

I2S Transceiver (VHDL)

- [Code Download](#)
- [Features](#)
- [Introduction](#)
- [Background](#)
- [Theory of Operation](#)
- [Configuring the I2S Transceiver](#)
 - [sclk_ws_ratio](#)
 - [mclk_sclk_ratio](#)
 - [d_width](#)
- [Providing the Master Clock](#)
- [Port Descriptions](#)
- [Streaming Audio Data Transactions](#)
- [Reset](#)
- [Conclusion](#)
- [Additional Information](#)
- [Contact](#)

Code Download

Version 1.0: [i2s_transceiver.vhd](#)

Initial Public Release

Features

- VHDL source code of an Inter-IC Sound (I2S) transceiver component
- Meets the I2S Bus Specification
- Configurable sample rate
- Configurable serial bit clock rate
- Configurable data width
- Master only
- Complete example design available (see [Additional Information](#))

Introduction

This details a master I2S transceiver component for use in FPGAs, written in VHDL. The component receives audio data over an I2S interface and presents this received data to user logic on a parallel interface at the sample rate. It also inputs parallel data from user logic at the sample rate, and transmits this data out over an I2S interface. Figure 1 illustrates a typical example of the I2S transceiver integrated into a system.

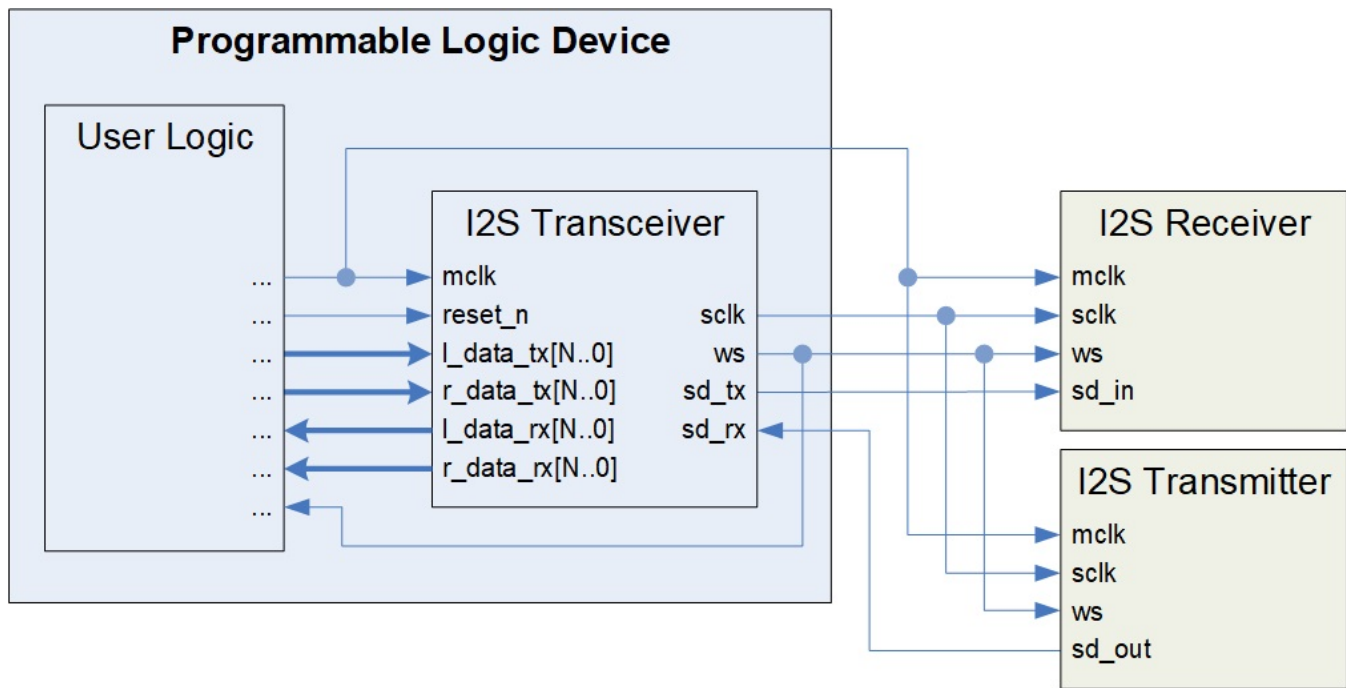


Figure 1. Example Implementation

Background

I2S (also I²S or IIS) is a standardized serial communication bus for digital audio. The bus consists of 3 lines: serial clock (SCK), word select (WS), and serial data (SD). Figure 2 shows the data framing for an I2S communication link.

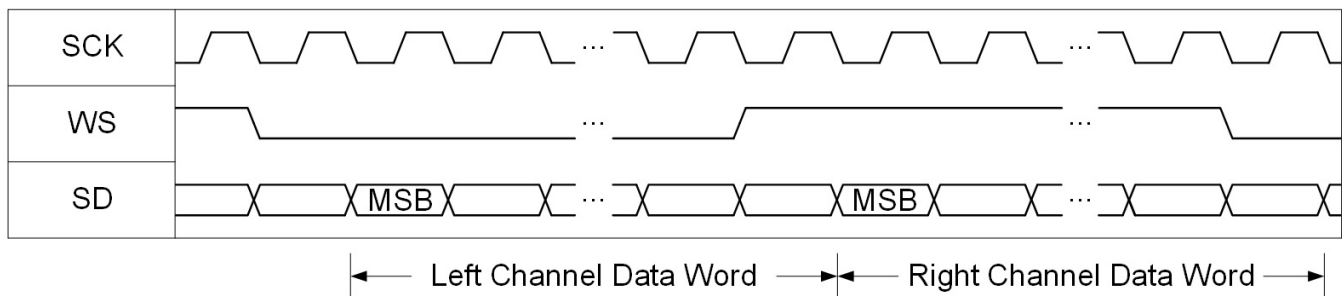


Figure 2. I2S Data Framing

The serial data line transmits two time-multiplexed data channels - the left and the right channel. Data is transmitted most significant bit (MSB) first in a 2's complement format. The word select line indicates which channel is presently being transmitted (low = left channel, high = right channel). That channel data word's MSB gets transmitted one clock cycle after the word select line toggles, followed by the rest of the data word. Data bits are transmitted on falling clock edges and read on rising clock edges.

Either the I2S transmitter or the I2S receiver can act as the master. The master provides the serial clock and word select signals.

Audio codecs also typically require a "master clock" (MCLK) to run their internal circuitry. The MCLK frequency is commonly a multiple of the sample rate, such as $256 \cdot F_s$ (where F_s is the sample rate). Requirements depend on the specific audio codec.

Theory of Operation

This I2S Transceiver inputs a master clock and derives the serial clock and word select signals from it using counters. The serial clock frequency is an integer multiple of the master clock, and the word select frequency (i.e. the sample rate) is an integer multiple of the serial clock.

The receive data is clocked into internal receive shift registers from the *sd_rx* port on serial clock rising edges. Shifting is enabled for the appropriate channel one serial clock cycle after the word select toggle and continues for the width of the data word, before being again disabled. The transceiver presents receive data for both right and left channels on the *r_data_rx* and *l_data_rx* ports, respectively, on each word select toggle.

Conversely, the right and left data presented on the *r_data_tx* and *l_data_tx* input ports is parallel loaded into internal transmit shift registers on each word select toggle. This data is clocked out the *sd_tx* port on serial clock falling edges. Shifting is enabled for the appropriate channel one serial clock cycle after the word select toggle and continues for one serial clock beyond the width of the data word, before being again disabled. The extra transmission bit sets the *sd_tx* data line to '0' in case there is a data width mismatch with the receiving I2S device, as per the I2S Bus Specification.

Configuring the I2S Transceiver

The I2S Transceiver is configured by setting the generic parameters in the ENTITY.

sclk_ws_ratio

The word select (*ws*) signal's frequency is the sample rate. The *sclk_ws_ratio* parameter defines the number of serial clock (*sclk*) cycles per word select period.

$$(1) \text{ sclk_ws_ratio} = \frac{f_{sclk}}{f_{sample}}$$

mclk_sclk_ratio

The *mclk_sclk_ratio* parameter defines the number of master clock (*mclk*) cycles per serial clock (*sclk*) cycle.

$$(2) \text{ mclk_sclk_ratio} = \frac{f_{mclk}}{f_{sclk}}$$

d_width

The parameter *d_width* defines the size of each data word in bits.

This does not need to equal the number of serial clock cycles during a word select half period (i.e. the number of serial clocks during a channel data transmission). If there are excess clock cycles during the word select half period, then any data on the serial data receive port (*sd_rx*) during the extra serial clock cycles is ignored, and the serial data transmit port (*sd_tx*) outputs '0' for these extra bits.

Providing the Master Clock

The I2S Transceiver requires a master clock *mclk* input to operate. This clock is intended to be the same master clock signal sent to the I2S slave device that the I2S Transceiver is communicating with.

Typically, the desired sample rate (*F_s*) is known, and the I2S device requires a master clock of some multiple of this sample rate. For instance, suppose the desired sample rate is the commonly used 44.1 kHz, and the I2S device accepts a master clock of 256**F_s*. This means the user needs to provide an 11.29 MHz *mclk* to both the I2S device and the I2S Transceiver.

The *mclk* signal can be derived inside the FPGA using a PLL.

Port Descriptions

Table 2 describes the I2S Transceiver's ports.

Table 2. Port Descriptions

Port	Width	Mode	Data Type	Interface	Description
reset_n	1	in	standard logic	user logic	Asynchronous active low reset
mclk	1	in	standard logic	user logic	Master clock from which sclk and ws are derived
sclk	1	out	standard logic	I2S device	Serial clock (i.e. bit clock)
ws	1	out	standard logic	I2S device	Word select (i.e. left right clk)
sd_tx	1	out	standard logic	I2S device	Serial data transmit
sd_rx	1	in	standard logic	I2S device	Serial data receive
l_data_tx	N*	in	standard logic vector	user logic	Left channel data word to transmit

r_data_tx	N*	in	standard logic vector	user logic	Right channel data word to transmit
l_data_rx	N*	out	standard logic vector	user logic	Left channel data word received
r_data_rx	N*	out	standard logic vector	user logic	Right channel data word received
Notes * N is the specified data width, set by the d_width generic					

Streaming Audio Data Transactions

The I2S Transceiver makes received data available to the user logic on the *l_data_rx* and *r_data_rx* ports. Both of these receive data ports update on each *ws* transition. Since new left channel data is only received while *ws* = '0', the *l_data_rx* port gets new values concurrent with the rising edge of *ws*, i.e. when the left channel data word finishes. Likewise, new right channel data is only received while *ws* = '1', so the *r_data_rx* port gets new values concurrent with the falling edge of *ws*, i.e. when the right channel data word finishes. The user logic can use the *ws* signal to indicate that new data is available and thereby retrieve this data at the sample rate. Figure 3 illustrates this behavior.

Similarly, the I2S Transceiver latches in new transmit data on the *l_data_tx* and *r_data_tx* ports on each *ws* transition. The user logic can use the *ws* signal to indicate when its outgoing data is latched in, and it is therefore free to present the next set of transmit data on these ports. The *l_data_tx* values latched in on the falling edge of *ws* are transmitted on *sd_tx* during that left channel data word transmission. Likewise, the *r_data_tx* values latched in on the rising edge of *ws* are transmitted on *sd_tx* during that right channel data word transmission. Figure 3 shows this behavior.

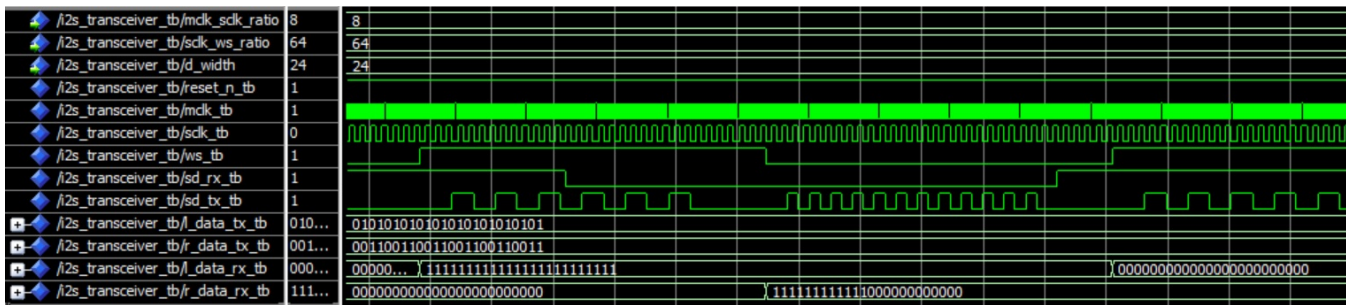


Figure 3. Timing Diagram

Reset

The *reset_n* input port must have a logic high for the I2S Transceiver to operate. A low signal on this port asynchronously resets the component. During reset, the component holds the *sclk*, *ws*, and *sd_tx* ports low. Any transmit currently underway is discontinued. Any receive currently in progress is abandoned, and the *l_data_rx* and *r_data_rx* output ports clear. All internal data buffers are cleared. Once released from reset, the I2S Transceiver resumes operation.

Conclusion

This I2S Transceiver is a configurable programmable logic component that transmits and receives audio data over a standard I2S bus. It operates as a master, providing the serial clock and word select signals for the interface. It allows the user to configure it for a particular sampling frequency, data width, master clock, and number of serial clock cycles per word.

Additional Information

[I2S Bus Specification, Philips Semiconductors](#)

[I2S Pmod Quick Start \(VHDL\)](#) - This design uses the I2S Transceiver component described on this page to interface an FPGA to a Cirrus Logic CS5343 Multi-Bit Audio ADC and also a Cirrus Logic CS4344 Stereo DAC.

Contact

Comments, feedback, and questions can be sent to eeewiki@digikikey.com.