDC bit by writing a 0.

Step 6: Read the receive buffer for character ”A” or ”a” to empty the buffer and buffer status.

Step 7: If ABD is set while CDC is 1, which indicates autobaud alignment, the SCI transmit FIFO interrupt will occur (TXINT). After the interrupt service CDC bit must be cleared by software.

Note:

At higher baud rates, the slew rate of the incoming data bits can be affected by transceiver and connector performance. While normal serial communications may work well, this slew rate may limit reliable autobaud detection at higher baud rates (typically beyond 100k baud) and cause the auto-baud-lock feature to fail.

To avoid this, the followng is recommended:

- ❑ Achieve a baud-lock between the host and 28x SCI boot loader using a lower baud rate.
- ❑ The host may then handshake with the loaded 28x application to set the SCI baud rate register to the desired higher baud rate.

Chapter 2

SCI Registers

The functions of the SCI are software configurable. Sets of control bits, organized into dedicated bytes, are programmed to initialize the desired SCI communications format. This includes operating mode and protocol, baud value, character length, even/odd parity or no parity, number of stop bits, and interrupt priorities and enables.

Topic **Page**

2.1 SCI Module Register Summary

The SCI is controlled and accessed through registers listed in Table 2–1 and Table 2–2, which are described in the sections that follow.

Table 2–1. SCIA Registers

Register Mnemonic	Address	Number of Bits	Description
SCICCR	0x0000–7050	1	SCI-A Communications Control Register
SCICTL1	0x0000–7051	1	SCI-A Control Register 1
SCIHBAUD	0x0000–7052	1	SCI-A Baud Register, High Bits
SCILBAUD	0x0000–7053	1	SCI-A Baud Register, Low Bits
SCICTL2	0x0000–7054	1	SCI-A Control Register 2
SCIRXST	0x0000–7055	1	SCI-A Receive Status Register
SCIRXEMU	0x0000–7056	1	SCI-A Receive Emulation Data Buffer Register
SCIRXBUF	0x0000–7057	1	SCI-A Receive Data Buffer Register
SCITXBUF	0x0000–7059	1	SCI-A Transmit Data Buffer Register
SCIFFTX	0x0000–705A	1	SCI-A FIFO Transmit Register
SCIFFRX	0x0000–705B	1	SCI-A FIFO Receive Register
SCIFFCT	0x0000–705C	1	SCI-A FIFO Control Register
SCIPRI	0x0000–705F	1	SCI-A Priority Control Register

Note: The shaded registers operate in enhanced mode.

Table 2–2. SCIB Registers

Name	Address Range	Number of Bits	Description
SCICCR	0x0000–7750	1	SCI-B Communications Control Register
SCICTL1	0x0000–7751	1	SCI-B Control Register 1
SCIHBAUD	0x0000–7752	1	SCI-B Baud Register, High Bits
SCILBAUD	0x0000–7753	1	SCI-B Baud Register, Low Bits
SCICTL2	0x0000–7754	1	SCI-B Control Register 2
SCIRXST	0x0000–7755	1	SCI-B Receive Status Register
SCIRXEMU	0x0000–7756	1	SCI-B Receive Emulation Data Buffer Register
SCIRXBUF	0x0000–7757	1	SCI-B Receive Data Buffer Register
SCITXBUF	0x0000–7759	1	SCI-B Transmit Data Buffer Register
SCIFFTX	0x0000–775A	1	SCI-B FIFO Transmit Register
SCIFFRX	0x0000–775B	1	SCI-B FIFO Receive Register
SCIFFCT	0x0000–775C	1	SCI-B FIFO Control Register
SCIPRI	0x0000–775F	1	SCI-B Priority Control Register

2.2 SCI Communication Control Register (SCICCR)

SCICCR defines the character format, protocol, and communications mode used by the SCI.

Figure 2–1. SCI Communication Control Register (SCICCR) — Address 7050h

7	6	5	4	3	2	1	0
STOP BITS	EVEN/ODD PARITY	PARITY ENABLE	LOOPBACK ENA	ADDR/IDLE MODE	SCICHAR2	SCICHAR1	SCICHAR0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

Legend: R = Read access, W = Write access, -0 = value after reset

Bit(s)	Name	Description
7	STOP BITS	SCI number of stop bits. This bit specifies the number of stop bits transmitted. The receiver checks for only one stop bit. 1 Two stop bits 0 One stop bit
6	PARITY	SCI parity odd/even selection. If the PARITY ENABLE bit (SCICCR, bit 5) is set, PARITY (bit 6) designates odd or even parity (odd or even number of bits with the value of 1 in both transmitted and received characters). 1 Even parity 0 Odd parity
5	PARITY ENABLE	SCI parity enable. This bit enables or disables the parity function. If the SCI is in the address-bit multiprocessor mode (set using bit 3 of this register), the address bit is included in the parity calculation (if parity is enabled). For characters of less than eight bits, the remaining unused bits should be masked out of the parity calculation. 1 Parity is enabled 0 Parity disabled; no parity bit is generated during transmission or is expected during reception
4	LOOP BACK ENA	Loop Back test mode enable. This bit enables the Loop Back test mode where the Tx pin is internally connected to the Rx pin. 1 Loop Back test mode enabled 0 Loop Back test mode disabled

Figure 2–1.SCI Communication Control Register (SCICCR) — Address 7050h (Continued)

Bit(s)	Name	Description
3	ADDR/IDLE MODE	SCI multiprocessor mode control bit. This bit selects one of the multiprocessor protocols Multiprocessor communication is different from the other communication modes because it uses SLEEP and TXWAKE functions (bits SCICTL1, bit 2 and SCICTL1, bit 3, respectively). The idle-line mode is usually used for normal communications because the address-bit mode adds an extra bit to the frame. The idle-line mode does not add this extra bit and is compatible with RS-232 type communications. 1 Address-bit mode protocol selected 0 Idle-line mode protocol selected
2–0	SCI CHAR2–0	Character-length control bits 2 – 0. These bits select the SCI character length from one to eight bits. Characters of less than eight bits are right-justified in SCIRXBUF and SCIRXEMU and are padded with leading zeros in SCIRXBUF. SCITXBUF *doesn't need to be padded with leading zeros.* The bit values and character lengths for SCI CHAR2-0 bits are as follows:

SCI CHAR2–0 Bit Values (Binary)

SCI CHAR2	SCI CHAR1	SCI CHAR0	Character Length (Bits)
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

2.3 SCI Control Register 1 (SCICTL1)

SCICTL1 controls the receiver/transmitter enable, TXWAKE and SLEEP functions, and the SCI software reset.

Figure 2–2. SCI Control Register 1 (SCICTL1) — Address 7051h

7	6	5	4	3	2	1	0
Reserved	RX ERR INT ENA	SW RESET	Reserved	TXWAKE	SLEEP	TXENA	RXENA
R-0	R/W-0	R/W-0	R-0	R/S-0	R/W-0	R/W-0	R/W-0

Legend: R = Read access, W = Write access, -0 = value after reset

Bit(s)	Name	Description
7	Reserved	Reads return zero; writes have no effect.
6	RX ERR INT ENA	SCI receive error interrupt enable. Setting this bit enables an interrupt if the RX ERROR bit (SCIRXST, bit 7) becomes set because of errors occurring. 1 Receive error interrupt enabled 0 Receive error interrupt disabled
5	SW RESET	SCI software reset (active low). Writing a 0 to this bit initializes the SCI state machines and operating flags (registers SCICTL2 and SCIRXST) to the reset condition. The SW RESET bit does not affect any of the configuration bits. All affected logic is held in the specified reset state until a 1 is written to SW RESET (the bit values following a reset are shown beneath each register diagram in this section). Thus, after a system reset, re-enable the SCI by writing a 1 to this bit. Clear this bit after a receiver break detect (BRKDT flag, bit SCIRXST, bit 5). SW RESET affects the operating flags of the SCI, but it neither affects the configuration bits nor restores the reset values. Once SW RESET is asserted, the flags are frozen until the bit is deasserted.

Figure 2–2.SCI Control Register 1 (SCICTL1) — Address 7051h (Continued)

The affected flags are as follows:

SCI Flag	Register Bit	Value After SW RESET
TXRDY	SCICTL2, bit 7	1
TX EMPTY	SCICTL2, bit 6	1
RXWAKE	SCIRXST, bit 1	0
PE	SCIRXST, bit 2	0
OE	SCIRXST, bit 3	0
FE	SCIRXST, bit 4	0
BRKDT	SCIRXST, bit 5	0
RXRDY	SCIRXST, bit 6	0
RX ERROR	SCIRXST, bit 7	0

4 Reserved — Reads return zero; writes have no effect.

3 TXWAKE — SCI transmitter wake-up method select. The TXWAKE bit controls selection of the data-transmit feature, depending on which transmit mode (idle-line or address-bit) is specified at the ADDR/IDLE MODE bit (SCICCR, bit 3)

1 Transmit feature selected is dependent on the mode, idle-line or address-bit:

0 Transmit feature is not selected
In *idle-line* mode: write a 1 to TXWAKE, then write data to register SCITXBUF to generate an idle period of 11 data bits
In *address-bit* mode: write a 1 to TXWAKE, then write data to SCITXBUF to set the address bit for that frame to 1

TXWAKE is not cleared by the SW RESET bit (SCICTL1, bit 5); it is cleared by a system reset or the transfer of TXWAKE to the WUT flag.

2 SLEEP — SCI sleep. The TXWAKE bit controls selection of the data-transmit feature, depending on which transmit mode (idle-line or address-bit) is specified at the ADDR/IDLE MODE bit (SCICCR, bit 3). In a multiprocessor configuration, this bit controls the receiver sleep function. Clearing this bit brings the SCI out of the sleep mode.

The receiver still operates when the SLEEP bit is set; however, operation does not update the receiver buffer ready bit (SCIRXST, bit 6, RXRDY) or the error status bits (SCIRXST, bit 5–2: BRKDT, FE, OE, and PE) unless the address byte is detected. SLEEP is *not* cleared when the address byte is detected.

1 Sleep mode enabled

0 Sleep mode disabled

Figure 2–2.SCI Control Register 1 (SCICTL1) — Address 7051h (Continued)

1	TXENA	SCI transmitter enable. Data is transmitted through the SCITXD pin only when TXENA is set. If reset, transmission is halted but only after all data previously written to SCITXBUF has been sent. 1 Transmitter enabled 0 Transmitter disabled
0	RXENA	SCI receiver enable. Data is received on the SCIRXD pin and is sent to the receiver shift register and then the receiver buffers. This bit enables or disables the receiver (transfer to the buffers). Clearing RXENA stops received characters from being transferred to the two receiver buffers and also stops the generation of receiver interrupts. However, the receiver shift register can continue to assemble characters. Thus, if RXENA is set during the reception of a character, the complete character will be transferred into the receiver buffer registers, SCIRXEMU and SCIRXBUF. 1 Send received characters to SCIRXEMU and SCIRXBUF 0 Prevent received characters from transfer into the SCIRXEMU and SCIRXBUF receiver buffers

2.4 SCI Baud-Select Registers (SCIHBAUD, SCILBAUD)

The values in SCIHBAUD and SCILBAUD specify the baud rate for the SCI.

Figure 2–3. Baud-Select MSbyte Register (SCIHBAUD) — Address 7052h

15	14	13	12	11	10	9	8
BAUD15 (MSB)	BAUD14	BAUD13	BAUD12	BAUD11	BAUD10	BAUD9	BAUD8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

Figure 2–4. Baud-Select LSbyte Register (SCILBAUD) — Address 7053h

7	6	5	4	3	2	1	0
BAUD7	BAUD6	BAUD5	BAUD4	BAUD3	BAUD2	BAUD1	BAUD0 (LSB)
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

Legend: R = Read access, W = Write access, -n = value after reset

Bit(s)	Name	Reset	Description
15–0	BAUD15–BAUD0	0	SCI 16-bit baud selection Registers SCIHBAUD (MSbyte) and SCILBAUD (LSbyte) are concatenated to form a 16-bit baud value, BRR. The internally-generated serial clock is determined by the low speed peripheral clock (LSPCLK) signal and the two baud-select registers. The SCI uses the 16-bit value of these registers to select one of 64K serial clock rates for the communication modes. The SCI baud rate is calculated using the following equation: $\text{SCI Asynchronous Baud} = \frac{\text{LSPCLK}}{(\text{BRR} + 1) \times 8}$ Alternatively, $\text{BRR} = \frac{\text{LSPCLK}}{\text{SCI Asynchronous Baud} \times 8} - 1$ Note that the above formulas are applicable only when $1 \le \text{BRR} \le 65535$. If BRR = 0, then $\text{SCI Asynchronous Baud} = \frac{\text{LSPCLK}}{16}$ Where: BRR = the 16-bit value (in decimal) in the baud-select registers.

2.5 SCI Control Register 2 (SCICTL2)

SCICTL2 enables the receive-ready, break-detect, and transmit-ready interrupts as well as transmitter-ready and -empty flags.

Figure 2–5. SCI Control Register 2 (SCICTL2) — Address 7054h

7	6	5 2	1	0
TXRDY	TX EMPTY	Reserved	RX/BK INT ENA	TX INT ENA
R-1	R-1	R-0	R/W-0	R/W-0

Legend: R = Read access, W = Write access, -n = value after reset

Bit(s)	Name	Description
7	TXRDY	Transmitter buffer register ready flag. When set, this bit indicates that the transmit data buffer register, SCITXBUF, is ready to receive another character. Writing data to the SCITXBUF automatically clears this bit. When set, this flag asserts a transmitter interrupt request if the interrupt-enable bit, TX INT ENA (SCICTL2.0), is also set. TXRDY is set to 1 by enabling the SW RESET bit (SCICTL.2) or by a system reset. 1 SCITXBUF is ready to receive the next character 0 SCITXBUF is full
6	TX EMPTY	Transmitter empty flag. This flag's value indicates the contents of the transmitter's buffer register (SCITXBUF) and shift register (TXSHF). An active SW RESET (SCICTL1.2), or a system reset, sets this bit. This bit *does not* cause an interrupt request. 1 Transmitter buffer and shift registers are both empty 0 Transmitter buffer or shift register or both are loaded with datal
5–2	Reserved	
1	RX/BK INT ENA	Receiver-buffer/break interrupt enable. This bit controls the interrupt request caused by *either* the RXRDY flag *or* the BRKDT flag (bits SCIRXST.6 and .5) being set. However, RX/BK INT ENA does not prevent the setting of these flags. 1 Enable RXRDY/BRKDT interrupt 0 Disable RXRDY/BRKDT interrupt
0	TX INT ENA	SCITXBUF-register interrupt enable.This bit controls the interrupt request caused by setting the TXRDY flag bit (SCICTL2.7). However, it does not prevent the TXRDY flag from being set (being set indicates that register SCITXBUF is ready to receive another character). 1 Enable TXRDY interrupt 0 Disable TXRDY interrupt

2.6 SCI Receiver Status Register (SCIRXST)

SCIRXST contains seven bits that are receiver status flags (two of which can generate interrupt requests). Each time a complete character is transferred to the receiver buffers (SCIRXEMU and SCIRXBUF), the status flags are updated. Figure 2–7 on page 2-12 shows the relationships between several of the register's bits.

Figure 2–6. SCI Receiver Status Register (SCIRXST) — Address 7055h

7	6	5	4	3	2	1	0
RX ERROR	RXRDY	BRKDT	FE	OE	PE	RXWAKE	Reserved
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

Legend: R = Read access, W = Write access, -n = value after reset

Bit(s)	Name	Description
7	RX ERROR	SCI receiver error flag. The RX ERROR flag indicates that one of the error flags in the receiver status register is set. RX ERROR is a logical OR of the break detect, framing error, overrun, and parity error enable flags (bits 5–2: BRKDT, FE, OE, and PE). A 1 on this bit will cause an interrupt if the RX ERR INT ENA bit (SCICTL1.6) is set. This bit can be used for fast error-condition checking during the interrupt service routine. This error flag cannot be cleared directly; it is cleared by an active SW RESET or by a system reset. 1 Error flag(s) set 0 No error flags set
6	RXRDY	SCI receiver-ready flag. When a new character is ready to be read from the SCIRXBUF register, the receiver sets this bit, and a receiver interrupt is generated if the RX/BK INT ENA bit (SCICTL2.1) is a 1. RXRDY is cleared by a reading of the SCIRXBUF register, by an active SW RESET, or by a system reset. 1 Character ready to be read from SCIRXBUF 0 No new character in SCIRXBUF

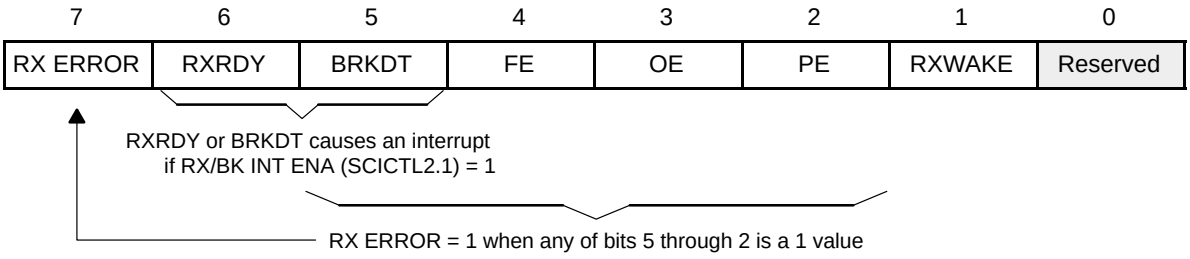
Figure 2–6.SCI Receiver Status Register (SCIRXST) — Address 7055h (Continued)

Bit(s)	Name	Description
5	BRKDT	SCI break-detect flag. The SCI sets this bit when a break condition occurs. A break condition occurs when the SCI receiver data line (SCIRXD) remains continuously low for at least ten bits, beginning after a missing first stop bit. The occurrence of a break causes a receiver interrupt to be generated if the RX/BK INT ENA bit is a 1, but it does not cause the receiver buffer to be loaded. A BRKDT interrupt can occur even if the receiver SLEEP bit is set to 1. BRKDT is cleared by an active SW RESET or by a system reset. It is not cleared by receipt of a character after the break is detected. In order to receive more characters, the SCI must be reset by toggling the SW RESET bit or by a system reset. 1 Break condition occurred 0 No break condition
4	FE	SCI framing-error flag. The SCI sets this bit 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. The FE bit is reset by a clearing of the SW RESET bit or by a system reset. 1 Framing error detected 0 No framing error detected
3	OE	SCI overrun-error flag. The SCI sets this bit when a character is transferred into registers SCIRXEMU and SCIRXBUF before the previous character is fully read by the CPU or DMAC. The previous character is overwritten and lost. The OE flag bit is reset by an active SW RESET or by a system reset. 1 Overrun error detected 0 No overrun error detected
2	PE	SCI parity-error flag. This flag bit is set when a character is received with a mismatch between the number of 1s and its parity bit. The address bit is included in the calculation. If parity generation and detection is not enabled, the PE flag is disabled and read as 0. The PE bit is reset by an active SW RESET or a system reset. 1 Parity error is detected 0 No parity error **or** parity is disabled

Figure 2–6.SCI Receiver Status Register (SCIRXST) — Address 7055h (Continued)

Bit(s)	Name	Description
1	RXWAKE	Receiver wake-up-detect flag. A value of 1 in this bit indicates detection of a receiver wake-up condition. In the address-bit multiprocessor mode (SCICCR.3 = 1), RXWAKE reflects the value of the address bit for the character contained in SCIRXBUF. In the idle-line multiprocessor mode, RXWAKE is set if the SCIRXD data line is detected as idle. RXWAKE is a read-only flag, cleared by one of the following: ❑ The transfer of the first byte after the address byte to SCIRXBUF ❑ The reading of SCIRXBUF ❑ An active SW RESET ❑ A system reset
0	Reserved	Reads return zero; writes have no effect.

Figure 2–7. Register SCIRXST Bit Associations — Address 7055h



2.7 Receiver Data Buffer Registers (SCIRXEMU, SCIRXBUF)

Received data is transferred from RXSHF to SCIRXEMU and SCIRXBUF. When the transfer is complete, the RXRDY flag (bit SCIRXST.6) is set, indicating that the received data is ready to be read. Both registers contain the same data; they have separate addresses but are not physically separate buffers. The only difference is that reading SCIRXEMU *does not* clear the RXRDY flag; however, reading SCIRXBUF clears the flag.

2.7.1 Emulation Data Buffer

Normal SCI data-receive operations read the data received from the SCIRXBUF register. The SCIRXEMU register is used principally by the emulator (EMU) because it can continuously read the data received for screen updates without clearing the RXRDY flag. SCIRXEMU is cleared by a system reset.

This is the register that should be used in an emulator watch window to view the contents of the SCIRXBUF register.

SCIRXEMU is not physically implemented; it is just a different address location to access the SCIRXBUF register without clearing the RXRDY flag.

Figure 2–8. Emulation Data Buffer Register (SCIRXEMU) — Address 7056h

7	6	5	4	3	2	1	0
ERXDT7	ERXDT6	ERXDT5	ERXDT4	ERXDT3	ERXDT2	ERXDT1	ERXDT0
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

Legend: R = Read access, -0 = value after reset

2.7.2 Receiver Data Buffer (SCIRXBUF)

When the current data received is shifted from RXSHF to the receiver buffer, flag bit RXRDY is set and the data is ready to be read. If the RX/BK INT ENA bit (SCICTL2.1) is set, this shift also causes an interrupt. When SCIRXBUF is read, the RXRDY flag is reset. SCIRXBUF is cleared by a system reset.

Figure 2–9. SCI Receive Data Buffer Register (SCIRXBUF) — Address 7057h

15	14	13	8
SCIFFFE	SCIFFPE	Reserved	
R–0	R–0	R–0	

7	6	5	4	3	2	1	0
RXDT7	RXDT6	RXDT5	RXDT4	RXDT3	RXDT2	RXDT1	RXDT0
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

Legend: R = Read access, W = Write access, -n = value after reset

Note: Shaded area is applicable only if the FIFO is enabled.

Bit(s)	Name	Description
15	SCIFFFE	SCIFFFE. SCI FIFO Framing error flag bit
		1 A frame error occurred while receiving the character in bits 7–0. This bit is associated with the character on the top of the FIFO.
		0 No frame error occurred while receiving the character, in bits 7–0. This bit is associated with the character on the top of the FIFO.
14	SCIFFPE	SCIFFPE. SCI FIFO parity error flag bit
		1 A parity error occurred while receiving the character in bits 7–0. This bit is associated with the character on the top of the FIFO.
		0 No parity error occurred while receiving the character, in bits 7–0. This bit is associated with the character on the top of the FIFO.
13:8	Reserved	
7:0	RXDT7–0	Receive Character bits

2.8 SCI Transmit Data Buffer Register (SCITXBUF)

Data bits to be transmitted are written to SCITXBUF. These bits must be right-justified because the leftmost bits are ignored for characters less than eight bits long. The transfer of data from this register to the TXSHF transmitter shift register sets the TXRDY flag (SCICTL2.7), indicating that SCITXBUF is ready to receive another set of data. If bit TX INT ENA (SCICTL2.0) is set, this data transfer also causes an interrupt.

Figure 2–10. Transmit Data Buffer Register (SCITXBUF) — Address 7059h

7	6	5	4	3	2	1	0
TXDT7	TXDT6	TXDT5	TXDT4	TXDT3	TXDT2	TXDT1	TXDT0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

Legend: R = Read access, W = Write access, -0 = value after reset

2.9 SCI FIFO Registers

Figure 2–11. SCI FIFO Transmit (SCIFFTX) Register — Address 705Ah

15	14	13	12	11	10	9	8
SCIRST	SCIFFENA	TXFIFO Reset	TXFFST4	TXFFST3	TXFFST2	TXFFST1	TXFFST0
R/W-1	R/W–0	R/W–1	R–0	R–0	R–0	R–0	R–0

7	6	5	4	3	2	1	0
TXFFINT Flag	TXFFINT CLR	TXFFIENA	TXFFIL4	TXFFIL3	TXFFIL2	TXFFIL1	TXFFIL0
R–0	W–0	R/W–0	R/W–0	R/W–0	R/W–0	R/W–0	R/W–0

Legend: R = Read access, W = Write access, -0 = value after reset

Bit(s)	Name	Description	
15	SCIRST	0	Write 0 to reset the SCI transmit and receive channels. SCI FIFO register configuration bits will be left as is.
		1	SCI FIFO can resume transmit or receive. SCIRST should be 1 even for Autobaud logic to work.
14	SCIFFENA	0	SCI FIFO enhancements are disabled and FIFO is in reset.
		1	SCI FIFO enhancements are enabled.
13	TXFIFO Reset	Transmit FIFO reset	
		0	Reset the FIFO pointer to zero and hold in reset.
		1	Re-enable transmit FIFO operation.
8–12	TXFFST4–0	00000	Transmit FIFO is empty.
		00001	Transmit FIFO has 1 words
		00010	Transmit FIFO has 2 words
		00011	Transmit FIFO has 3 words
		0xxxx	Transmit FIFO has x words
		10000	Transmit FIFO has 16 words
7	TXFFINT	Transmit FIFO interrupt	
		0	TXFIFO interrupt has not occurred, read-only bit
		1	TXFIFO interrupt has occurred, read-only bit

Figure 2–11. SCI FIFO Transmit (SCIFFTX) Register — Address 705Ah (Continued)

Bit(s)	Name	Description	
6	TXFFINT CLR	0	Write 0 has no effect on TXFIFINT flag bit, Bit reads back a zero
		1	Write 1 to clear TXFFINT flag in bit 7
5	TXFFIENA	0	TX FIFO interrupt based on TXFFIVL match (less than or equal to) is disabled
		1	TX FIFO interrupt based on TXFFIVL match (less than or equal to) is enabled.
0–4	TXFFIL4–0	TXFFIL4–0 Transmit FIFO interrupt level bits. Transmit FIFO will generate interrupt when the FIFO status bits (TXFFST4–0) and FIFO level bits (TXFFIL4–0) match (less than or equal to).	
		Default value should be 0x00000.	

Figure 2–12. SCI FIFO Receive (SCIFFRX) Register — Address 705Bh

15	14	13	12	11	10	9	8
RXFFOVF	RXFFOVR CLR	RXFIFO Reset	RXFIFST4	RXFIFST3	RXFIFST2	RXFIFST1	RXFIFST0
R-0	W–0	R/W–1	R–0	R–0	R–0	R–0	R–0

7	6	5	4	3	2	1	0
RXFFINT Flag	RXFFINT CLR	RXFFIENA	RXFFIL4	RXFFIL3	RXFFIL2	RXFFIL1	RXFFIL0
R-0	W–0	R/W–0	R/W–1	R/W–1	R/W–1	R/W–1	R/W–1

Note: R = Read access, W = Write access, -0 = value after reset

Bit(s)	Name	Description	
15	RXFFOVF	Receive FIFO overflow. This will function as flag, but cannot generate interrupt by itself. This condition will occur while receive interrupt is active. Receive interrupts should service this flag condition.	
		0	Receive FIFO has not overflowed, read-only bit
		1	Receive FIFO has overflowed, read-only bit. More than 16 words have been received in to the FIFO, and the first received word is lost
14	RXFFOVF CLR	0	Write 0 has no effect on RXFFOVF flag bit, Bit reads back a zero
		1	Write 1 to clear RXFFOVF flag in bit 15
13	RXFIFO Reset	Receive FIFO reset	
		0	Write 0 to reset the FIFO pointer to zero, and hold in reset.
		1	Re-enable receive FIFO operation
8–12	RXFFST4–0	00000	Receive FIFO is empty
		00001	Receive FIFO has 1 word
		00010	Receive FIFO has 2 words
		00011	Receive FIFO has 3 words
		0xxxx	Receive FIFO has x words
		10000	Receive FIFO has 16 words

Figure 2−12. SCI FIFO Receive (SCIFFRX) Register — Address 705Bh (Continued)

Bit(s)	Name	Description
7	RXFFINT	Receive FIFO interrupt 0 RXFIFO interrupt has not occurred, read-only bit 1 RXFIFO interrupt has occurred, read-only bit
6	RXFFINT CLR	Receive FIFO interrupt clear 0 Write 0 has no effect on RXFIFINT flag bit. Bit reads back a zero. 1 Write 1 to clear RXFFINT flag in bit 7
5	RXFFIENA	Receive FIFO interrupt enable 0 RX FIFO interrupt based on RXFFIVL match (less than or equal to) will be disabled 1 RX FIFO interrupt based on RXFFIVL match (less than or equal to) will be enabled.
0−4	RXFFIL4−0	Receive FIFO interrupt level bits Receive FIFO generates interrupt when the FIFO status bits (RXFFST4−0) and FIFO level bits (RXFFIL4−0) match (i.e., are greater than or equal to). Default value of these bits after reset − 11111. This will avoid frequent interrupts, after reset, as the receive FIFO will be empty most of the time.

Figure 2–13. SCI FIFO Control (SCIFFCT) Register — Address 705Ch

15	14	13	12	8
ABD	ABD CLR	CDC	Reserved	
R-0	W–0	R/W–0	R–0	

7	6	5	4	3	2	1	0
FFTXDLY7	FFTXDLY6	FFTXDLY5	FFTXDLY4	FFTXDLY3	FFTXDLY2	FFTXDLY1	FFTXDLY0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

Note: R = Read access, W = Write access, -0 = value after reset

Bit(s)	Name	Description
15	ABD	Auto-baud detect (ABD) bit. 0 Auto-baud detection is not complete. ”A”,”a” character has not been received successfully. 1 Auto-baud hardware has detected ”A” or ”a” character on the SCI receive register. Auto-detect is complete. This will work only if CDC bit is set to enable auto-baud.
14	ABD CLR	ABD-clear bit 0 Write 0 has no effect on ABD flag bit. Bit reads back a zero. 1 Write 1 to clear ABD flag in bit 15.
13	CDC	CDC calibrate A-detect bit 0 Disables auto-baud alignment 1 Enables auto-baud alignment
8–12	Reserved	Reserved
0–7	FFTXDLY7–0	FIFO transfer delay. These bits define the delay between every transfer from FIFO transmit buffer to transmit shift register. The delay is defined in the number of SCI serial baud clock cycles. The 8 bit register could define a minimum delay of 0 baud clock cycles and a maximum of 256 baud clock cycles In FIFO mode, the buffer (TXBUF) between the shift register and the FIFO should be filled only after the shift register has completed shifting of the last bit. This is required to pass on the delay between transfers to the data stream. In FIFO mode, TXBUF should not be treated as one additional level of buffer. The delayed transmit feature will help to create an auto-flow scheme without RTS/CTS controls as in standard UARTS.

2.10 Priority Control Register (SCIPRI)

Figure 2–14. SCI Priority Control Register (SCIPRI) — Address 705Fh

7	5	4	3	2	0
Reserved		SCI SOFT	SCI FREE	Reserved	
R-0		R/W-0	R/W-0	R-0	

Note: R = Read access, W = Write access, -0 = value after reset

Bit(s)	Name	Description
7–5	Reserved	Reads return zero; writes have no effect.
4:3	SOFT and FREE	These bits determine what occurs when an emulation suspend event occurs (for example, when the debugger hits a breakpoint). The peripheral can continue whatever it is doing (free-run mode), or if in stop mode, it can either stop immediately or stop when the current operation (the current receive/transmit sequence) is complete.

Bit 4 SOFT	Bit 3 FREE	
0	0	Immediate stop on suspend
1	0	Complete current receive/transmit sequence before stopping
x	1	Free run. Continues SCI operation regardless of suspend

2:0	Reserved	Reads return zero; writes have no effect.

Appendix A

SCI Register Summary

Figure A–1. SCI Communication Control Register (SCICCR) — Address 7050h

7	6	5	4	3	2	1	0
STOP BITS	EVEN/ODD PARITY	PARITY ENABLE	LOOPBACK ENA	ADDR/IDLE MODE	SCICHAR2	SCICHAR1	SCICHAR0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

Figure A–2. SCI Control Register 1 (SCICTL1) — Address 7051h

7	6	5	4	3	2	1	0
Reserved	RX ERR INT ENA	SW RESET	Reserved	TXWAKE	SLEEP	TXENA	RXENA
R-0	R/W-0	R/W-0	R-0	R/S-0	R/W-0	R/W-0	R/W-0

Figure A–3. Baud-Select MSbyte Register (SCIHBAUD) — Address 7052h

15	14	13	12	11	10	9	8
BAUD15 (MSB)	BAUD14	BAUD13	BAUD12	BAUD11	BAUD10	BAUD9	BAUD8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

Figure A–4. Baud-Select LSbyte Register (SCILBAUD) — Address 7053h

7	6	5	4	3	2	1	0
BAUD7	BAUD6	BAUD5	BAUD4	BAUD3	BAUD2	BAUD1	BAUD0 (LSB)
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

Figure A–5. SCI Control Register 2 (SCICTL2) — Address 7054h

7	6	5	2	1	0
TXRDY	TX EMPTY	Reserved		RX/BK INT ENA	TX INT ENA
R-1	R-1	R-0		R/W-0	R/W-0

Figure A–6. SCI Receiver Status Register (SCIRXST) — Address 7055h

7	6	5	4	3	2	1	0
RX ERROR	RXRDY	BRKDT	FE	OE	PE	RXWAKE	Reserved
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

Figure A–7. Emulation Data Buffer Register (SCIRXEMU)— Address 7056h

7	6	5	4	3	2	1	0
ERXDT7	ERXDT6	ERXDT5	ERXDT4	ERXDT3	ERXDT2	ERXDT1	ERXDT0
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

Figure A–8. SCI Receive Data Buffer Register (SCIRXBUF) — Address 7057h

15	14	13	8
SCIFFFE	SCIFFPE	Reserved	
R–0	R–0	R–0	

7	6	5	4	3	2	1	0
RXDT7	RXDT6	RXDT5	RXDT4	RXDT3	RXDT2	RXDT1	RXDT0
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

Figure A–9. Transmit Data Buffer Register (SCITXBUF) — Address 7059h

7	6	5	4	3	2	1	0
TXDT7	TXDT6	TXDT5	TXDT4	TXDT3	TXDT2	TXDT1	TXDT0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

Figure A–10. SCI FIFO Transmit (SCIFFTX) Register — Address 705Ah

15	14	13	12	11	10	9	8
SCIRST	SCIFFENA	TXFIFO Reset	TXFFST4	TXFFST3	TXFFST2	TXFFST1	TXFFST0
R/W-1	R/W–0	R/W–1	R–0	R–0	R–0	R–0	R–0

7	6	5	4	3	2	1	0
TXFFINT Flag	TXFFINT CLR	TXFFIENA	TXFFIL4	TXFFIL3	TXFFIL2	TXFFIL1	TXFFIL0
R–0	W–0	R/W–0	R/W–0	R/W–0	R/W–0	R/W–0	R/W–0

Figure A–11. SCI FIFO Receive (SCIFFRX) Register — Address 705Bh

15	14	13	12	11	10	9	8
RXFFOVF	RXFFOVR CLR	RXFIFO Reset	RXFIFST4	RXFIFST3	RXFIFST2	RXFIFST1	RXFIFST0
R-0	W–0	R/W–1	R–0	R–0	R–0	R–0	R–0

7	6	5	4	3	2	1	0
RXFFINT Flag	RXFFINT CLR	RXFFIENA	RXFFIL4	RXFFIL3	RXFFIL2	RXFFIL1	RXFFIL0
R-0	W–0	R/W–0	R/W–1	R/W–1	R/W–1	R/W–1	R/W–1

Figure A–12. SCI FIFO Control (SCIFFCT) Register — Address 705Ch

15	14	13	12				8
ABD	ABD CLR	CDC	Reserved				
R-0	W–0	R/W–0	R–0				

7	6	5	4	3	2	1	0
FFTXDLY7	FFTXDLY6	FFTXDLY5	FFTXDLY4	FFTXDLY3	FFTXDLY2	FFTXDLY1	FFTXDLY0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

Figure A–13. SCI Priority Control Register (SCIPRI) — Address 705Fh

7	5	4	3	2	0
Reserved		SCI SOFT	SCI FREE	Reserved	
R-0		R/W-0	R/W-0	R-0	