

Section 14 Serial Communication Interface

14.1 Overview

The H8/532 chip includes a single-channel serial communication interface (SCI) for transferring serial data to and from other chips. The SCI supports both synchronous and asynchronous data transfer. Communication control functions are provided by eight internal registers.

14.1.1 Features

The features of the on-chip serial communication interface are:

- Selection of asynchronous or synchronous mode
 - Asynchronous mode

The SCI can communicate with a UART (Universal Asynchronous Receiver/Transmitter), ACIA (Asynchronous Communication Interface Adapter), or other chip that employs standard asynchronous serial communication. Eight data formats are available.

 - Data length: 7 or 8 bits
 - Stop bit length: 1 or 2 bits
 - Parity: Even, odd, or none
 - Error detection: Parity, overrun, and framing errors
 - Synchronous mode

The SCI can communicate with chips able to synchronize data transfers with clock pulses.

 - Data length: 8 bits
 - Error detection: Overrun errors
- Full duplex communication

The transmitting and receiving sections are independent, so the SCI can transmit and receive simultaneously. Both the transmit and receive sections use double buffering, so continuous data transfer is possible in either direction.
- Built-in baud rate generator

Any specified bit rate can be generated.
- Internal or external clock source

The baud rate generator can operate on an internal clock source, or an external clock signal input at the SCK pin.
- Three interrupts

Transmit-end, receive-end, and receive-error interrupts are requested independently. The transmit-end and receive-end interrupts can be served by the on-chip data transfer controller (DTC), providing a convenient way to transfer data with minimal CPU programming.

14.1.2 Block Diagram

Figure 14-1 shows a block diagram of serial communication interface.

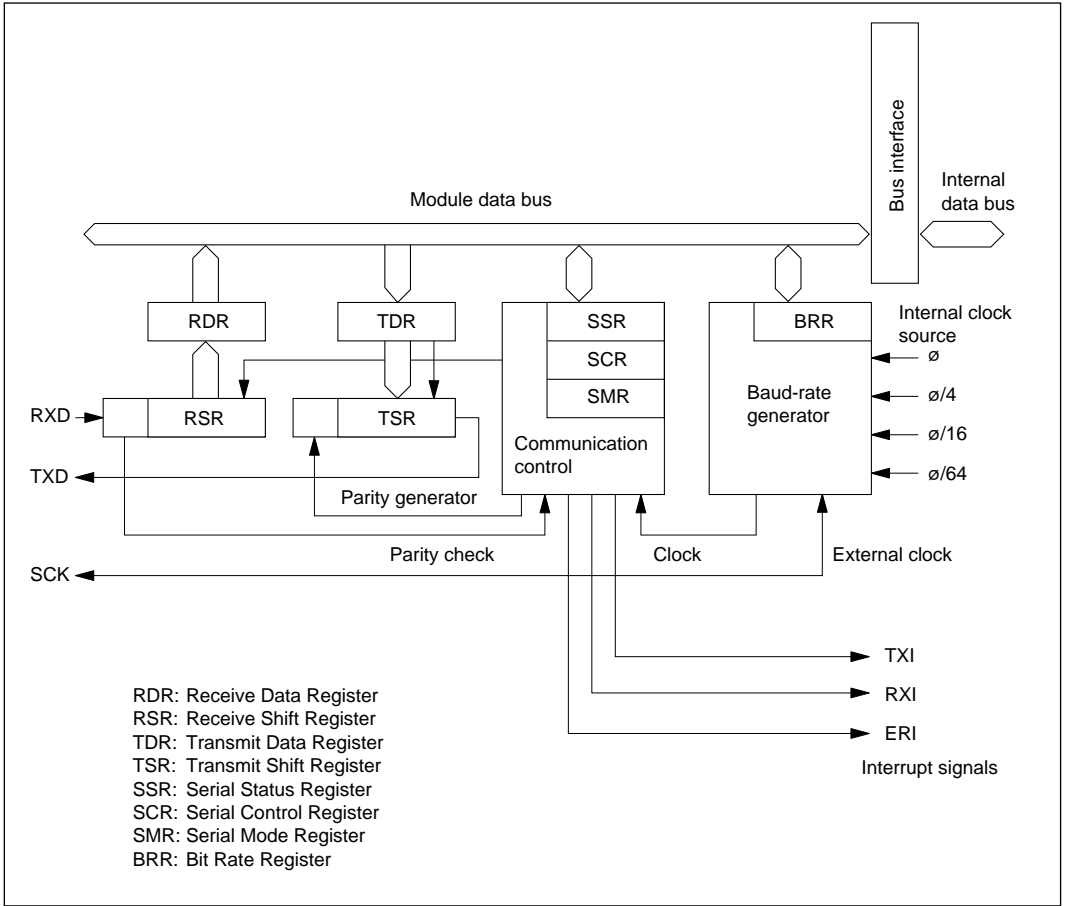


Figure 14-1 Block Diagram of Serial Communication Interface

14.1.3 Input and Output Pins

Table 14-1 lists the input and output pins used by the SCI module.

Table 14-1 SCI Input/Output Pins

Name	Abbreviation	I/O	Function
Serial clock	SCK	Input/output	Serial clock input and output.
Receive data	RXD	Input	Receive data input.
Transmit data	TXD	Output	Transmit data output.

14.1.4 Register Configuration

Table 14-2 lists the SCI registers.

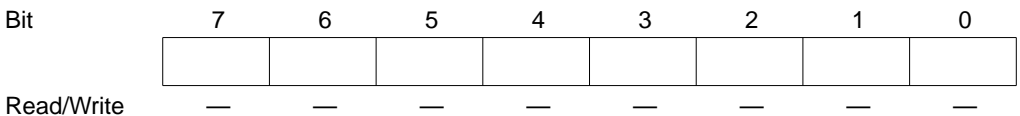
Table 14-2 SCI Registers

Name	Abbreviation	R/W	Initial Value	Address
Receive shift register	RSR	—	—	—
Receive data register	RDR	R	H'00	H'FFDD
Transmit shift register	TSR	—	—	—
Transmit data register	TDR	R/W	H'FF	H'FFDB
Serial mode register	SMR	R/W	H'04	H'FFD8
Serial control register	SCR	R/W	H'0C	H'FFDA
Serial status register	SSR	R/(W)*	H'87	H'FFDC
Bit rate register	BRR	R/W	H'FF	H'FFD9

* Software can write a "0" to clear the status flag bits, but cannot write a "1."

14.2 Register Descriptions

14.2.1 Receive Shift Register (RSR)



The RSR receives incoming data bits. When one data character has been received, it is transferred to the receive data register (RDR).

The CPU cannot read or write the RSR directly.

14.2.2 Receive Data Register (RDR)—H'FFDD

Bit	7	6	5	4	3	2	1	0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R	R

The RDR stores received data. As each character is received, it is transferred from the RSR to the RDR, enabling the RSR to receive the next character. This double-buffering allows the SCI to receive data continuously.

The CPU can read but not write the RDR. The RDR is initialized to H'00 at a reset and in the standby modes.

14.2.3 Transmit Shift Register (TSR)

Bit	7	6	5	4	3	2	1	0
Read/Write	—	—	—	—	—	—	—	—

The TSR holds the character currently being transmitted. When transmission of this character is completed, the next character is moved from the transmit data register (TDR) to the TSR and transmission of that character begins. If the TDR does not contain valid data, the SCI stops transmitting.

The CPU cannot read or write the TSR directly.

14.2.4 Transmit Data Register (TDR)—H'FFDB

Bit	7	6	5	4	3	2	1	0
Initial value	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The TDR is an 8-bit readable/writable register that holds the next character to be transmitted. When the TSR becomes empty, the character written in the TDR is transferred to the TSR.

Continuous data transmission is possible by writing the next byte in the TDR while the current byte is being transmitted from the TSR.

The TDR is initialized to H'FF at a reset and in the standby modes.

14.2.5 Serial Mode Register (SMR)—H'FFD8

Bit	7	6	5	4	3	2	1	0
	C/ \bar{A}	CHR	PE	O/ \bar{E}	STOP	—	CKS1	CKS0
Initial value	0	0	0	0	0	1	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	—	R/W	R/W

The SMR is an 8-bit readable/writable register that controls the communication format and selects the clock rate for the internal clock source. It is initialized to H'04 at a reset and in the standby modes.

Bit 7—Communication Mode (C/ \bar{A}): This bit selects the asynchronous or synchronous communication mode.

Bit 7

C/ \bar{A}	Description
0	Asynchronous communication. (Initial value)
1	Communication is synchronized with the serial clock.

Bit 6—Character Length (CHR): This bit selects the character length in asynchronous mode. It is ignored in synchronous mode.

Bit 6

CHR	Description
0	8 Bits per character. (Initial value)
1	7 Bits per character.

Bit 5—Parity Enable (PE): This bit selects whether to add a parity bit in asynchronous mode. It is ignored in synchronous mode.

Bit 5

PE	Description
0	Transmit: No parity bit is added. (Initial value) Receive: Parity is not checked.
1	Transmit: A parity bit is added. Receive: Parity is not checked.

Bit 4—Parity Mode (O/E): In asynchronous mode, when parity is enabled (PE = 1), this bit selects even or odd parity.

Even parity means that a parity bit is added to the data bits for each character to make the total number of 1's even. Odd parity means that the total number of 1's is made odd.

This bit is ignored when PE = 0 and in the synchronous mode.

Bit 4

O/E	Description
0	Even parity. (Initial value)
1	Odd parity.

Bit 3—Stop Bit Length (STOP): This bit selects the number of stop bits. It is ignored in the synchronous mode.

Bit 3

STOP	Description
0	1 Stop bit. (Initial value)
1	2 Stop bits.

Bit 2—Reserved: This bit cannot be modified and is always read as 1.

Bits 1 and 0—Clock Select 1 and 0 (CKS1 and CKS0): These bits select the internal clock source when the baud rate generator is clocked from within the H8/532 chip.

Bit 1 CKS1	Bit 0 CKS0	Description
0	0	∅ clock (Initial value)
0	1	∅/4 clock
1	0	∅/16 clock
1	1	∅/64 clock

14.2.6 Serial Control Register (SCR)—H'FFDA

Bit	7	6	5	4	3	2	1	0
	TIE	RIE	TE	RE	—	—	CKE1	CKE0
Initial value	0	0	0	0	1	1	0	0
Read/Write	R/W	R/W	R/W	R/W	—	—	R/W	R/W

The SCR is an 8-bit readable/writable register that enables or disables various SCI functions. It is initialized to H'0C at a reset and in the standby modes.

Bit 7—Transmit Interrupt Enable (TIE): This bit enables or disables the transmit-end interrupt (TXI) requested when the transmit data register empty (TDRE) bit in the serial status register (SSR) is set to 1.

Bit 7

TIE	Description
0	The transmit-end interrupt request (TXI) is disabled. (Initial value)
1	The transmit-end interrupt request (TXI) is enabled.

Bit 6—Receive Interrupt Enable (RIE): This bit enables or disables the receive-end interrupt (RXI) requested when the receive data register full (RDRF) bit in the serial status register (SSR) is set to 1. It also enables and disables the receive-error interrupt (ERI) request.

Bit 6

RIE	Description
0	The receive-end interrupt (RXI) and receive-error interrupt (ERI) requests are disabled. (Initial value)
1	The receive-end interrupt (RXI) and receive-error interrupt (ERI) requests are enabled.

Bit 5—Transmit Enable (TE): This bit enables or disables the transmit function. When the transmit function is enabled, the TXD pin is automatically used for output. When the transmit function is disabled, the TXD pin can be used as a general-purpose I/O port.

Bit 5

TE	Description
0	The transmit function is disabled. The TXD pin can be used as a general-purpose I/O port. (Initial value)
1	The transmit function is enabled. The TXD pin is used for output.

Bit 4—Receive Enable (RE): This bit enables or disables the receive function. When the receive function is enabled, the RXD pin is automatically used for input. When the receive function is disabled, the RXD pin is available as a general-purpose I/O port.

Bit 4

RE	Description
0	The receive function is disabled. The RXD pin can be used as a general-purpose I/O port. (Initial value)
1	The receive function is enabled. The RXD pin is used for input.

Bits 3 and 2—Reserved: These bits cannot be modified and are always read as 1.

Bit 1—Clock Enable 1 (CKE1): This bit selects the internal or external clock source for the baud rate generator. When the external clock source is selected, the SCK pin is automatically used for input of the external clock signal.

Bit 1

CKE1	Description
0	Internal clock source. (Initial value)
1	External clock source. (The SCK pin is used for input.)

Bit 0—Clock Enable 0 (CKE0): When an internal clock source is used in synchronous mode, this bit enables or disables serial clock output at the SCK pin.

This bit is ignored when the external clock is selected, or when the asynchronous mode is selected.

For further information on the communication format and clock source selection, see tables 14-5 and 14-6 in section 14.3, “Operation.”

Bit 0

CKE0	Description
0	The SCK pin is not used by the SCI (and is available as a general-purpose I/O port). (Initial value)
1	The SCK pin is used for serial clock output.

14.2.7 Serial Status Register (SSR)—H'FFDC

Bit	7	6	5	4	3	2	1	0
	TDRE	RDRF	ORER	FER	PER	—	—	—
Initial value	1	0	0	0	0	1	1	1
Read/Write	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	—	—	—

* Software can write a 0 to clear the flags, but cannot write a 1 in these bits.

The SSR is an 8-bit register that indicates transmit and receive status. It is initialized to H'87 at a reset and in the standby modes.

Bit 7—Transmit Data Register Empty (TDRE): This bit indicates when the TDR contents have been transferred to the TSR and the next character can safely be written in the TDR.

Bit 7

TDRE Description

0	This bit is cleared from 1 to 0 when: <ol style="list-style-type: none"> 1. The CPU reads the TDRE bit, then writes a 0 in this bit. 2. The data transfer controller (DTC) writes data in the TDR. 	
1	This bit is set to 1 at the following times: <ol style="list-style-type: none"> 1. The chip is reset or enters a standby mode. 2. When TDR contents are transferred to the TSR. 3. When TDRE = 0 and the TE bit is cleared to 0. 	(Initial value)

Bit 6—Receive Data Register Full (RDRF): This bit indicates when one character has been received and transferred to the RDR.

Bit 6

RDRF Description

0	This bit is cleared from 1 to 0 when: <ol style="list-style-type: none"> 1. The CPU reads the RDRF bit, then writes a 0 in this bit. 2. The data transfer controller (DTC) reads the RDR. 3. The chip is reset or enters a standby mode. 	(Initial value)
1	This bit is set to 1 when one character is received without error and transferred from the RSR to the RDR.	

Bit 5—Overrun Error (ORER): This bit indicates an overrun error during reception.

Bit 5

ORER	Description
0	This bit is cleared from 1 to 0 when: (Initial value) 1. The CPU reads the ORER bit, then writes a 0 in this bit. 2. The chip is reset or enters a standby mode.
1	This bit is set to 1 if reception of the next character ends while the receive data register is still full (RDRF = 1).

Bit 4—Framing Error (FER): This bit indicates a framing error during data reception in the synchronous mode. It has no meaning in the asynchronous mode.

Bit 4

FER	Description
0	This bit is cleared to from 1 to 0 when: (Initial value) 1. The CPU reads the FER bit, then writes a 0 in this bit. 2. The chip is reset or enters a standby mode.
1	This bit is set to 1 if a framing error occurs (stop bit = 0).

Bit 3—Parity Error (PER): This bit indicates a parity error during data reception in the asynchronous mode, when a communication format with parity bits is used.

This bit has no meaning in the synchronous mode, or when a communication format without parity bits is used.

Bit 3

PER	Description
0	This bit is cleared from 1 to 0 when: (Initial value) 1. The CPU reads the PER bit, then writes a 0 in this bit. 2. The chip is reset or enters a standby mode.
1	This bit is set to 1 when a parity error occurs (the parity of the received data does not match the parity selected by the bit in the SMR).

Bits 2 to 0—Reserved: These bits cannot be modified and are always read as 1.

14.2.8 Bit Rate Register (BRR)—H'FFD9

Bit	7	6	5	4	3	2	1	0
Initial value	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The BRR is an 8-bit register that, together with the CKS1 and CKS0 bits in the SMR, determines the bit rate output by the baud rate generator.

The BRR is initialized to H'FF (the slowest rate) at a reset and in the standby modes.

Tables 14-3 and 14-4 show examples of BRR (N) and CKS (n) settings for commonly used bit rates.

Table 14-3 Examples of BRR Settings in Asynchronous Mode (1)

Bit Rate	XTAL Frequency (MHz)											
	2			2.4576			4			4.194304		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	1	70	+0.03	1	86	+0.31	1	141	+0.03	1	148	-0.04
150	0	207	+0.16	0	255	0	1	103	+0.16	1	108	+0.21
300	0	103	+0.16	0	127	0	0	207	+0.16	0	217	+0.21
600	0	51	+0.16	0	63	0	0	103	+0.16	0	108	+0.21
1200	0	25	+0.16	0	31	0	0	51	+0.16	0	54	-0.70
2400	0	12	+0.16	0	15	0	0	25	+0.16	0	26	+1.14
4800	—	—	—	0	7	0	0	12	+0.16	0	13	-2.48
9600	—	—	—	0	3	0	—	—	—	—	—	—
19200	—	—	—	0	1	0	—	—	—	—	—	—
31250	—	—	—	—	—	—	0	1	0	—	—	—
38400	—	—	—	0	0	0	—	—	—	—	—	—

Table 14-3 Examples of BRR Settings in Asynchronous Mode (2)

Bit Rate	XTAL Frequency (MHz)											
	4.9152			6			7.3728			8		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	1	174	-0.26	2	52	+0.50	2	64	+0.70	2	70	+0.03
150	1	127	0	1	155	+0.16	1	191	0	1	207	+0.16
300	0	255	0	1	77	+0.16	1	95	0	1	103	+0.16
600	0	127	0	0	155	+0.16	0	191	0	0	207	+0.16
1200	0	63	0	0	77	+0.16	0	95	0	0	103	+0.16
2400	0	31	0	0	38	+0.16	0	47	0	0	51	+0.16
4800	0	15	0	0	19	-2.34	0	23	0	0	25	+0.16
9600	0	7	0	—	—	—	0	11	0	0	12	+0.16
19200	0	3	0	—	—	—	0	5	0	—	—	—
31250	—	—	—	0	2	0	—	—	—	0	3	0
38400	0	1	0	—	—	—	0	2	0	—	—	—

Table 14-3 Examples of BRR Settings in Asynchronous Mode (3)

Bit Rate	XTAL Frequency (MHz)											
	9.8304			10			12			12.288		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	86	+0.31	2	88	-0.25	2	106	-0.44	2	108	+0.08
150	1	255	0	2	64	+0.16	2	77	0	2	79	0
300	1	127	0	1	129	+0.16	1	155	0	1	159	0
600	0	255	0	1	64	+0.16	1	77	0	1	79	0
1200	0	127	0	0	129	+0.16	0	155	+0.16	0	159	0
2400	0	63	0	0	64	+0.16	0	77	+0.16	0	79	0
4800	0	31	0	0	32	-1.36	0	38	+0.16	0	39	0
9600	0	15	0	0	15	+1.73	0	19	-2.34	0	19	0
19200	0	7	0	0	7	+1.73	—	—	—	0	9	0
31250	0	4	-1.70	0	4	0	0	5	0	0	5	+2.40
38400	0	3	0	0	3	+1.73	—	—	—	0	4	0

Table 14-3 Examples of BRR Settings in Asynchronous Mode (4)

Bit Rate	XTAL Frequency (MHz)											
	14.7456			16			19.6608			20		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	130	-0.07	2	141	+0.03	2	174	-0.26	3	43	+0.88
150	2	95	0	2	103	+0.16	2	127	0	2	129	+0.16
300	1	191	0	1	207	+0.16	1	255	0	2	64	+0.16
600	1	95	0	1	103	+0.16	1	127	0	1	129	+0.16
1200	0	191	0	0	207	+0.16	0	255	0	1	64	+0.16
2400	0	95	0	0	103	+0.16	0	127	0	0	129	+0.16
4800	0	47	0	0	51	+0.16	0	63	0	0	64	+0.16
9600	0	23	0	0	25	+0.16	0	31	0	0	32	-1.36
19200	0	11	0	0	12	+0.16	0	15	0	0	15	+1.73
31250	—	—	—	0	7	0	0	9	-1.70	0	9	0
38400	0	5	0	—	—	—	0	7	0	0	7	+1.73

$$B = \text{OSC} \times 10^6 / [64 \times 2^{2n} \times (N + 1)]$$

B : Bit rate

N : BRR value ($0 \leq N \leq 255$)

OSC : Crystal oscillator frequency in MHz

n : Internal clock source (0, 1, 2, or 3)

The meaning of n is given by the table below:

n	CKS1	CKS0	Clock
0	0	0	\emptyset
1	0	1	$\emptyset/4$
2	1	0	$\emptyset/16$
3	1	1	$\emptyset/64$

Table 14-4 Examples of BRR Settings in Synchronous Mode

Bit Rate	XTAL Frequency (MHz)											
	2		4		8		10		16		20	
	n	N	n	N	n	N	n	N	n	N	n	N
100	—	—	—	—	—	—	—	—	—	—	—	—
250	1	249	2	124	2	249	—	—	3	124	—	—
500	1	124	1	249	2	124	—	—	2	249	—	—
1K	0	249	1	124	1	249	—	—	2	124	—	—
2.5M	0	99	0	199	1	99	1	124	1	199	1	249
5K	0	49	0	99	0	199	0	249	1	99	1	124
10K	0	24	0	49	0	99	0	124	0	199	0	249
25K	0	9	0	19	0	39	0	49	0	79	0	99
50K	0	4	0	9	0	19	0	24	0	39	0	49
100K	—	—	0	4	0	9	—	—	0	19	0	24
250K	0	0	0	1	0	3	0	4	0	7	0	9
500K			0	0	0	1	—	—	0	3	0	4
1M					0	0	—	—	0	1	—	—
2.5M											0	0

Notes:

Blank: No setting is available.

—: A setting is available, but the bit rate is inaccurate.

$$B = \text{OSC} / [8 \times 2^{2n} \times (N + 1)]$$

B : Bit rate

N : BRR value ($0 \leq N \leq 255$)

OSC : Crystal oscillator frequency in MHz

n : Internal clock source (0, 1, 2, or 3)

The meaning of n is given by the table below:

n	CKS1	CKS0	Clock
0	0	0	\emptyset
1	0	1	$\emptyset/4$
2	1	0	$\emptyset/16$
3	1	1	$\emptyset/64$

14.3 Operation

14.3.1 Overview

The SCI supports serial data transfer in both asynchronous and synchronous modes.

The communication format depends on settings in the SMR as indicated in table 14-5. The clock source and usage of the SCK pin depend on settings in the SMR and SCR as indicated in table 14-6.

Table 14-5 Communication Formats Used by SCI

C/A	SMR			Mode	Format	Parity	Stop Bit Length			
	CHR	PE	STOP							
0	0	0	0	Asynchronous	8-Bit data	None	1			
			1				2			
		1	0				Yes	1		
			1					2		
	1	0	0			Synchronous	8-Bit data	None	1	
			1						2	
		1	0						Yes	1
			1							2
1	—	—	—	Synchronous	8-Bit data	—	—			

Table 14-6 SCI Clock Source Selection

SMR C/A	SCR		Clock	
	CKE1	CKE0	Source	SCK Pin
0 (Async mode)	0	0	Internal	I/O port*
		1		Clock output at same frequency as baud rate
1 (Sync mode)	1	0	External	Clock input at 16 times the baud rate frequency
		1		
1 (Sync mode)	0	0	Internal	Serial clock output
		1		
1 (Sync mode)	1	0	External	Serial clock input
		1		

* Cannot be used by the SCI.

Transmitting and receiving operations in the two modes are described next.

14.3.2 Asynchronous Mode

In asynchronous mode, each character is individually synchronized by framing it with a start bit and stop bit.

Full duplex data transfer is possible because the SCI has independent transmit and receive sections. Double buffering in both sections enables the SCI to be programmed for continuous data transfer.

Figure 14-2 shows the general format of one character sent or received in the asynchronous mode. The communication channel is normally held in the mark state (High). Character transmission or reception starts with a transition to the space state (Low).

The first bit transmitted or received is the start bit (Low). It is followed by the data bits, in which the least significant bit (LSB) comes first. The data bits are followed by the parity bit, if present, then the stop bit or bits (High) confirming the end of the frame.

In receiving, the SCI synchronizes on the falling edge of the start bit, and samples each bit at the center of bit (at the 8th cycle of the internal serial clock, which runs at 16 times the bit rate).

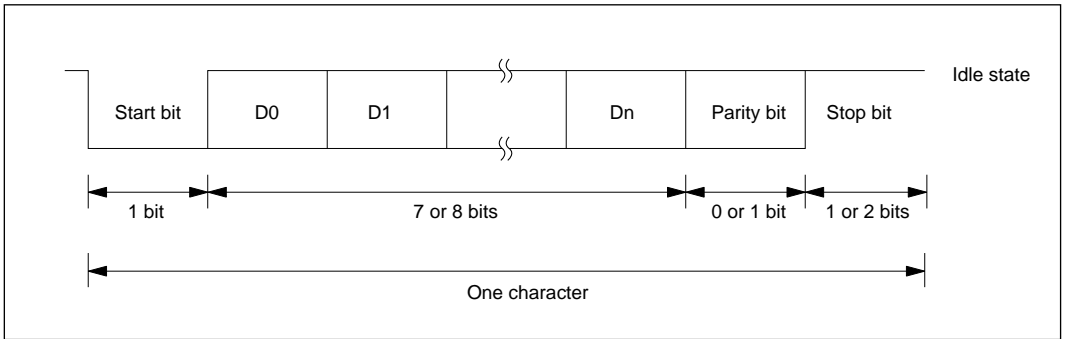


Figure 14-2 Data Format in Asynchronous Mode

1. Data Format: Table 14-7 lists the data formats that can be sent and received in asynchronous mode. Eight formats can be selected by bits in the SMR.

Table 14-7 Data Formats in Asynchronous Mode

SMR Bits			Data Format					
CHR	PE	STOP						
0	0	0	START	8-Bit data		STOP		
0	0	1	START	8-Bit data		STOP	STOP	
0	1	0	START	8-Bit data		P	STOP	
0	1	1	START	8-Bit data		P	STOP	STOP
1	0	0	START	7-Bit data	STOP			
1	0	1	START	7-Bit data	STOP	STOP		
1	1	0	START	7-Bit data	P	STOP		
1	1	1	START	7-Bit data	P	STOP	STOP	

Note:

START: Start bit

STOP: Stop bit

P: Parity bit

- 2. Clock:** In the asynchronous mode it is possible to select either an internal clock created by the on-chip baud rate generator, or an external clock input at the SCK pin. Refer to table 14-6.

If an external clock is input at the SCK pin, its frequency should be 16 times the desired baud rate.

If the internal clock provided by the on-chip baud rate generator is selected and the SCK pin is used for clock output, the output clock frequency is equal to the baud rate, and the clock pulse rises at the center of the transmit data bits. Figure 14-3 shows the phase relationship between the output clock and transmit data.

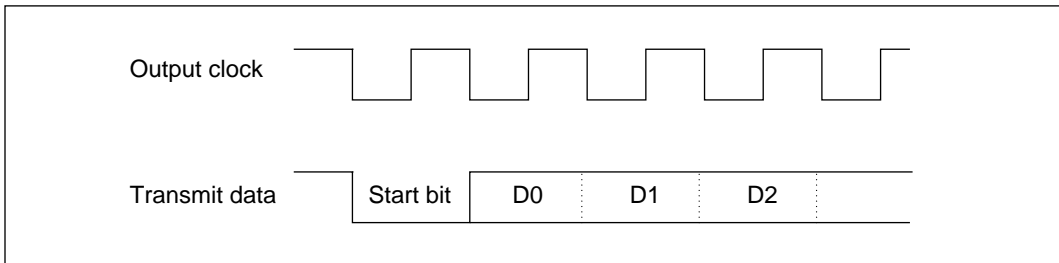


Figure 14-3 Phase Relationship between Clock Output and Transmit Data

3. Data Transmission and Reception

- **SCI Initialization:** Before data can be transmitted or received, the SCI must be initialized by software. To initialize the SCI, software must clear the TE and RE bits to 0, then execute the following procedure.

- (1) Set the desired communication format in the SMR.
- (2) Write the value corresponding to the desired bit rate in the BRR. (This step is not necessary if an external clock is used.)
- (3) Select the clock and enable desired interrupts in the SCR.
- (4) Set the TE and/or RE bit in the SCR to 1.

The TE and RE bits must both be cleared to 0 whenever the operating mode or data format is changed.

After changing the operating mode or data format, before setting the TE and RE bits to 1 software must wait for at least the transfer time for 1 bit at the selected baud rate, to make sure the SCI is initialized. If an external clock is used, the clock must not be stopped.

When clearing the TDRE bit during data transmission, to assure transfer of the correct data, do not clear the TDRE bit until after writing data in the TDR. Similarly, in receiving data, do not clear the RDRF bit until after reading data from the RDR.

- **Data Transmission:** The procedure for transmitting data is as follows.

- (1) Set up the desired transmitting conditions in the SMR, SCR, and BRR.
- (2) Set the TE bit in the SCR to 1.
The TXD pin will automatically be switched to output and one frame* of all 1's will be transmitted, after which the SCI is ready to transmit data.
- (3) Check that the TDRE bit is set to 1, then write the first byte of transmit data in the TDR. Next clear the TDRE bit to 0.

* A frame is the data for one character, including the start bit and stop bit(s).

- (4) The first byte of transmit data is transferred from the TDR to the TSR and sent in the designated format as follows.
 - i) Start bit (one 0 bit)
 - ii) Transmit data (seven or eight bits, starting from bit 0)
 - iii) Parity bit (odd or even parity bit, or no parity bit)
 - iv) Stop bit (one or two consecutive 1 bits)
- (5) Transfer of the transmit data from the TDR to the TSR makes the TDR empty, so the TDRE bit is set to 1.

If the TIE bit is set to 1, a transmit-end interrupt (TXI) is requested.

When the transmit function is enabled but the TDR is empty (TDRE = 1), the output at the TXD pin is held at 1 until the TDRE bit is cleared to 0.

- **Data Reception:** The procedure for receiving data is as follows.

- (1) Set up the desired receiving conditions in the SMR, SCR, and BRR.
- (2) Set the RE bit in the SCR to 1.

The RXD pin will automatically be switched to input and the SCI is ready to receive data.
- (3) The SCI synchronizes with the incoming data by detecting the start bit, and places the received bits in the RSR. At the end of the data, the SCI checks that the stop bit is 1. If the stop bit length is 2 bits, in ZTAT versions the SCI checks that both bits are 1, but in masked-ROM versions, only the first bit is checked.
- (4) When a complete frame has been received, the SCI transfers the received data to the RDR so that it can be read. If the character length is 7 bits, the most significant bit of the RDR is cleared to 0. At the same time, the SCI sets the RDRF bit in the SSR to 1. If the RIE bit is set to 1, a receive-end interrupt (RXI) is requested.
- (5) The RDRF bit is cleared to 0 when the CPU reads the SSR, then writes a 0 in the RDRF bit, or when the RDR is read by the data transfer controller (DTC). The RDR is then ready to receive the next character from the RSR.

When a frame is not received correctly, a receive error occurs. There are three types of receive errors, listed in table 14-8.

If a receive error occurs, the RDRF bit in the SSR is not set to 1. The corresponding error flag is set to 1 instead. If the RIE bit in the SCR is set to 1, a receive-error interrupt (ERI) is requested.

When a framing or parity error occurs, the RSR contents are transferred to the RDR. If an overrun error occurs, however, the RSR contents are not transferred to the RDR.

If multiple receive errors occur simultaneously, all the corresponding error flags are set to 1.

To clear a receive-error flag (ORER, FER, or PER), software must read the SSR, then write a 0 in the flag bit.

Table 14-8 Receive Errors

Name	Abbreviation	Description
Overrun error	ORER	Reception of the next frame ends while the RDRF bit is still set to 1. The RSR contents are not transferred to the RDR.
Framing error	FER	A stop bit is 0. The RSR contents are transferred to the RDR.
Parity error	PER	The parity of a frame does not match the value selected by the bit in the SMR. The RSR contents are transferred to the RDR.

14.3.3 Synchronous Mode

The synchronous mode is suited for high-speed, continuous data transfer. Each bit of data is synchronized with a serial clock pulse.

Continuous data transfer is enabled by the double buffering employed in both the transmit and receive sections of the SCI. Full duplex communication is possible because the transmit and receive sections are independent.

- 1. Data Format:** Figure 14-4 shows the communication format used in the synchronous mode. The data length is 8 bits for both the transmit and receive directions. The least significant bit (LSB) is sent and received first. Each bit of transmit data is output from the falling edge of the serial clock pulse to the next falling edge. Received bits are latched on the rising edge of the serial clock pulse.

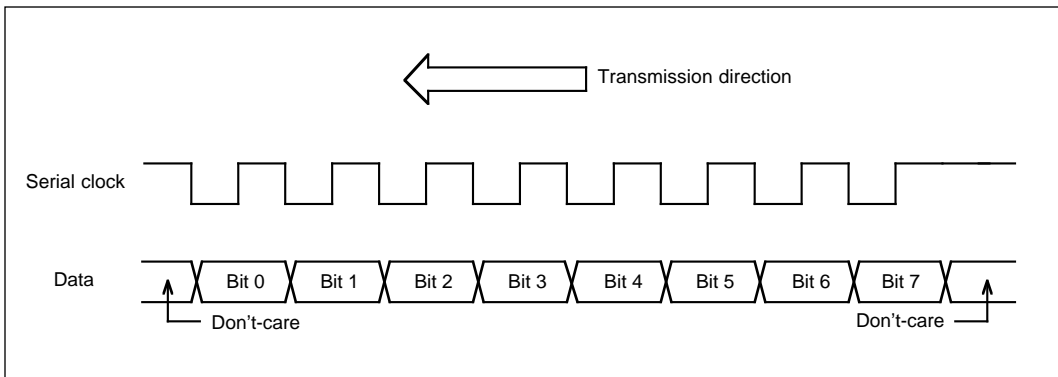


Figure 14-4 Data Format in Synchronous Mode

2. Clock: Either the internal serial clock created by the on-chip baud rate generator or an external clock input at the SCK pin can be selected in the synchronous mode. See table 14-6 for details.

3. Data Transmission and Reception

- **SCI Initialization:** Before data can be transmitted or received, the SCI must be initialized by software. To initialize the SCI, software must clear the TE and RE bits to 0 to disable both the transmit and receive functions, then execute the following procedure.

- (1) Write the value corresponding to the desired bit rate in the BRR. (This step is not necessary if an external clock is used.)
- (2) Select the clock in the SCR.
- (3) Select the synchronous mode in the SMR*.
- (4) Set the TE and/or RE bit to 1, and enable desired interrupts in the SCR.

The TE and RE bits must both be cleared to 0 whenever the operating mode or data format is changed. After changing the operating mode or data format, before setting the TE and RE bits to 1 software must wait for at least 1 bit transfer time at the selected communication speed, to make sure the SCI is initialized.

* The SCK pin is used for input or output according to the $\overline{C/A}$ bit in the serial mode register (SMR) and the CKE0 and CKE1 bits in the serial control register (SCR). (See table 14-6.) To prevent unwanted output at the SCK pin, pay attention to the order in which you set SMR and SCR.

When clearing the TDRE bit during data transmission, to assure correct data transfer, do not clear the TDRE bit until after writing data in the TDR. Similarly, in receiving data, do not clear the RDRF bit until after reading data from the RDR.

• **Data Transmission:** The procedure for transmitting data is as follows.

- (1) Set up the desired transmitting conditions in the SMR, BRR, and SCR.
- (2) Set the TE bit in the SCR to 1.
The TXD pin will automatically be switched to output, after which the SCI is ready to transmit data.
- (3) Check that the TDRE bit is set to 1, then write the first byte of transmit data in the TDR. Next clear the TDRE bit to 0.
- (4) The first byte of transmit data is transferred from the TDR to the TSR and sent, each bit synchronized with a clock pulse. Bit 0 is sent first.
Transfer of the transmit data from the TDR to the TSR makes the TDR empty, so the TDRE bit is set to 1. If the TIE bit is set to 1, a transmit-end interrupt (TXI) is requested.

The TDR and TSR function as a double buffer. Continuous data transmission can be achieved by writing the next transmit data in the TDR and clearing the TDRE bit to 0 while the SCI is transmitting the current data from the TSR.

If an internal clock source is selected, after transferring the transmit data from the TDR to the TSR, while transmitting the data from the TSR the SCI also outputs a serial clock signal at the SCK pin. When all data bits in the TSR have been transmitted, if the TDR is empty (TDRE = 1), serial clock output is suspended until the next data byte is written in the TDR and the TDRE bit is cleared to 0. During this interval the TXD pin is held at the value of the last bit transmitted.

If the external clock source is selected, data transmission is synchronized with the clock signal input at the SCK pin. When all data bits in the TSR have been transmitted, if the TDR is empty (TDRE = 1) but external clock pulses continue to arrive, the TXD pin outputs a string of bits equal to the last bit transmitted.

• **Data Reception:** The procedure for receiving data is as follows.

- (1) Set up the desired receiving conditions in the SMR, BRR, and SCR.

- (2) Set the RE bit in the SCR to 1.
The RXD pin will automatically be switched to input and the SCI is ready to receive data.
- (3) Incoming data bits are latched in the RSR on eight clock pulses.
When 8 bits of data have been received, the SCI sets the RDRF bit in the SSR to 1. If the RIE bit is set to 1, a receive-end interrupt (RXI) is requested.
- (4) The SCI transfers the received data byte to the RDR so that it can be read.
The RDRF bit is cleared when the program reads the RDRF bit in the SSR, then writes a 0 in the RDRF bit, or when the data transfer controller (DTC) reads the RDR.

The RDR and RSR function as a double buffer. Data can be received continuously by reading each byte of data from the RDR and clearing the RDRF bit to 0 before the last bit of the next byte is received.

In general, an external clock source should be used for receiving data.

If an internal clock source is selected, the SCI starts receiving data as soon as the RE bit is set to 1. The serial clock is also output at the SCK pin. The SCI continues receiving until the RE bit is cleared to 0.

If the last bit of the next data byte is received while the RDRF bit is still set to 1, an overrun error occurs and the ORER bit is set to 1. If the RIE bit is set to 1, a receive-error interrupt (ERI) is requested. The data received in the RSR are not transferred to the RDR when an overrun error occurs.

After an overrun error, reception of the next data is enabled when the ORER bit is cleared to 0.

- **Simultaneous Transmit and Receive:** The procedure for transmitting and receiving simultaneously is as follows:
 - (1) Set up the desired communication conditions in the SMR, BRR, and SCR.
 - (2) Set the TE and RE bits in the SCR to 1.
The TXD and RXD pins are automatically switched to output and input, respectively, and the SCI is ready to transmit and receive data.
 - (3) Data transmitting and receiving start when the TDRE bit in the SSR is cleared to 0.
 - (4) Data are sent and received in synchronization with eight clock pulses.

- (5) First, the transmit data are transferred from the TDR to the TSR. This makes the TDR empty, so the TDRE bit is set to 1. If the TIE bit is set to 1, a transmit-end interrupt (TXI) is requested.

If continuous data transmission is desired, the CPU must read the TDRE bit in the SSR, write the next transmit data in the TDR, then clear the TDRE bit to 0. Alternatively, the DTC can write the next transmit data in the TDR, in which case the TDRE bit is cleared automatically.

If the TDRE bit is not cleared to 0 by the time the SCI finishes sending the current byte from the TSR, the TXD pin continues to output the last bit in the TSR.

- (6) In the receiving section, when 8 bits of data have been received they are transferred from the RSR to the RDR and the RDRF bit in the SSR is set to 1. If the RIE bit is set to 1, a receive-end interrupt (RXI) is requested.

- (7) To clear the RDRF bit software read the RDRF bit in the SSR, read the data in the RDR, then write a 0 in the RDRF bit. Alternatively, the DTC can read the RDR, in which case the RDRF bit is cleared automatically.

For continuous data reception, the RDRF bit must be cleared to 0 before the last bit of the next byte of data is received.

If the last bit of the next byte is received while the RDRF bit is still set to 1, an overrun error occurs. The error is handled as described under “Data Reception” above. The overrun error does not affect the transmit section of the SCI, which continues to transmit normally.

14.4 CPU Interrupts and DTC Interrupts

The SCI can request three types of interrupts: transmit-end (TXI), receive-end (RXI), and receive-error (ERI). Interrupt requests are enabled or disabled by the TIE and RIE bits in the SCR. Independent signals are sent to the interrupt controller for each type of interrupt. The transmit-end and receive-end interrupt request signals are obtained from the TDRE and RDRF flags. The receive-error interrupt request signal is the logical OR of the three error flags: overrun error (ORER), framing error (FER), and parity error (PER). Table 14-9 lists information about these interrupts.

Table 14-9 SCI Interrupts

Interrupt	Description	DTC Service Available?	Priority
ERI	Receive-error interrupt, requested when ORER, FER, or PER is set.	No	High
RXI	Receive-end interrupt, requested when RDRF is set.	Yes	↑ Low
TXI	Transmit-end interrupt, requested when TDRE is set.	Yes	

The TXI and RXI interrupts can be served by the data transfer controller (DTC) to have a data transfer performed. When the DTC serves one of these interrupts, it clears the TDRE or RDRF bit to 0 under the following conditions, which differ between the two bits.

When invoked by a TXI request, if the DTC writes to the TDR, it automatically clears the TDRE bit to 0. When invoked by an RXI request, if the DTC reads from the RDR, it automatically clears the RDRF bit to 0.

See section 6, “Data Transfer Controller” for further information on the DTC.

14.5 Application Notes

Application programmers should note the following features of the SCI.

- 1. TDR Write:** The TDRE bit in the SSR is simply a flag that indicates that the TDR contents have been transferred to the TSR. The TDR contents can be rewritten regardless of the TDRE value. If a new byte is written in the TDR while the TDRE bit is 0, before the old TDR contents have been moved into the TSR, the old byte will be lost. Normally, software should check that the TDRE bit is set to 1 before writing to the TDR.
- 2. Multiple Receive Errors:** Table 14-10 lists the values of flag bits in the SSR when multiple receive errors occur, and indicates whether the RSR contents are transferred to the RDR.

Table 14-10 SSR Bit States and Data Transfer When Multiple Receive Errors Occur

Receive Error	SSR Bits				
	RDRF	ORER	FER	PER	RSR to RDR*2
Overrun error	1*1	1	0	0	No
Framing error	0	0	1	0	Yes
Parity error	0	0	0	1	Yes
Overrun + framing errors	1*1	1	1	0	No
Overrun + parity errors	1*1	1	0	1	No
Framing + parity errors	0	0	1	1	Yes
Overrun + framing + parity errors	1*1	1	1	1	No

*1 Set to 1 before the overrun error occurs.

*2 Yes: The RSR contents are transferred to the RDR.

No: The RSR contents are not transferred to the RDR.

3. Line Break Detection: When the RXD pin receives a continuous stream of 0's in the asynchronous mode (line-break state), a framing error occurs because the SCI detects a 0 stop bit. The value H'00 is transferred from the RSR to the RDR. Software can detect the line-break state as a framing error accompanied by H'00 data in the RDR.

The SCI continues to receive data, so if the FER bit is cleared to 0 another framing error will occur.

4. Sampling Timing and Receive Margin in Asynchronous Mode: The serial clock used by the SCI in asynchronous mode runs at 16 times the bit rate. The falling edge of the start bit is detected by sampling the RXD input on the falling edge of this clock. After the start bit is detected, each bit of receive data in the frame (including the start bit, parity bit, and stop bit or bits) is sampled on the rising edge of the serial clock pulse at the center of the bit. See figure 14-5.

It follows that the receive margin can be calculated as in equation (1).

When the absolute frequency deviation of the clock signal is 0 and the clock duty factor is 0.5, data can theoretically be received with distortion up to the margin given by equation (2). This is a theoretical limit, however. In practice, system designers should allow a margin of 20% to 30%.

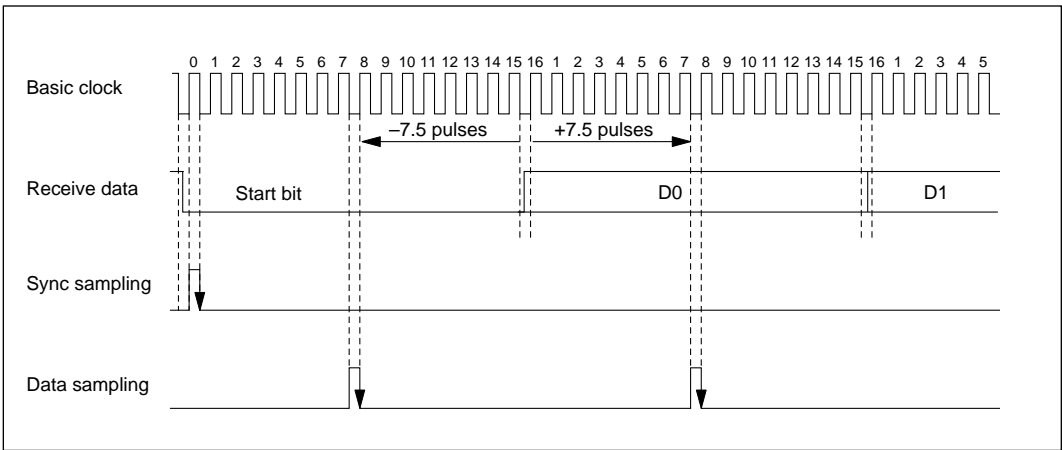


Figure 14-5 Sampling Timing (Asynchronous Mode)

$$M = \{ (0.5 - 1/2N) - (D - 0.5)/N - (L - 0.5)F \} \times 100 \text{ [\%]} \quad (1)$$

N: Receive margin

N: Ratio of basic clock to bit rate (16)

D: Duty factor of clock—ratio of High pulse width to Low width (0.5 to 1.0)

L: Frame length (9 to 12)

F: Absolute clock frequency deviation

When $D = 0.5$ and $F = 0$

$$M = (0.5 - 1/2 \times 16) \times 100 \text{ [\%]} = 46.875\% \quad (2)$$