SRL0 Serial Port Unit

Summary

This document provides detailed reference information with respect to the serial port unit peripheral devices.

Core Reference
CR0111 (v2.0) March 13, 2008

The serial communications port peripheral devices can be configured for communications between a microprocessor and peripheral devices, or for multiprocessor communications.

Features

- Full Duplex - capable of simultaneous transmission and reception
- Single byte buffering of received data
- Four modes of operation (1 synchronous, 3 asynchronous)
  - Mode 0 : Synchronous mode with fixed Baud rate
  - Mode 1 : 8-bit UART mode with variable Baud rate
  - Mode 2 : 9-bit UART mode with fixed Baud rate
  - Mode 3 : 9-bit UART mode with variable Baud rate
- Falling edge detection on external receive input
- Dedicated internal Baud Rate Generator
- Additional Timer Unit for baud rate generation
- Support for multiprocessor communications
- Wishbone-compliant (**SRL0_W only**)

Available Devices

Both Wishbone and non-Wishbone versions of the serial port unit are available – the SRL0_W and SRL0 respectively. These devices can be found in the FPGA Peripherals integrated library (`FPGA Peripherals.IntLib`), located in the `\Library\Fpga` folder of the installation.

SRL0_W – Wishbone-Compliant Version

The SRL0_W is a Wishbone-compliant, serial communications port, for use with the following Wishbone-compliant microcontrollers/processors:

- TSK52B_W (and its debug-enabled version - TSK52B_WD)
- TSK3000A
- PPC405A.

Functional Description

Symbol

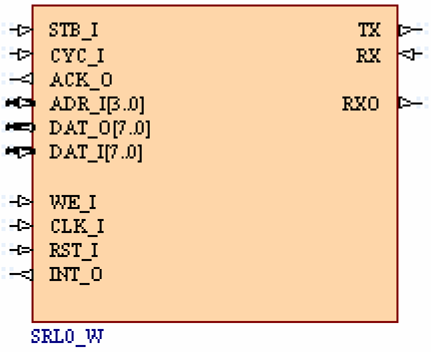

Figure 1. Serial Port Unit Symbol – Wishbone version (SRL0_W)

Pin Description

Table 1. SRL0_W pin description

Name	Type	Polarity/Bus size	Description
Control Signals			
CLK_I	I	Rise	External system clock
RST_I	I	High	External system reset
Microcontroller Interface Signals			
STB_I	I	High	Strobe signal. When asserted, indicates the start of a valid Wishbone data transfer cycle
CYC_I	I	High	Cycle signal. When asserted, indicates the start of a valid Wishbone cycle
ACK_O	O	High	Standard Wishbone-device acknowledgement signal. When this signal goes high, the Serial Interface Unit (Wishbone Slave) has finished execution of the requested action and the current bus cycle is terminated
ADR_I	I	4	Standard Wishbone address bus, used to select an internal register of the Wishbone slave device for writing to/reading from.
DAT_O	O	8	Data to be sent to an external Wishbone master device (e.g. host microcontroller).
DAT_I	I	8	Data received from an external Wishbone master device (e.g. host microcontroller).
DAT_O	O	8	Data to be sent to an external Wishbone master device (e.g. host microcontroller).
WE_I	I	Level	Write enable signal. Used to indicate whether the current local bus cycle is a Read or Write cycle. 0 – Read 1 - Write

Name	Type	Polarity/Bus size	Description
INT_O	O	High	Interrupt signal. If either bit 1 or bit 0 in the Serial Interface Control register is set, this signal is activated. This alerts the host microcontroller to the fact that transmission or reception (respectively) of serial data has completed
Serial Interface Input Signals			
RX	I	Fall	Serial data receive
Serial Interface Output Signals			
TX	O		Serial data transmit
RXO	O		Serial data transmit (in Mode 0)

Hardware Description

Block Diagram

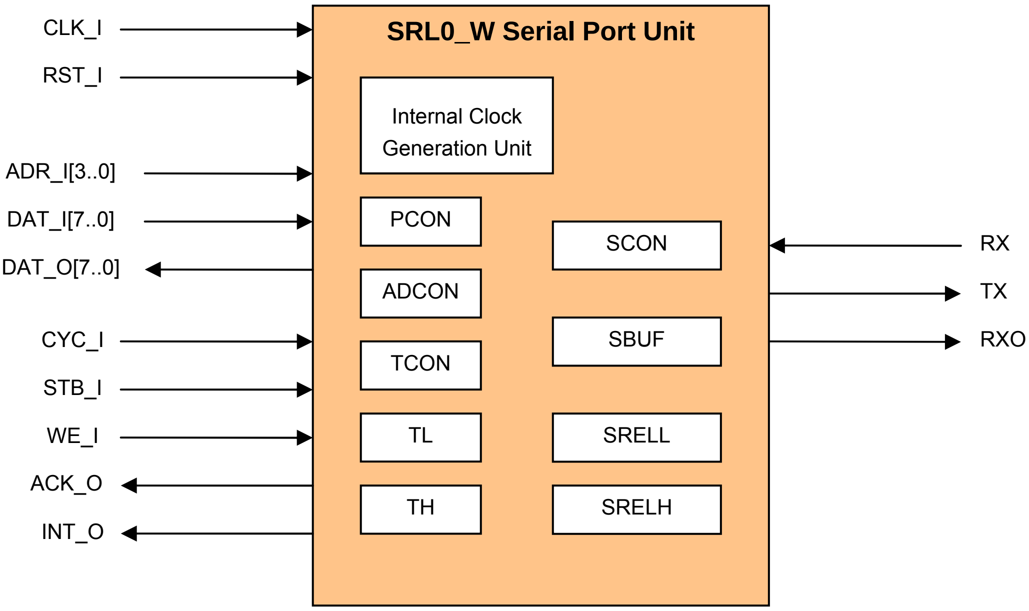

Figure 2. SRL0_W block diagram

Special Function Registers

Power Control Register (PCON)

Table 2. The PCON register

MSB							LSB
SMOD	F4	F3	F2	F1	F0	0	0

Table 3. The PCON register bit functions

Bit	Symbol	Function
PCON.7	SMOD	Double baud rate bit. If the additional Timer Unit is used to generate the baud rate and SMOD is set (1), the baud rate is doubled when the Serial Port is used in modes 1,2 or 3
PCON.6	F4	General purpose Flag 4 available for user
PCON.5	F3	General purpose Flag 3 available for user
PCON.4	F2	General purpose Flag 2 available for user
PCON.3	F1	General purpose Flag 1 available for user
PCON.2	F0	General purpose Flag 0 available for user
PCON.1	0	This bit is read only and is permanently cleared (0)
PCON.0	0	This bit is read only and is permanently cleared (0)

Serial Interface Control Register (SCON)

The function of the serial port depends on the setting of the Serial Interface Control Register SCON.

Table 4. The SCON register

MSB							LSB
SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Table 5. The SCON register bit functions

Bit	Symbol	Function
SCON.7	SM0	*Defines operational mode of the serial interface (see* Table 18)
SCON.6	SM1	*Defines operational mode of the serial interface (see* Table 18)
SCON.5	SM2	When set, enables multiprocessor communication feature for operational modes 2 and 3
SCON.4	REN	Controlled by software to enable/disable reception: 0 = disable reception of serial data 1 = enable reception of serial data
SCON.3	TB8	The 9th transmitted bit in Modes 2 and 3. Set or cleared by the CPU, depending on the function it performs (parity check, multiprocessor communication etc
SCON.2	RB8	The 9th bit received in Modes 2 and 3. In Mode 1, if SM2 is 0, RB8 is the stop bit. In Mode 0 this bit is not used. Must be cleared by software
SCON.1	TI	Transmit interrupt flag, set by hardware after completion of a serial transfer. Must be cleared by software
SCON.0	RI	Receive interrupt flag, set by hardware after completion of a serial reception. Must be cleared by software

Table 6. Serial Interface operational modes

SM0	SM1	Mode	Description	Baud Rate
0	0	0	Shift Register	F_{CLK_I}/12
0	1	1	8-bit UART	Variable
1	0	2	9-bit UART	F_{CLK_I} /32 or F_{CLK_I} /64
1	1	3	9-bit UART	Variable

Note: Speed in Mode 2 depends on the state of bit 7 (SMOD) in the Special Function Register PCON. When SMOD = 1, F_{CLK_I} /32 is used as the baud rate.

Serial Data Buffer Register (SBUF)

The serial data interface consists of two separate registers, a transmit register and a receive register (with buffer), both of which are accessed through the Special Function Register SBUF. Writing to the SBUF register loads data into the serial transmit register and starts the transmission. Reading the SBUF register accesses data in the serial receive buffer.

The SRL0_W is full duplex, meaning that it is capable of simultaneous transmission and reception of serial data. Single-byte buffering is incorporated at the reception stage, enabling a second byte of data to be received whilst the first is still being read. Bear in mind that the first byte of received data must be read out before reception of the subsequent byte has completed, otherwise the first byte of data will be lost (overwritten with the second byte).

Baud Rate Generator Reload Register (SREL)

The SRL0_W includes a dedicated internal Baud Rate Generator which can be used for baud rate generation in Modes 1 and 3. The baud rate is generated based on the overflow rate of a 10-bit Baud Timer register – internal to the Generator Unit.

The Baud Timer register is incremented after every 2 cycles of the external system clock (CLK_I) signal. The required clock signal for the timer is obtained from the Internal Clock Generation Unit (providing an output clock signal to the timer which is CLK_I/2).

When the Baud Timer register reaches overflow, it is automatically loaded with the value stored in the SREL register.

The SREL register is a 16-bit register, which is further sub-divided into two 8-bit registers – SRELL (low 8 bits) and SRELH (high 8 bits). These two sub-registers are used to provide the 10-bit value for SREL as follows:

SREL9..8 ← SRELH1..0

SREL7..0 ← SRELL7..0

Timer Control Register (TCON)

Table 7. The TCON register

MSB							LSB
F11	TR	F10	F9	F8	F7	F6	F5

Table 8. The TCON register bit functions

Bit	Symbol	Function
TCON.7	F11	General purpose Flag 11 available for user
TCON.6	TR	Timer Run control bit. If cleared, Timer Unit is stopped.
TCON.5	F10	General purpose Flag 10 available for user
TCON.4	F9	General purpose Flag 9 available for user
TCON.3	F8	General purpose Flag 8 available for user
TCON.2	F7	General purpose Flag 7 available for user
TCON.1	F6	General purpose Flag 6 available for user
TCON.0	F5	General purpose Flag 5 available for user

Timer Register (T)

The SRL0_W includes an additional Timer Unit which can be used for baud rate generation in Modes 1 and 3. The unit has a 16-bit Timer register, which is further sub-divided into two 8-bit registers – TL (low 8 bits) and TH (high 8 bits).

The Timer is hard-coded to operate as an 8-bit auto-reload timer. The reload value is stored in the TH register.

The TL register is incremented after every 12 cycles of the external system clock (CLK_I) signal. The required clock signal for the timer is obtained from the Internal Clock Generation Unit (providing an output clock signal to the timer which is CLK_I/12).

When the TL register reaches overflow, the value stored in the TH register is automatically copied to TL.

Operation of the Timer is controlled by the TR bit in the TCON register. If TR is '1', the Timer is running. If TR is '0' the Timer stops.

A/D Converter Control Register (ADCON)

Table 9. The ADCON register

MSB							LSB
BD	-	-	-	-	-	-	-

Table 10. The ADCON register bit functions

Bit	Symbol	Function
ADCON.7	BD	Used to determine the source for baud rate generation when the serial interface is operating in Modes 1 and 3. 0 = Timer Unit 1 = Internal Baud rate Generator (using SREL register)
ADCON.6	-	Not Used
ADCON.5	-	Not Used
ADCON.4	-	Not Used
ADCON.3	-	Not Used
ADCON.2	-	Not Used
ADCON.1	-	Not Used
ADCON.0	-	Not Used

Register Reset Values

Table 11 shows the values contained in each of the SRL0_W’s internal registers after an external system reset has been received on the RST_I input.

Table 11. Register reset values

Register	Value after reset
PCON	00000000
SCON	00000000
SBUF	00000000
SRELL	11011001
SRELH	00000011
TCON	00000000
TL	00000000
TH	00000000
ADCON	00000000

Serial Interface Operation

The Serial Interface can operate in the following four modes:

Mode 0

Pin RXD serves as serial input, with RXDO as serial output. TXD outputs the shift clock.

8 bits are transmitted/received with LSB first.

In Mode 0, the baud rate is fixed at 1/12 of the crystal frequency.

Mode 1

Pin RXD serves as serial input and TXD serves as serial output. No external shift clock is used.

10 bits are transmitted/received: a start bit (always 0), 8 data bits (LSB first) and a stop bit (always 1).

On reception, the start bit synchronizes the transmission, 8 data bits are made available by reading the SBUF register and the stop bit sets flag RB8 in the SCON register.

In mode 1, the baud rate is variable and can be specified by either the internal baud rate generator or the additional Timer Unit.

Mode 2

This mode is similar to Mode 1, with two differences:

- The baud rate is fixed at 1/32 or 1/64 of oscillator frequency and
- 11 bits are transmitted/received: a start bit (0), 8 data bits (LSB first), a programmable 9th bit, and a stop bit (1).

At transmission, bit TB8 in the SCON register is output as the 9th bit. This bit could be used to provide parity checking, by moving the parity bit (P in the PSW register) into TB8. On reception of serial data, the 9th bit is moved into bit RB8 in the SCON register and the stop bit is ignored.

Mode 3

The only difference between Mode 2 and Mode 3 is that the baud rate is variable in Mode 3. Either the internal baud rate generator or the additional Timer Unit can be used to specify the baud rate.

In all four operational modes, any instruction that has the SBUF register as its destination will initiate a serial transmission of data. Initiating the reception of serial data depends on the particular operating mode. In Mode 0, reception is initiated by setting the appropriate flags in the SCON register as follows:

- RI = 0
- REN = 1

In Modes 1, 2 and 3, a start bit when REN = 1 starts the reception of serial data.

Multiprocessor Communications

When the serial interface is operating in Modes 2 or 3, the programmable 9th data bit can be utilized to enable multiprocessor communications.

When serial data is received in either of these modes, the 9th data bit is loaded into bit 2 (RB8) of the SCON register. The next bit received in a serial reception is the stop bit. The serial interface can be programmed to generate an interrupt when this stop bit is received – by setting the receive interrupt flag (RI) in the SCON register - ONLY if RB8 is a '1'. This feature is enabled only when bit 5 (SM2) of the SCON register is set.

By using the state of the 9th data bit to distinguish between address ('1') and data ('0') bytes sent from a microprocessor, this feature can be used to establish efficient multiprocessor communications in the following manner:

- Each Slave processor has bit SM2 in its SCON register set to 1.
- The Master processor sends, via the serial interface, an address byte. This address byte identifies which Slave processor the Master wishes to communicate with. The 9th bit in this serial transmission is set.
- All Slave microprocessors connected to the Master will be interrupted, enabling each to determine whether or not it is the Slave that the Master wishes to communicate with.
- The addressed Slave processor will clear its SM2 bit and prepare to receive the serial transmission from the Master. All other Slave processors leave the SM2 bit set.

- The Master processor then sends its data, with the 9th bit cleared. The addressed Slave receives the transmission of data. As the 9th bit is '0' for data transmission, RB8 will be '0' in all other Slaves and no interrupts will be generated. These Slaves carry on with what they were previously doing.
- After the serial data from the Master has been completely received, the addressed Slave sets the SM2 bit back to '1'.

Baud Rate Generation

The following sections detail the methods and/or formulas involved in generating the baud rate for each of the SRL0_W's operational modes.

Mode 0

When the serial interface is operating in Mode 0, the baud rate is fixed and is defined as follows:

$$\text{baud rate} = \frac{F_{CLK\_I}}{12}$$

Mode 2

When the serial interface is operating in Mode 2, the baud rate depends on the state of bit 7 (SMOD) in the PCON register (see 0) and is defined as follows:

$$\text{baud rate} = \frac{2^{SMOD}}{64} \times F_{CLK\_I}$$

Therefore, if SMOD = 0, the baud rate will be 1/64 of the frequency of the Wishbone clock input to the device. If SMOD = 1, the baud rate will be 1/32 of this frequency.

Modes 1 and 3

When the serial interface is operating in either Mode 1 or Mode 3, the baud rate is variable and can be generated using either the dedicated internal baud rate generator or the Timer Unit. In both cases, bit 7 (SMOD) of the PCON register can be used to provide doubling of the baud rate (SMOD = 1).

Using the Internal Baud Rate Generator

To use the internal baud rate generator, bit 7 (BD) of the ADCON register (see 0) must be set. In this case, the Baud Rate Generator Reload register (SREL) is used in providing the required prescaling.

The baud rate is generated as follows:

$$\text{baud rate} = \frac{2^{SMOD}}{64 \times (2^{10} - \text{SREL})} \times F_{CLK\_I}$$

Note that the 10-bit value for SREL comes from the two, 8-bit registers SRELL and SRELH. The full 8 bits of SRELL are used (SREL7..0), while only the lowest two bits of SRELH are used (SREL9..8).

Using the Timer Unit

To use the Timer Unit to generate the baud rate, bit 7 (BD) of the ADCON register (see 0) must be cleared.

The Timer is hard-coded to operate as a 'timer' in auto-reload mode. When the TL register reaches the overflow state, it is automatically loaded with the value stored in the TH register. It is the value in the TH register that is used in determining the required prescaling.

The baud rate is generated as follows:

$$\text{baud rate} = \frac{2^{SMOD}}{32 \times 12 \times (256 - \text{TH})} \times F_{CLK\_I}$$

Serial Interface Timing

Serial Transmission

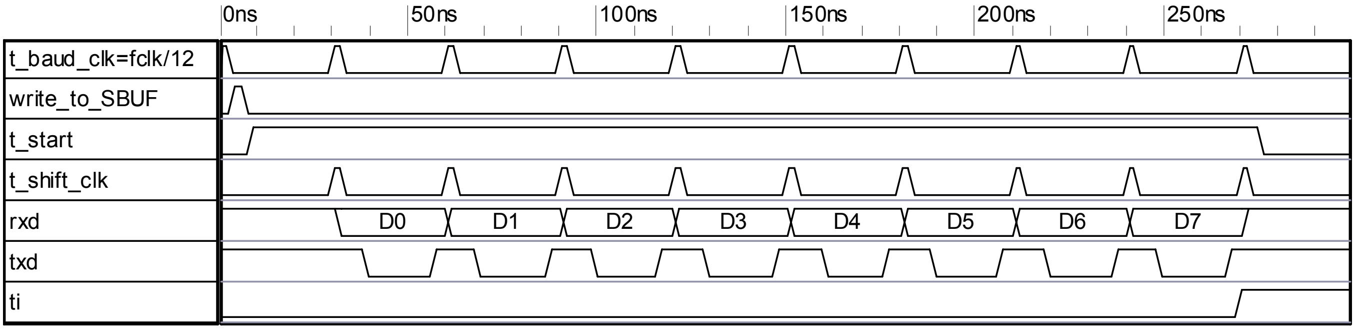

Figure 3. Mode 0 transmission

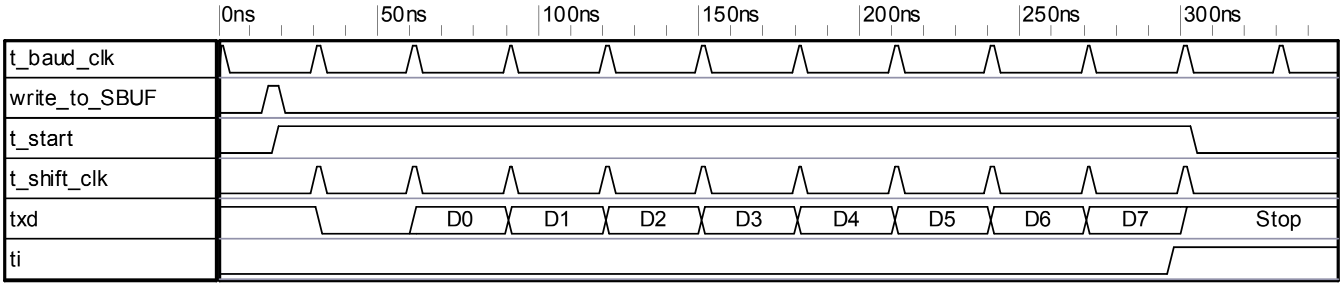

Figure 4. Mode 1 transmission

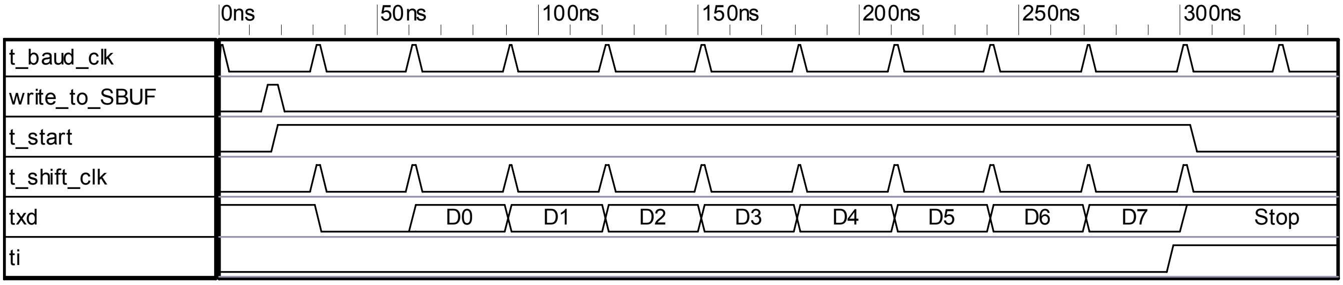

Figure 5. Modes 2 and 3 transmission

Serial Reception

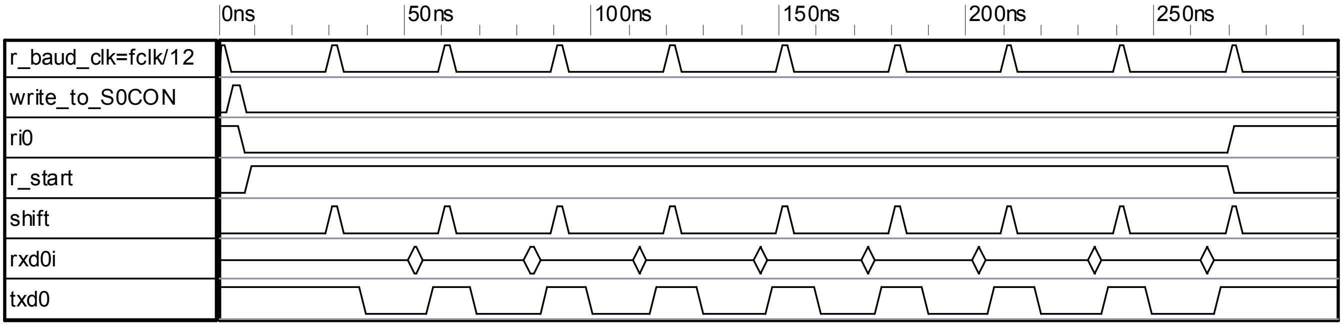

Figure 6. Mode 0 reception

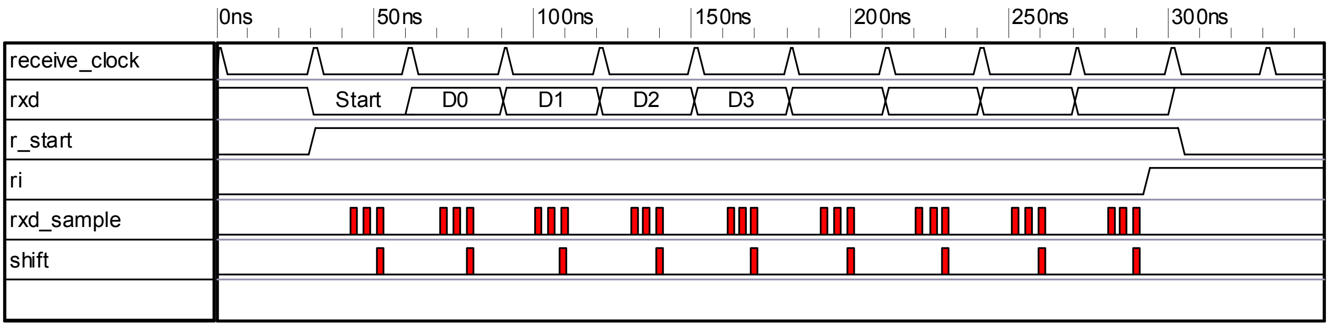

Figure 7. Mode 1 reception

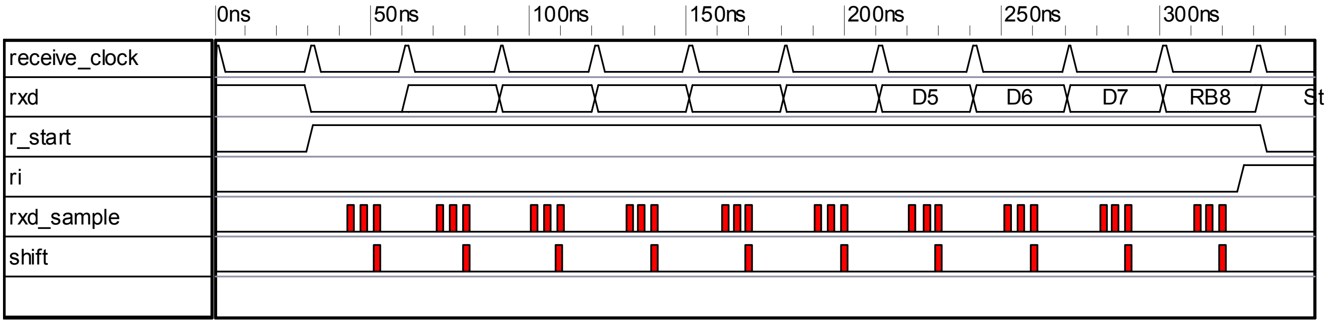

Figure 8. Modes 2 and 3 reception

Reading/Writing the Internal Registers

The host microcontroller can read/write any of the SRL0_W's internal Special Function Registers. Selection of a particular register is achieved by supplying the unique, 4-bit binary ID address code of the register. This code is sent to the SRL0_W and appears at the ADR_I input. Table 12 shows the unique address IDs associated with each of the addressable registers.

Table 12. Internal register unique address IDs

Register	Unique Register Address ID
PCON	0000
SCON	0001
SBUF	0010
SRELL	0011
SRELH	0100
TCON	0101
TL	0110
TH	0111
ADCON	1000

Writing to an Internal Register

Data is written from the host microcontroller to an internal register in the SRL0_W, in accordance with the standard Wishbone data transfer handshaking protocol. The write operation occurs on the rising edge of the CLK_I input and can be summarized as follows:

- The host presents the unique 4-bit address ID for the register to be written (see Table 12) on its ADR_O output and a valid byte of data on its DAT_O output. It then asserts its WE_O signal, to specify a Write cycle.
- The SRL0_W receives the address ID on its ADR_I input and prepares to receive data into the selected register.
- The host asserts its STB_O and CYC_O outputs, indicating that the transfer is to begin. The SRL0_W, which monitors its STB_I and CYC_I inputs on each rising edge of the CLK_I signal, reacts to this assertion by latching the byte of data appearing at its DAT_I input, into the specified target register, and asserting its ACK_O signal – to indicate to the host that the data has been received.

- The host, which monitors its ACK_I input on each rising edge of the CLK_I signal, responds by negating the STB_O and CYC_O signals. At the same time, the SRL0_W negates the ACK_O signal and the data transfer cycle is naturally terminated.

Reading from an Internal Register

Data is read from one of the SRL0_W's internal registers, in accordance with the standard Wishbone data transfer handshaking protocol. This data transfer cycle can be summarized as follows:

- The host presents the unique 4-bit address ID for the register to be read (see Table 12) on its ADR_O output. It then negates its WE_O signal, to specify a Read cycle
- The SRL0_W receives the address ID on its ADR_I input and prepares to transmit data from the selected register
- The host asserts its STB_O and CYC_O outputs, indicating that the transfer is to begin. The SRL0_W, which monitors its STB_I and CYC_I inputs on each rising edge of the CLK_I signal, reacts to this assertion by presenting the valid byte of data at its DAT_O output and asserting its ACK_O signal – to indicate to the host that valid data is present
- The host, which monitors its ACK_I input on each rising edge of the CLK_I signal, responds by latching the byte of data appearing at its DATA_I input and negating the STB_O and CYC_O signals. At the same time, the SRL0_W negates the ACK_O signal and the data transfer cycle is naturally terminated.

SRL0 – Non-Wishbone Version

The SRL0 is a serial communications port, for use with the TSK80A Microcontroller.

Functional Description

Symbol

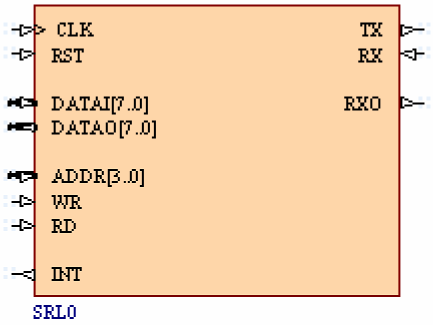

Figure 9. Serial Port Unit Symbol – non-Wishbone version (SRL0)

Pin Description

Table 13. SRL0 pin description

Name	Type	Polarity/Bus size	Description
Control Signals			
CLK	I	Rise	External system clock
RST	I	High	External system reset
Microcontroller Interface Signals			
DATAI	I	8	Data received from host microcontroller.
DATAO	O	8	Data to be sent to host microcontroller.
ADDR	I	4	Address bus, used to select an internal register of the device for writing to/reading from.
WR	I	High	Write enable signal. When active, data can be written to an internal register of the device.
RD	I	High	Read enable signal. When active, data can be read from an internal register of the device
INT	O	High	Interrupt signal. If either bit 1 or bit 0 in the Serial Interface Control register is set, this signal is activated. This alerts the host to the fact that transmission or reception (respectively) of serial data has completed.

Name	Type	Polarity/Bus size	Description
Serial Interface Input Signals			
RX	I	-	Serial data receive
Serial Interface Output Signals			
TX	O	-	Serial data transmit
RXO	O	-	Serial data transmit (in Mode 0)

Hardware Description

Block Diagram

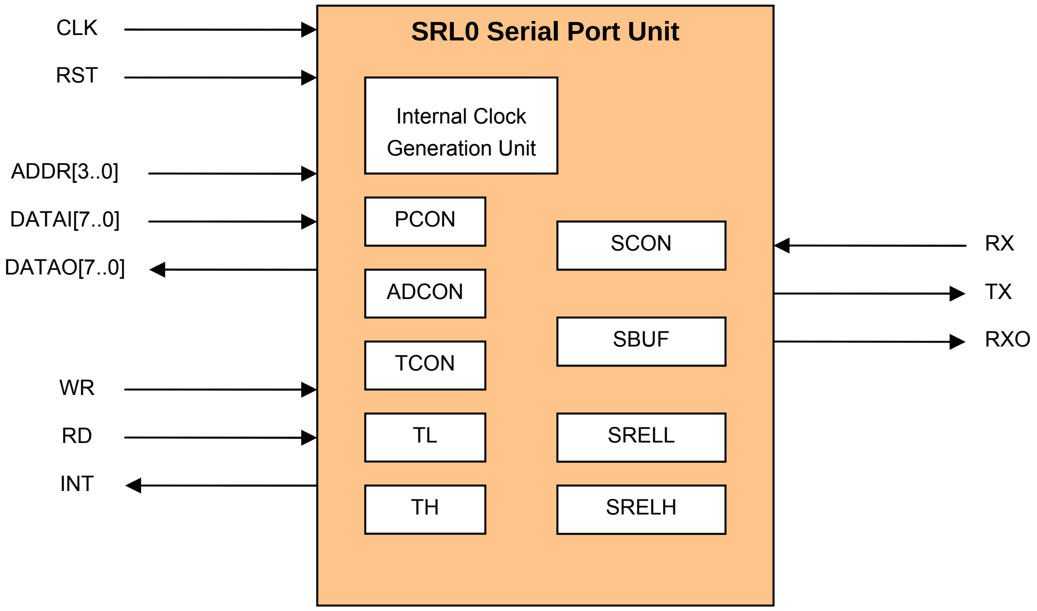

Figure 10. SRL0 bock diagram

Special Function Registers

Power Control Register (PCON)

Table 14. The PCON register

MSB							LSB
SMOD	F4	F3	F2	F1	F0	0	0

Table 15. The PCON register bit functions

Bit	Symbol	Function
PCON.7	SMOD	Double baud rate bit. If the additional Timer Unit is used to generate the baud rate and SMOD is set (1), the baud rate is doubled when the Serial Port is used in modes 1,2 or 3
PCON.6	F4	General purpose Flag 4 available for user
PCON.5	F3	General purpose Flag 3 available for user
PCON.4	F2	General purpose Flag 2 available for user

Bit	Symbol	Function
PCON.3	F1	General purpose Flag 1 available for user
PCON.2	F0	General purpose Flag 0 available for user
PCON.1	0	This bit is read only and is permanently cleared (0)
PCON.0	0	This bit is read only and is permanently cleared (0)

Serial Interface Control Register (SCON)

The function of the serial port depends on the setting of the Serial Interface Control Register SCON.

Table 16. The SCON register

MSB							LSB
SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Table 17. The SCON register bit functions

Bit	Symbol	Function
SCON.7	SM0	*Defines operational mode of the serial interface (see* Table 18)
SCON.6	SM1	*Defines operational mode of the serial interface (see* Table 18)
SCON.5	SM2	When set, enables multiprocessor communication feature for operational modes 2 and 3
SCON.4	REN	Controlled by software to enable/disable reception: 0 = disable reception of serial data 1 = enable reception of serial data
SCON.3	TB8	The 9th transmitted bit in Modes 2 and 3. Set or cleared by the CPU, depending on the function it performs (parity check, multiprocessor communication etc)
SCON.2	RB8	The 9th bit received in Modes 2 and 3. In Mode 1, if SM2 is 0, RB8 is the stop bit. In Mode 0 this bit is not used. Must be cleared by software
SCON.1	TI	Transmit interrupt flag, set by hardware after completion of a serial transfer. Must be cleared by software
SCON.0	RI	Receive interrupt flag, set by hardware after completion of a serial reception. Must be cleared by software

Table 18. Serial Interface operational modes

SM0	SM1	Mode	Description	Baud Rate
0	0	0	Shift Register	F_{CLK} /12
0	1	1	8-bit UART	Variable
1	0	2	9-bit UART	F_{CLK} /32 or F_{CLK} /64
1	1	3	9-bit UART	Variable

Note: Speed in Mode 2 depends on the state of bit 7 (SMOD) in the Special Function Register PCON. When SMOD = 1, F_{CLK} /32 is used as the baud rate.

Serial Data Buffer Register (SBUF)

The serial data interface consists of two separate registers, a transmit register and a receive register (with buffer), both of which are accessed through the Special Function Register SBUF. Writing to the SBUF register loads data into the serial transmit register and starts the transmission. Reading the SBUF register, accesses data in the serial receive buffer.

The SRL0 is full duplex, meaning that it is capable of simultaneous transmission and reception of serial data. Single-byte buffering is incorporated at the reception stage, enabling a second byte of data to be received whilst the first is still being read. Bear in mind that the first byte of received data must be read out before reception of the subsequent byte has completed, otherwise the first byte of data will be lost (overwritten with the second byte).

Baud Rate Generator Reload Register (SREL

The SRL0 includes a dedicated internal Baud Rate Generator which can be used for baud rate generation in Modes 1 and 3. The baud rate is generated based on the overflow rate of a 10-bit Baud Timer register – internal to the Generator Unit.

The Baud Timer register is incremented after every 2 cycles of the external system clock (CLK) signal. The required clock signal for the timer is obtained from the Internal Clock Generation Unit (providing an output clock signal to the timer which is CLK/2).

When the Baud Timer register reaches overflow, it is automatically loaded with the value stored in the SREL register.

The SREL register is a 16-bit register, which is further sub-divided into two 8-bit registers – SRELL (low 8 bits) and SRELH (high 8 bits). These two sub-registers are used to provide the 10-bit value for SREL as follows:

SREL9..8 ← SRELH1..0

SREL7..0 ← SRELL7..0

Timer Control Register (TCON)

Table 19. The TCON register

MSB							LSB
F11	TR	F10	F9	F8	F7	F6	F5

Table 20. The TCON register bit functions

Bit	Symbol	Function
TCON.7	F11	General purpose Flag 11 available for user
TCON.6	TR	Timer Run control bit. If cleared, Timer Unit is stopped.
TCON.5	F10	General purpose Flag 10 available for user
TCON.4	F9	General purpose Flag 9 available for user
TCON.3	F8	General purpose Flag 8 available for user
TCON.2	F7	General purpose Flag 7 available for user
TCON.1	F6	General purpose Flag 6 available for user
TCON.0	F5	General purpose Flag 5 available for user

Timer Register (T)

The SRL0 includes an additional Timer Unit which can be used for baud rate generation in Modes 1 and 3. The unit has a 16-bit Timer register, which is further sub-divided into two 8-bit registers – TL (low 8 bits) and TH (high 8 bits).

The Timer is hard-coded to operate as an 8-bit auto-reload timer. The reload value is stored in the TH register.

The TL register is incremented after every 12 cycles of the external system clock (CLK) signal. The required clock signal for the timer is obtained from the Internal Clock Generation Unit (providing an output clock signal to the timer which is CLK/12).

When the TL register reaches overflow, the value stored in the TH register is automatically copied to TL.

Operation of the Timer is controlled by the TR bit in the TCON register. If TR is '1', the Timer is running. If TR is '0' the Timer stops.

A/D Converter Control Register (ADCON)

Table 21. The ADCON register

MSB							LSB
BD	-	-	-	-	-	-	-

Table 22. The ADCON register bit functions

Bit	Symbol	Function
ADCON.7	BD	Used to determine the source for baud rate generation when the serial interface is operating in Modes 1 and 3. 0 = Timer Unit 1 = Internal Baud rate Generator (using SREL register)
ADCON.6	-	Not Used
ADCON.5	-	Not Used
ADCON.4	-	Not Used
ADCON.3	-	Not Used
ADCON.2	-	Not Used
ADCON.1	-	Not Used
ADCON.0	-	Not Used

Register Reset Values

Table 23 shows the values contained in each of the SRL0's internal registers after an external system reset has been received on the RST input.

Table 23. Register reset values

Register	Value after reset
PCON	00000000
SCON	00000000
SBUF	00000000
SRELL	11011001
SRELH	00000011
TCON	00000000
TL	00000000
TH	00000000
ADCON	00000000

Serial Interface Operation

The Serial Interface can operate in the following four modes:

Mode 0

Pin RXD serves as serial input, with RXDO as serial output. TXD outputs the shift clock.

8 bits are transmitted/received with LSB first.

In Mode 0, the baud rate is fixed at 1/12 of the crystal frequency.

Mode 1

Pin RXD serves as serial input and TXD serves as serial output. No external shift clock is used.

10 bits are transmitted/received: a start bit (always 0), 8 data bits (LSB first) and a stop bit (always 1).

On reception, the start bit synchronizes the transmission, 8 data bits are made available by reading the SBUF register and the stop bit sets flag RB8 in the SCON register.

In mode 1, the baud rate is variable and can be specified by either the internal baud rate generator or the additional Timer Unit.

Mode 2

This mode is similar to Mode 1, with two differences:

- The baud rate is fixed at 1/32 or 1/64 of oscillator frequency and
- 11 bits are transmitted/received: a start bit (0), 8 data bits (LSB first), a programmable 9th bit, and a stop bit (1).

At transmission, bit TB8 in the SCON register is output as the 9th bit. This bit could be used to provide parity checking, by moving the parity bit (P in the PSW register) into TB8. On reception of serial data, the 9th bit is moved into bit RB8 in the SCON register and the stop bit is ignored.

Mode 3

The only difference between Mode 2 and Mode 3 is that the baud rate is variable in Mode 3. Either the internal baud rate generator or the additional Timer Unit can be used to specify the baud rate.

In all four operational modes, any instruction that has the SBUF register as its destination will initiate a serial transmission of data. Initiating the reception of serial data depends on the particular operating mode. In Mode 0, reception is initiated by setting the appropriate flags in the SCON register as follows:

- RI = 0
- REN = 1

In Modes 1, 2 and 3, a start bit when REN = 1 starts the reception of serial data.

Multiprocessor Communications

When the serial interface is operating in Modes 2 or 3, the programmable 9th data bit can be utilized to enable multiprocessor communications.

When serial data is received in either of these modes, the 9th data bit is loaded into bit 2 (RB8) of the SCON register. The next bit received in a serial reception is the stop bit. The serial interface can be programmed to generate an interrupt when this stop bit is received – by setting the receive interrupt flag (RI) in the SCON register - ONLY if RB8 is a '1'. This feature is enabled only when bit 5 (SM2) of the SCON register is set.

By using the state of the 9th data bit to distinguish between address ('1') and data ('0') bytes sent from a microprocessor, this feature can be used to establish efficient multiprocessor communications in the following manner:

- Each Slave processor has bit SM2 in its SCON register set to 1.
- The Master processor sends, via the serial interface, an address byte. This address byte identifies which Slave processor the Master wishes to communicate with. The 9th bit in this serial transmission is set.
- All Slave microprocessors connected to the Master will be interrupted, enabling each to determine whether or not it is the Slave that the Master wishes to communicate with.
- The addressed Slave processor will clear its SM2 bit and prepare to receive the serial transmission from the Master. All other Slave processors leave the SM2 bit set.
- The Master processor then sends its data, with the 9th bit cleared. The addressed Slave receives the transmission of data. As the 9th bit is '0' for data transmission, RB8 will be '0' in all other Slaves and no interrupts will be generated. These Slaves carry on with what they were previously doing.
- After the serial data from the Master has been completely received, the addressed Slave sets the SM2 bit back to '1'.

Baud Rate Generation

The following sections detail the methods and/or formulas involved in generating the baud rate for each of the SRL0's operational modes.

Mode 0

When the serial interface is operating in Mode 0, the baud rate is fixed and is defined as follows:

$$\text{baud rate} = \frac{F_{CLK}}{12}$$

Mode 2

When the serial interface is operating in Mode 2, the baud rate depends on the state of bit 7 (SMOD) in the PCON register and is defined as follows:

$$\text{baud rate} = \frac{2^{SMOD}}{64} \times F_{CLK}$$

Therefore, if SMOD = 0, the baud rate will be 1/64 of the frequency of the clock input to the device. If SMOD = 1, the baud rate will be 1/32 of this frequency.

Modes 1 and 3

When the serial interface is operating in either Mode 1 or Mode 3, the baud rate is variable and can be generated using either the dedicated internal baud rate generator or the Timer Unit. In both cases, bit 7 (SMOD) of the PCON register can be used to provide doubling of the baud rate (SMOD = 1).

Using the Internal Baud Rate Generator

To use the internal baud rate generator, bit 7 (BD) of the ADCON register must be set. In this case, the Baud Rate Generator Reload register (SREL) is used in providing the required prescaling.

The baud rate is generated as follows:

$$\text{baud rate} = \frac{2^{SMOD}}{64 \times (2^{10} - \text{SREL})} \times F_{CLK}$$

Note that the 10-bit value for SREL comes from the two, 8-bit registers SRELL and SRELH. The full 8 bits of SRELL are used (SREL7..0), while only the lowest two bits of SRELH are used (SREL9..8).

Using the Timer Unit

To use the Timer Unit to generate the baud rate, bit 7 (BD) of the ADCON register must be cleared.

The Timer is hard-coded to operate as a 'timer' in auto-reload mode. When the TL register reaches the overflow state, it is automatically loaded with the value stored in the TH register. It is the value in the TH register that is used in determining the required prescaling.

The baud rate is generated as follows:

$$\text{baud rate} = \frac{2^{SMOD}}{32 \times 12 \times (256 - \text{TH})} \times F_{CLK}$$

Serial Interface Timing

Serial Transmission

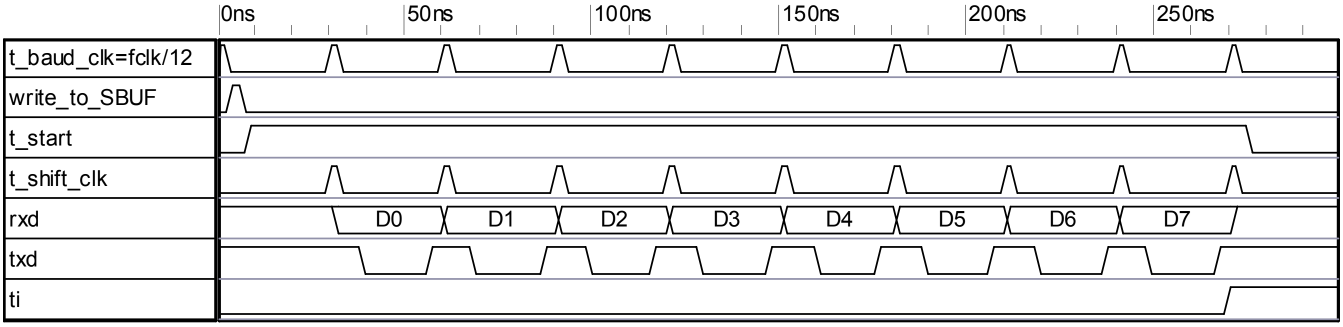

Figure 11. Mode 0 transmission

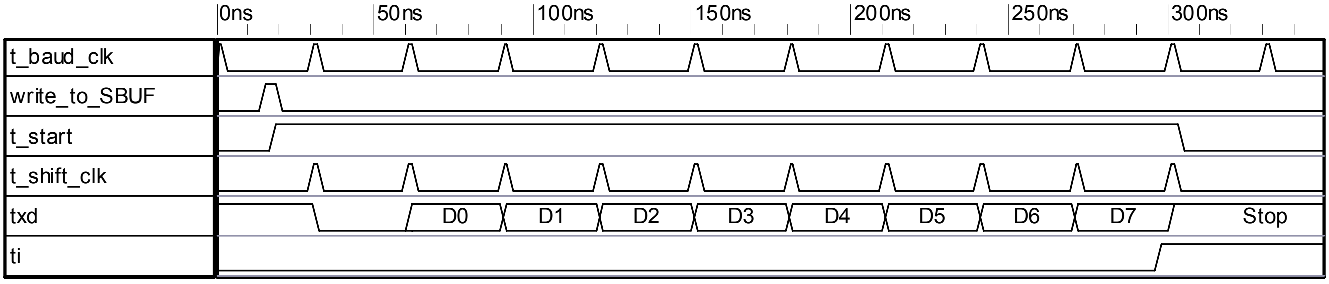

Figure 12. Mode 1 transmission

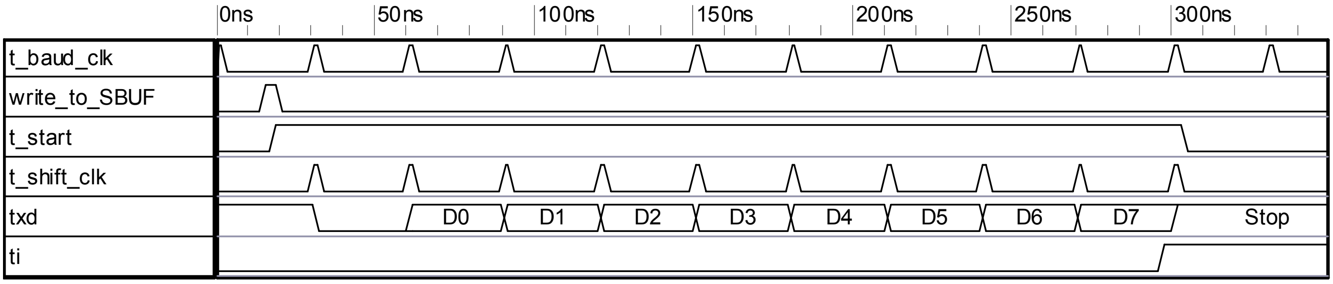

Figure 13. Modes 2 and 3 transmission

Serial Reception

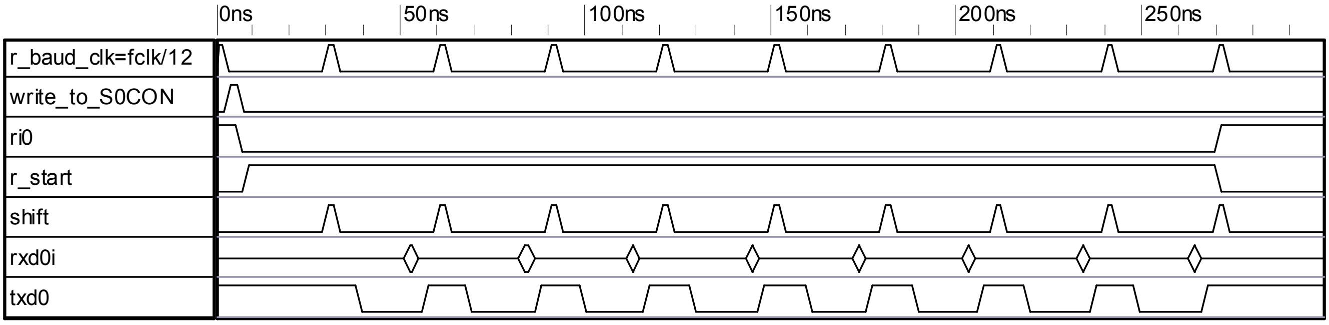

Figure 14. Mode 0 reception

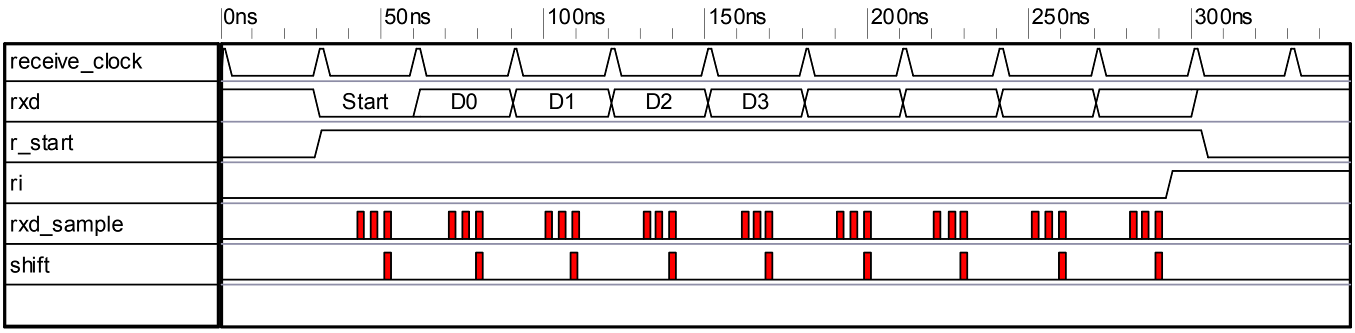

Figure 15. Mode 1 reception

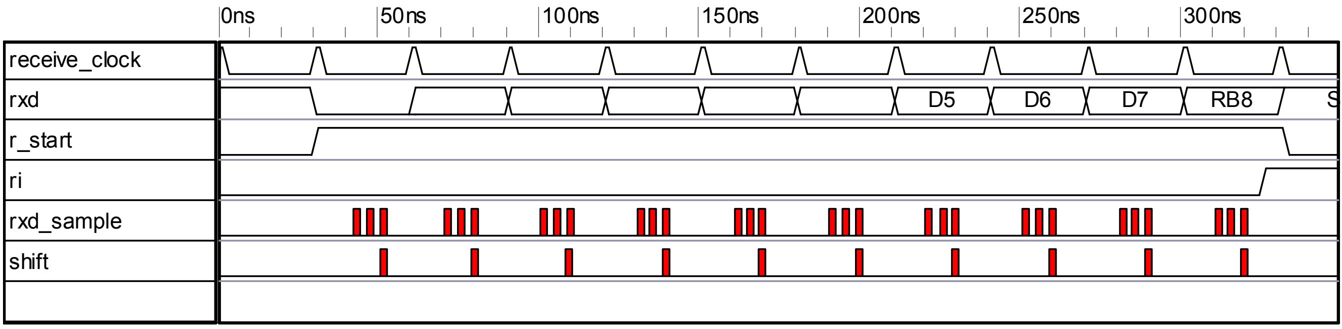

Figure 16. Modes 2 and 3 reception

Reading/Writing the Internal Registers

The host microcontroller can read/write any of the SRL0's internal Special Function Registers. Selection of a particular register is achieved by supplying the unique, 4-bit binary ID address code of the register. This code is sent to the SRL0 and appears at the ADDR input. Table 24 shows the unique address IDs associated with each of the addressable registers.

Table 24. Internal register unique address IDs

Register	Unique Register Address ID
PCON	0000
SCON	0001
SBUF	0010
SRELL	0011
SRELH	0100
TCON	0101
TL	0110
TH	0111
ADCON	1000

Writing to an Internal Register

The write operation occurs on the rising edge of the CLK input and can be summarized as follows:

- The host writes the unique 4-bit address ID for the register to be written (see Table 24) to the ADDR input of the SRL0
- The host sends a valid byte of data to the SRL0's DATAI input
- The host then sets the SRL0's WR signal High, to specify a Write cycle
- The SRL0 latches the byte of data into the target register.

Reading from an Internal Register

The read operation can be summarized as follows:

- The host writes the unique 4-bit address ID for the internal register to be read (see Table 24) to the ADDR input of the SRL0
- The host then sets the SRL0's RD signal High, to specify a Read cycle
- The SRL0 presents the byte of data from the chosen register at the DATAO output.

Revision History

Date	Version No.	Revision
07-Jan-2004	1.0	New product release
21-Sep-2004	1.1	Addition of Wishbone version of the device (SRL0_W)
01-Dec-2004	1.2	Schematic symbol updates
08-Feb-2005	1.3	Changed polarity of INT signal for SRL0 from Low to High.
26-May-2005	1.4	Updated for Altium Designer SP4
12-Dec-2005	1.5	Path references updated for Altium Designer 6
13-Mar-2008	2.0	Updated for Altium Designer Summer 08