Configurable Kilo-Channel I^{2}S/TDM Transceiver

User Guide

October 5, 2020

Contents

List of Tables

List of Figures

1 Introduction

The IObundle I²S/TDM Transceiver core is a configurable audio interface core with transmission and receiving capabilities. It supports master/slave, internal DMA, multi-channel and several sample sizes and frame formats. The IP is currently supported for use in ASICs and FPGAs.

2 Symbol

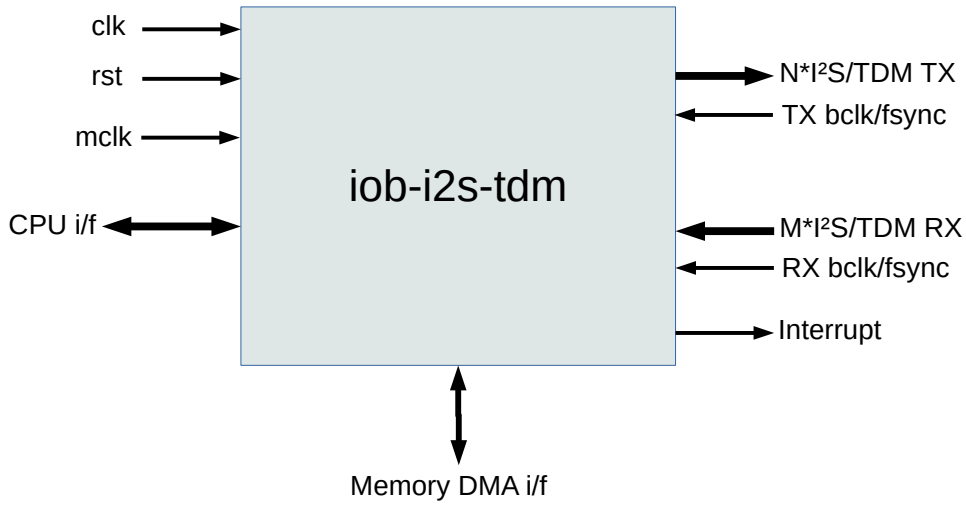

Figure 1: IP Core Symbol

3 Features

- Master/slave operation.
- Full duplex operation for maximum 10 receive pins and 10 transmit pin
- Maximum of 2560 receive channels and 2560 transmit channels
- Internal Direct Memory Access (DMA)
- Interrupt pin triggered automatically on FIFO levels
- Software example driver in C
- Configurable sample rate, channel size, sample size, and data delay.
- Double clock domain design: system and audio master clock
- Comprehensive configuration, control and status register file
- Native or AXI4 Lite interface with CPU, and AXI4 interface to user memory controller

4 Benefits

- Compact hardware implementation
- Can fit many instances in low cost FPGAs
- Can fit many instances in small ASICs
- Low power consumption

5 Deliverables

- FPGA netlist or source code (optional)
- Example testbench
- Example integration in a system on-chip with scripts for FPGA and ASIC implementation
- User documentation for easy system integration

6 Block Diagram and Description

A high-level block diagram of the hardware constituting the I²S/TDM transceiver core is presented in Figure 6 and a brief explanation of each block is given in Table 1.

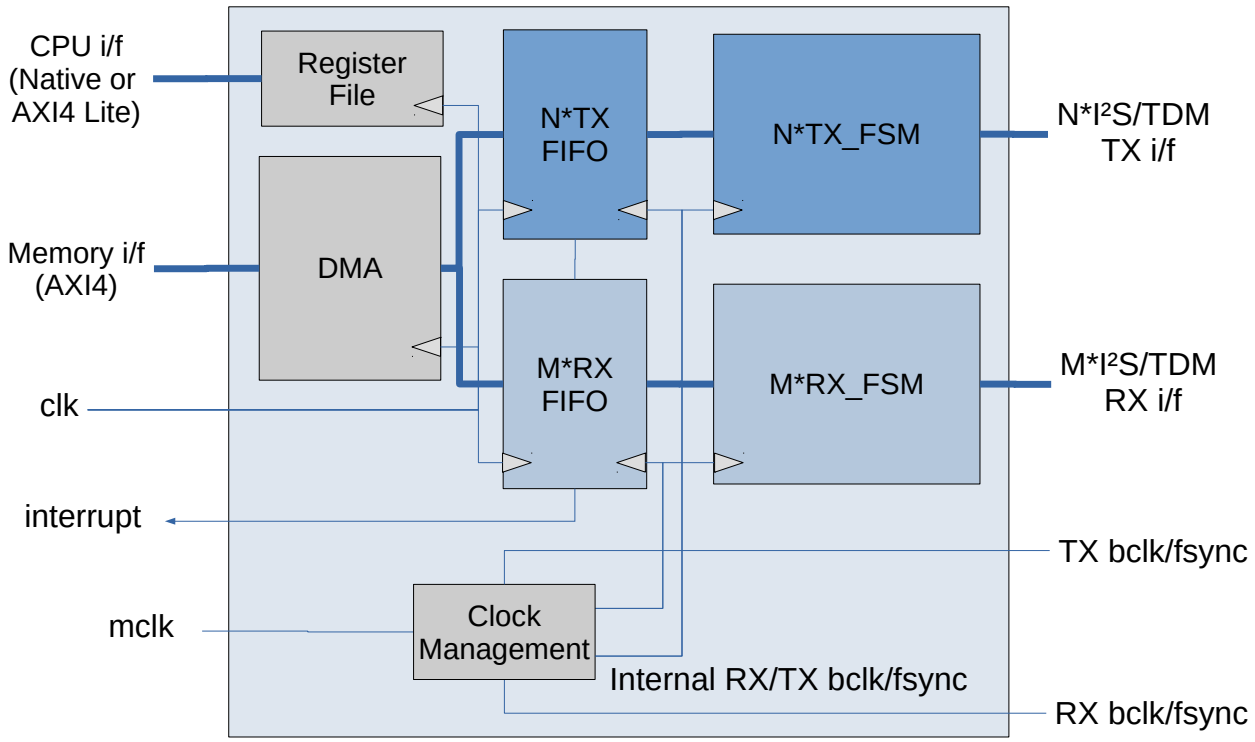

Figure 2: High-level block diagram

Block	Description
Register File	Holds the current configuration of the I²S channel for each module (TX and RX) as well as internal parameters. Data to be sent or that has been received is stored here temporarily.
DMA	Receives requests from the CPU interface (address, direction and data length). This block transfers from memory to the TX FIFO and from the RX FIFO to the memory.
Transmitter	Reads from the TX FIFO buffers, serializes the outbound data and drives the corresponding serial data line (sdata). Drives the word selection signal (fsync). There is one transmitter per data pin. An asynchronous FIFO stores the data to be transmitted through the I²S/TDM pin. Its write clock is the system clock (clk); its read clock is the serial bit clock (bclk).
Receiver	Deserializes the inbound data read from the corresponding serial data line (sdata). Writes the received data to the RX FIFO. There is one receiver per data pin. An asynchronous FIFO stores the data received from the I²S/TDM pins. Its write clock is the bit clock (bclk); its read clock is the system clock (clk).
Clock Manager	Generates and drives the bit clock (bclk) and word clock (fsync) signals in master mode, and routes external bclk and fsync signals in slave mode. Uses user-configurable dividers in master mode to generate bclk and fsync and tri-state buffers to drive the respective pins.

Table 1: Block descriptions.

7 Synthesis Parameters

The I²S/TDM transceiver core can be configured pre-synthesis to match the intended application. The synthesis parameters are presented in Table 2.

Parameter	Default Value	Description
N_RXDATA_PINS	10	Number of RX serial data pins
N_TXDATA_PINS	10	Number of TX serial data pins
DMA_ADDR_W	30	Addressable memory space (log2)
DMA_DATA_W	128	Number of bits of each word transferred by DMA. Possible values: 32, 64, 128
TXFIFO_ADDR_W	10	Transmit FIFO size (log2)
RXFIFO_ADDR_W	10	Receive FIFO size (log2)

Table 2: Synthesis Parameters

8 Interface Signals

The interface signals of the I²S/TDM transceiver core are described in the following tables.

Name	Direction	Width	Description
clk	input	1	System clock input
rst	input	1	System reset asynchronous and active high
mclk	input	1	Audio master clock its frequency is divided by the values in the TXCLK_DIV and RXCLK_DIV registers to obtain the transmit and receive bit clocks respectively
interrupt	output	1	Interrupt pin goes high if the RX FIFO goes above the threshhold level or TX FIFO goes below threshhold level remains low otherwise

Table 3: General Interface Signals

Name	Direction	Width	Description
valid	input	1	Native CPU interface valid signal
address	input	6	Native CPU interface address signal
wdata	input	32	Native CPU interface data write signal
wstrb	input	1	Native CPU interface write strobe signal
rdata	output	32	Native CPU interface read data signal
ready	output	1	Native CPU interface ready signal

Table 4: CPU Native Slave Interface Signals

Name	Direction	Width	Description
s_axil_awaddr	input	ADDR_W	Address write channel address
s_axil_awcache	input	4	Address write channel memory type. Transactions set with Normal Non-cacheable Modifiable and Bufferable (0011).
s_axil_awprot	input	3	Address write channel protection type. Transactions set with Normal Secure and Data attributes (000).
s_axil_awvalid	input	1	Address write channel valid
s_axil_awready	output	1	Address write channel ready
s_axil_wdata	input	DATA_W	Write channel data
s_axil_wstrb	input	DATA_W/8	Write channel write strobe
s_axil_wvalid	input	1	Write channel valid
s_axil_wready	output	1	Write channel ready
s_axil_bresp	output	2	Write response channel response
s_axil_bvalid	output	1	Write response channel valid
s_axil_bready	input	1	Write response channel ready
s_axil_araddr	input	ADDR_W	Address read channel address
s_axil_arcache	input	4	Address read channel memory type. Transactions set with Normal Non-cacheable Modifiable and Bufferable (0011).
s_axil_arprot	input	3	Address read channel protection type. Transactions set with Normal Secure and Data attributes (000).
s_axil_arvalid	input	1	Address read channel valid
s_axil_arready	output	1	Address read channel ready
s_axil_rdata	output	DATA_W	Read channel data
s_axil_rresp	output	2	Read channel response
s_axil_rvalid	output	1	Read channel valid
s_axil_rready	input	1	Read channel ready

Table 5: CPU AXI4 Lite Slave Interface Signals

Name	Direction	Width	Description
m_axi_awid	output	1	Address write channel ID
m_axi_awaddr	output	AXI_ADDR_W	Address write channel address
m_axi_awlen	output	8	Address write channel burst length
m_axi_awsize	output	3	Address write channel burst size. This signal indicates the size of each transfer in the burst
m_axi_awburst	output	2	Address write channel burst type
m_axi_awlock	output	1	Address write channel lock type
m_axi_awcache	output	4	Address write channel memory type. Transactions set with Normal Non-cacheable Modifiable and Bufferable (0011).
m_axi_awprot	output	3	Address write channel protection type. Transactions set with Normal Secure and Data attributes (000).
m_axi_awqos	output	4	Address write channel quality of service
m_axi_awvalid	output	1	Address write channel valid
m_axi_awready	input	1	Address write channel ready
m_axi_wdata	output	AXI_DATA_W	Write channel data
m_axi_wstrb	output	AXI_DATA_W/8	Write channel write strobe
m_axi_wlast	output	1	Write channel last word flag
m_axi_wvalid	output	1	Write channel valid
m_axi_wready	input	1	Write channel ready
m_axi_bid	input	1	Write response channel ID
m_axi_bresp	input	2	Write response channel response
m_axi_bvalid	input	1	Write response channel valid
m_axi_bready	output	1	Write response channel ready
m_axi_arid	output	1	Address read channel id
m_axi_araddr	output	AXI_ADDR_W	Address read channel address
m_axi_arlen	output	8	Address read channel burst length
m_axi_arsize	output	3	Address read channel burst size. This signal indicates the size of each transfer in the burst
m_axi_arburst	output	2	Address read channel burst type
m_axi_arlock	output	1	Address read channel lock type
m_axi_arcache	output	4	Address read channel memory type. Transactions set with Normal Non-cacheable Modifiable and Bufferable (0011).
m_axi_arprot	output	3	Address read channel protection type. Transactions set with Normal Secure and Data attributes (000).
m_axi_arqos	output	4	Address read channel quality of service
m_axi_arvalid	output	1	Address read channel valid
m_axi_arready	input	1	Address read channel ready
m_axi_rid	input	1	Read channel ID
m_axi_rdata	input	AXI_DATA_W	Read channel data
m_axi_rresp	input	2	Read channel response
m_axi_rlast	input	1	Read channel last word
m_axi_rvalid	input	1	Read channel valid
m_axi_rready	output	1	Read channel ready

Table 6: CPU AXI4 Master Interface Signals

Name	Direction	Width	Description
bclk_rx	inout	1	RX bit clock bidirectional signal
fsync_rx	inout	1	RX frame sync bidirectional signal
sdata_rx	input	N_RXDATA_PINS	RX data serial input signal

Table 7: I²S/TDM Receiver Interface Signals

Name	Direction	Width	Description
bclk_tx	inout	1	TX bit clock bidirectional signal
fsync_tx	inout	1	TX frame sync bidirectional signal
sdata_tx	output	N_TXDATA_PINS	TX data serial output signal

Table 8: I²S/TDM Transmitter Interface Signals

9 Registers

The software accessible registers of the I²S/TDM core are described in Tables 9, 12. The table gives information on the name, read/write capability, word aligned addresses, used word bits and a textual description.

Name	R/W	Addr	Bits	Initial Value	Description
RXINTERRUPT_STATUS	R	0x60	0:0	0	Receiver interruption status.
RXEN	W	0x64	0:0	0	Enables the receiver.
RXRD	W	0x68	0:0	0	Reads a word of width DMA_DATA_W in bits from the selected FIFO and places it in the RX_DATA_FIFO registers as below as an alternative to DMA operation.
RXCLK_DIV	W	0x6c	11:0	128	Number of mclk periods per bit minus 1 maximum value is 4095.
RXSOFTRST	W	0x70	0:0	0	Receiver soft reset.
RXSAMPLE_W	W	0x74	4:0	24	Sets sample width in bits - 1.
RXN_CHANNELS	W	0x78	7:0	2	Number of audio channels minus one per serial data pin
RXFSYNC_W	W	0x7c	12:0	32	Sets the width of the word clock (fsync) pulse in bit clock cycles from the beginning of the first channel
RXDATADELAY	W	0x80	4:0	1	Sets the delay from fsync activation to first data bit in bit (bclk) periods
RXBCLK_POLARITY	W	0x84	0:0	0	Set to 0 to sample sdata at the positive edge set to 1 to sample sdata at the negative edge.
RXFSYNC_POLARITY	W	0x88	0:0	0	Should be set according to the polarity of fsync. Set to 0 for a positive pulse set to 1 for a negative pulse.
RXMS_SELECT	W	0x8c	0:0	0	Master (1) or slave (0) select.
RXINTERRUPT_EN	W	0x90	0:0	0	Enables (1) or disables (0) the interrupt pin.

Table 9: Software accessible registers for the RX block.

Name	R/W	Addr	Bits	Initial Value	Description
RXFIFO_SELECT	W	0x38	3:0	0	Selects the FIFO associated with the serial data pin number written to this register for CPU access.
RXFIFO_DATA_0	R	0x3c	DATA_W-1:0	0	Data output from FIFO for received data word 0.
RXFIFO_DATA_1	R	0x40	DATA_W-1:0	0	Data output from FIFO for received data word 1 if DMA_DATA_W = 64 or 128.
RXFIFO_DATA_2	R	0x44	DATA_W-1:0	0	Data output from FIFO for received data word 2 if DMA_DATA_W = 128.
RXFIFO_DATA_3	R	0x48	DATA_W-1:0	0	Data output from FIFO for received data word 3 if DMA_DATA_W = 128.
RXFIFO_EMPTY	R	0x4c	0:0	1	Reads 1 when FIFO is empty otherwise 0.
RXFIFO_FULL	R	0x50	0:0	0	Reads 1 when FIFO is full otherwise 0.
RXFIFO_LEVEL	R	0x54	RXFIFO_ADDR_W-1:0	0	Indicates the number of samples currently in the FIFO.
RXFIFO_FLUSH	W	0x58	0:0	0	Empties the FIFO.
RXFIFO_THRESH	W	0x5c	RXFIFO_ADDR_W-1:0	256	Level threshhold for triggering interrupt.

Table 10: Software accessible registers for the RX block (FIFO control).

Name	R/W	Addr	Bits	Initial Value	Description
RXDMA_ADDR	W	0xc8	AXI_ADDR_W-1:0	0	Memory address for RX FIFO to memory transfers.
RXDMA_LEN	W	0xcc	7:0	0	Burst length for RX FIFO to memory transfers.
RXDMA_RUN	W	0xd0	0:0	0	Starts the DMA for RX FIFO to memory transfers.
RXDMA_READY	R	0xd4	0:0	1	DMA is ready to start an RX RUN (1) or not (0).

Table 11: Software accessible registers for the RX block (DMA control).

Name	R/W	Addr	Bits	Initial Value	Description
TXINTERRUPT_STATUS	R	0x94	0:0	0	Transmitter interruption status.
TXEN	W	0x98	0:0	0	Enables the transmitter.
TXWR	W	0x9c	0:0	0	Writes a word of width DMA_DATA_W in bits to the selected TX_DATA_FIFO registers as below which is written to the FIFO input in the next system clock cycle as an alternative to DMA operation.
TXCLK_DIV	W	0xa0	11:0	128	Number of mclk periods per bit minus 1 maximum value is 4095.
TXSOFTRST	W	0xa4	0:0	0	Transmitter soft reset.
TXSAMPLE_W	W	0xa8	4:0	24	Sets sample width in bits - 1.
TXN_CHANNELS	W	0xac	7:0	2	Number of audio channels minus one per serial data pin.
TXFSYNC_W	W	0xb0	12:0	32	Sets the width of the word clock (fsync) pulse in bit clock cycles from the beginning of the first channel.
TXDATADELAY	W	0xb4	4:0	1	Sets the delay from fsync activation to first data bit in bit (bclk) periods.
TXBCLK_POLARITY	W	0xb8	0:0	1	Set to 0 to send bits aligned with the bclk positive edge set to 1 to send bits aligned with the bclk negative edge.
TXFSYNC_POLARITY	W	0xbc	0:0	0	Should be set according to the polarity of fsync. Set to 0 for a positive pulse set to 1 for a negative pulse.
TXMS_SELECT	W	0xc0	0:0	1	Master (1) or slave (0) select.
TXINTERRUPT_EN	W	0xc4	0:0	0	Enables (1) or disables (0) the interrupt pin.

Table 12: Software accessible registers for the TX block.

Name	R/W	Addr	Bits	Initial Value	Description
TXFIFO_SELECT	W	0x00	3:0	0	Selects the FIFO associated with the serial data pin number written to this register for CPU access.
TXFIFO_DATA_0	W	0x04	DATA_W-1:0	0	Data input to FIFO for transmission word 0.
TXFIFO_DATA_1	W	0x08	DATA_W-1:0	0	Data input to FIFO for transmission word 1 if DMA_DATA_W = 64 or 128.
TXFIFO_DATA_2	W	0x0c	DATA_W-1:0	0	Data input to FIFO for transmission word 2 if DMA_DATA_W = 128.
TXFIFO_DATA_3	W	0x10	DATA_W-1:0	0	Data input to FIFO for transmission word 3 if DMA_DATA_W = 128.
TXFIFO_EMPTY	R	0x14	0:0	1	Reads 1 when FIFO is empty otherwise 0.
TXFIFO_FULL	R	0x18	0:0	0	Reads 1 when FIFO is full otherwise 0.
TXFIFO_LEVEL	R	0x1c	TXFIFO_ADDR_W-1:0	0	Indicates the number of samples currently in the FIFO.
TXFIFO_FLUSH	W	0x20	0:0	0	Empties the FIFO.
TXFIFO_THRESH	W	0x24	TXFIFO_ADDR_W-1:0	128	Level threshhold for triggering interrupt.

Table 13: Software accessible registers for the TX block (FIFO control).

Name	R/W	Addr	Bits	Initial Value	Description
TXDMA_ADDR	W	0x28	AXI_ADDR_W-1:0	0	Memory address for memory to TX FIFO transfers.
TXDMA_LEN	W	0x2c	7:0	0	Burst length for TX FIFO to memory transfers.
TXDMA_RUN	W	0x30	0:0	0	Starts the DMA for TX FIFO to memory transfers.
TXDMA_READY	R	0x34	0:0	1	DMA is ready to start a RUN (1) or not (0).

Table 14: Software accessible registers for the TX block (DMA control).

10 Data Memory Organization

The organization of the audio samples for the various channels in the system's memory is depicted in Figure 3. Each data pin should correspond to a contiguous memory section. In each data section, the audio channels should be interleaved.

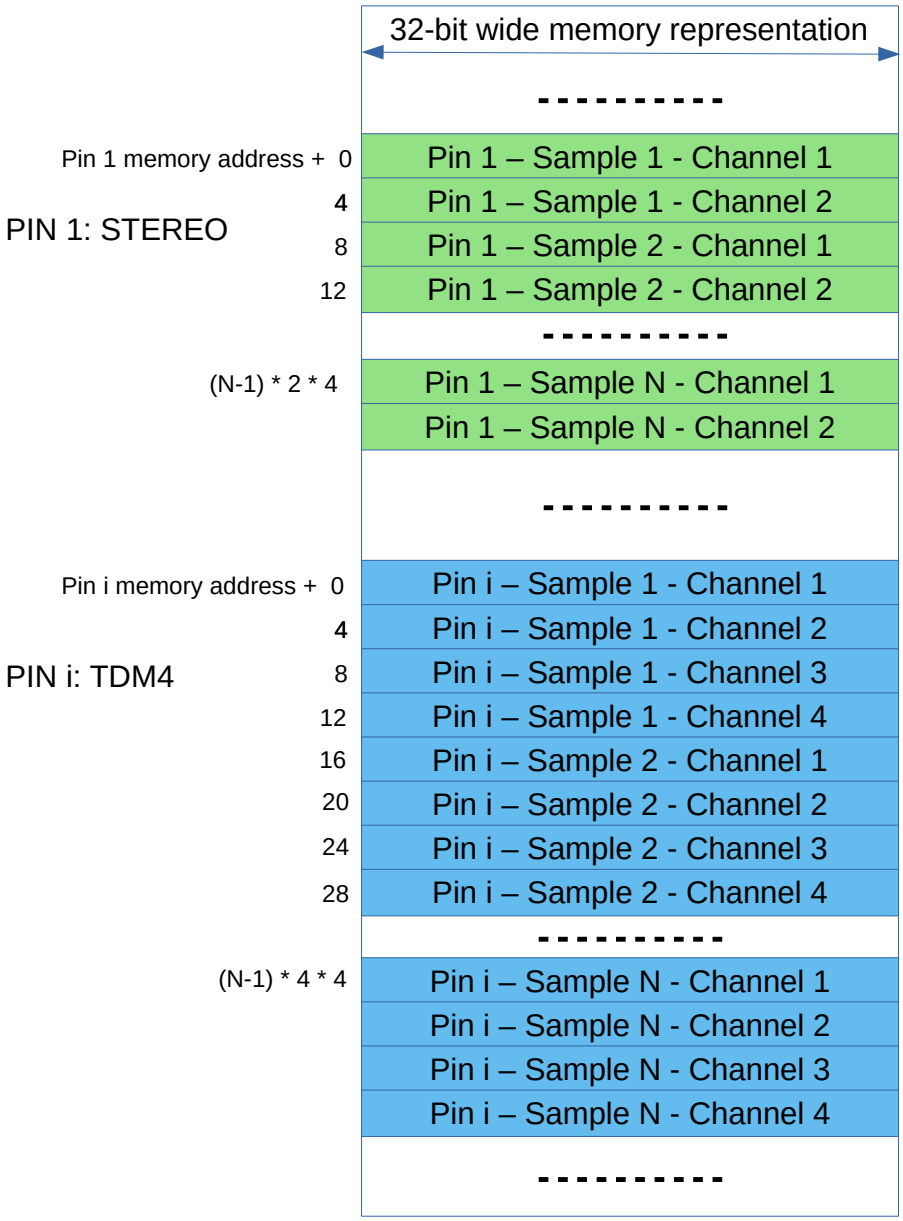

Figure 3: Data Memory Organisation

11 FPGA Results

The following are FPGA implementation results for two FPGA device families. BRAM usage results depend on the configured size of the RX and TX FIFOs, and can be changed. The below results use the configurations specified in Table 2, with the exception of the parameter DMA_DATA_W for Intel Cyclone V Devices, which has the value equal to 32.

Resource	Used
LUTs	2933
Registers	2646
DSPs	0
BRAM	40
PIN	0

Table 15: Implementation Resources for Xilinx Kintex Ultrascale Devices

Resource	Used
ALM	447,457
FF	659627
DSP	0
BRAM blocks	0
BRAM bits	
PIN	538

Table 16: Implementation Resources for Intel Cyclone V Devices

12 Timing Diagrams

The timing diagrams of the I²S/TDM transceiver are described in the following figures. Figure 4 uses a standard I2S configuration with 2 channels, the transmitter as master, receiver as slave, data delay of 1 clock cycle, serial data produced at the bit clock negative edge by the transmitter and consumed at the positive edge by the receiver, negative 8-period fsync pulse, and 8-bit samples. In each of the following figures, a single configuration parameter is modified to illustrate its effect. In all figures the highlighted edge of the bit clock signal corresponds to the sampling instants in the master core, transmitter or receiver. The slave core samples data at the opposite clock edge always.

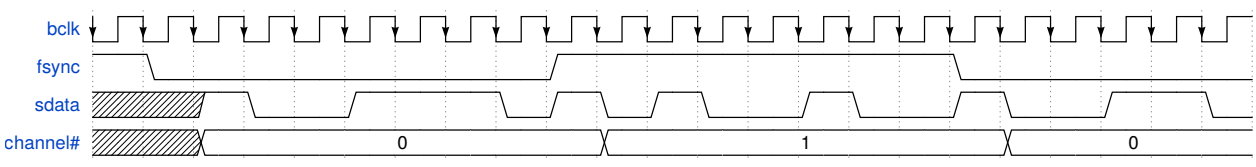

Figure 4: Timing diagram of the I²S/TDM transceiver, with a standard configuration.

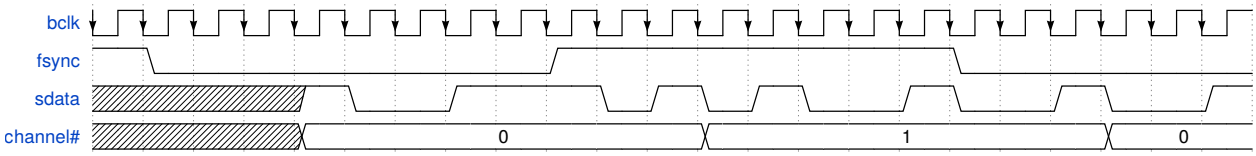

Figure 5: Timing diagram with data delay set to 3.

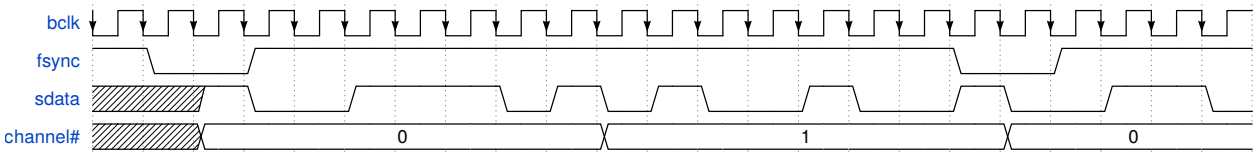

Figure 6: Timing diagram with fsync width set to 2.

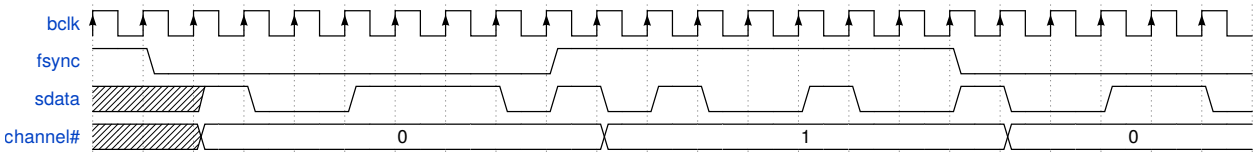

Figure 7: Timing diagram with inverted bclk.

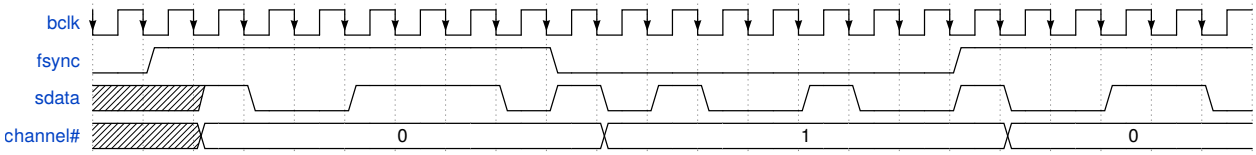

Figure 8: Timing diagram with inverted fsync.

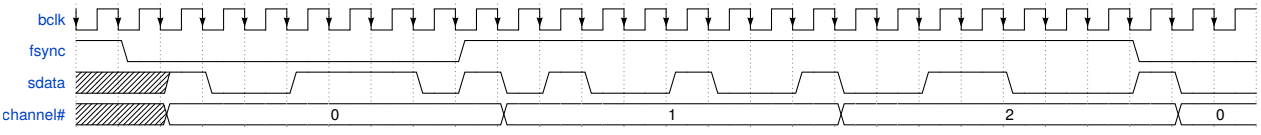

Figure 9: Timing diagram with 3 channels.

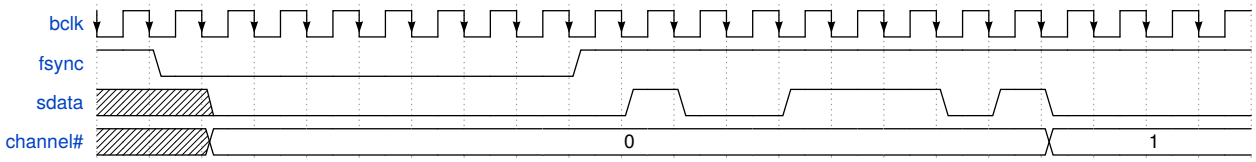

Figure 10: Timing diagram with sample width set to 16.

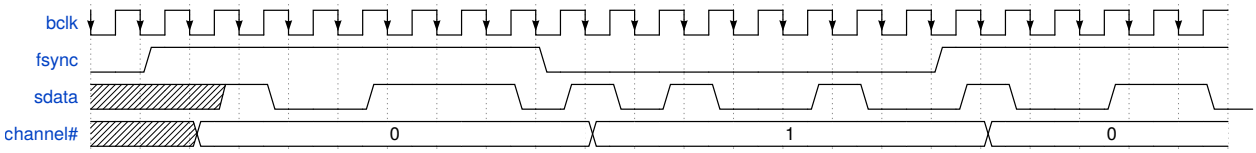

Figure 11: Timing diagram with receiver as master and the transmitter as slave