SMPTE 2022-5/6 Video over IP Receiver v4.0

LogiCORE IP Product Guide

Vivado Design Suite

PG033 October 1, 2014

XILINX
ALL PROGRAMMABLE™

Table of Contents

Appendix C: Debugging

Appendix D: Additional Resources and Legal Notices

Introduction

The Xilinx LogiCORE™ IP SMPTE 2022-5/6 Video over IP Receiver is a module for broadcast applications that requires bridging between SMPTE video connectivity standards bridging between uncompressed SMPTE video connectivity standards (SD/HD/3G-SDI) and 10Gb/s IP networks. The module is capable of recovering IP packets lost due to network transmission errors and ensure the picture quality of uncompressed, high bandwidth professional video is maintained. The core is for developing Internet protocol-based systems to reduce overall cost in broadcast facilities for distribution and routing of audio and video data.

Features

- Handle up to 8 channels of SD/HD/3G-SDI streams according to SMPTE 2022-6. Supports SD-SDI, HD-SDI, 3G-SDI Level-A, 3G-SDI Level-B single stream and 3G-SDI Level-B dual stream.
- Per stream basis Forward Error Correction (FEC) recovery in accordance to SMPTE 2022-5.
- Supports Level A and Level B FEC operations.
- Supports block aligned and non-block aligned FEC operations.
- Supports Virtual Local Area Network (VLAN) filtering.

LogiCORE IP Facts Table	
Core Specifics	
Supported Device Family(1)	Zynq®-7000, Virtex®-7, Kintex®-7
Supported User Interfaces	AXI4-Lite, AXI4-Stream, AXI4
Resources	See Table 2-1, Table 2-2, and Table 2-3
Provided with Core	
Design Files	Encrypted HDL
Example Design	*SMPTE 2022-5/6 High Bit Rate Media Transport Over IP Networks with Forward Error Correction* (XAPP1199) [Ref 1]
Test Bench	Verilog and VHDL
Constraints File	XDC
Simulation Model	Encrypted RTL, VHDL Behavioral, VHDL or Verilog source HDL
Supported S/W Driver	N/A
Tested Design Flows(2)	
Design Entry	Vivado® Design Suite
Simulation	For supported simulators, see the Xilinx Design Tools: Release Notes Guide.
Synthesis	Vivado Synthesis
Support	
Provided by Xilinx @ www.xilinx.com/support	

Notes:

1. For a complete list of supported devices, see the Vivado IP catalog.
2. For the supported versions of the tools, see the Xilinx Design Tools: Release Notes Guide.

Features (continued)

- Configurable channel filtering based on any combinations of the following:
  - IP source address
  - IP destination address
  - User Datagram Protocol (UDP) source port
  - UDP destination port
  - Real-time Transport Protocol (RTP) Synchronization Source (SSRC) identifier
  - VLAN tag value
- Seamless switching (SMPTE2022-7)
- RTP timestamp check bypass
- Statistic indicators
  - Received packet
  - Reordered packet, Duplicated packet count
  - Recovered packet count
  - Valid packet count, Unrecoverable packet count
  - Out of range packet count
  - Packet interval measure
  - Buffer overflow flag
  - Seamless protect flag
  - Link differential measure
- Include or remove FEC engine or secondary link during compile time
- AXI4-Stream data interfaces
- AXI4-Lite control interface

Overview

As broadcast and communications markets converge, and the use of IP networks for transport of video streams becomes more attractive to broadcasters and telecommunication companies alike, the adoption of 10 Gb/s Ethernet for the transmission of multiple uncompressed Serial Digital Interface (SDI) streams is becoming a major customer requirement. The industry is primarily looking at the SMPTE 2022 set of standards to create an open and interoperable way of connecting video over 10GbE equipment together and ensuring that Quality of Service (QoS) is high and packet loss is kept to a minimum or recovered through FEC. As shown in Figure 1-1, high bit rate SMPTE 2022-5/6 is aimed at contribution networks (for example, between broadcast center and regional studio).

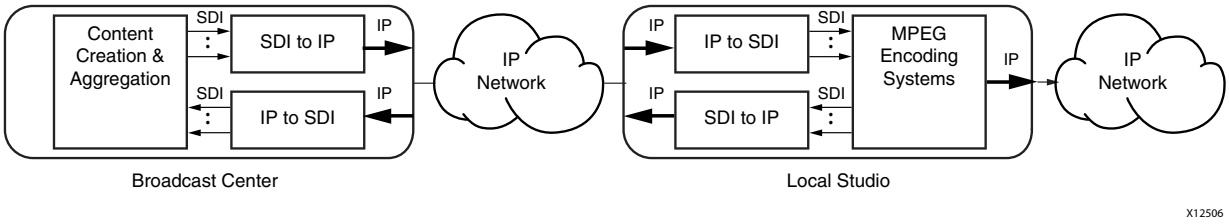

Figure 1-1: **High Bit Rate SMPTE 2022-5/6 between Broadcast Center and Local Studio**

The core includes Forward Error Correction (FEC). FEC protects the video stream during transport of high-quality video over IP networks. With FEC, the transmitter adds systematically generated redundant data to its video. This carefully designed redundancy allows the receiver to detect and correct a limited number of packet errors occurring anywhere in the video without the need to ask the transmitter for additional video data.

These errors, in the form of lost video packets, can be caused by many reasons, from thermal noise to storage system defects and transmission noise introduced by the environment. FEC gives the receiver the ability to correct these errors without needing a reverse channel to request retransmission of data. In real time systems, the latency is too great to request a retransmission. The ability of Xilinx FPGAs to bridge the broadcast and the communications industries by performing highly integrated real-time video interfaces help broadcasters reduce costs as well as reduce the overall time it takes to acquire, edit and produce content. Now that video can be reliably delivered over 10 Gb/s Ethernet (10GbE), broadcasters can replace some of the expensive mobile infrastructures supporting

outside live broadcasts, as well as enable remote production from existing fixed studio set ups, which dramatically reduces both capital expenditure and operating expenses.

Feature Summary

The core maps Ethernet packets into raw SD/HD/3G-SDI video streams and is capable of recovering IP packets lost to network transmission errors to ensure the highest picture quality of uncompressed, high bandwidth professional video.

The core support of VLAN comes from being able to operate seamlessly when receiving VLAN tagged Ethernet packets. You can configure and instantiate the core from the Vivado® design tools. Core functionality can be controlled dynamically through an AXI4-Lite interface.

Applications

- Transport uncompressed high bandwidth professional video streams over IP networks
- Support real-time audio/video applications such as contribution, primary distribution, and digital cinema

Licensing and Ordering Information

License Checkers

If the IP requires a license key, the key must be verified. The Vivado design tools have several license check points for gating licensed IP through the flow. If the license check succeeds, the IP can continue generation. Otherwise, generation halts with error. License checkpoints are enforced by the following Vivado flow:

Vivado Synthesis, Vivado Implementation, write_bitstream (Tcl Console command)

IMPORTANT: *IP license level is ignored at checkpoints. The test confirms a valid license exists. It does not check IP license level.*

If a Hardware Evaluation License is being used, the core will stop transmitting video after timeout.

License Type

This Xilinx LogiCORE™ IP module is provided under the terms of the Xilinx Core License Agreement. The module is shipped as part of the Vivado Design Suite.

IMPORTANT: *For full access to all core functionalities in simulation and in hardware, you must purchase a license for the core. Contact your local Xilinx sales representative for information about pricing and availability.*

For more information, visit the SMPTE 2022-5/6 Video Over IP product web page.

Information about other Xilinx LogiCORE IP modules is available at the Xilinx Intellectual Property page. For information on pricing and availability of other Xilinx LogiCORE IP modules and tools, contact your local Xilinx sales representative.

Product Specification

Standards

The core is compliant with the AXI4, AXI4-Stream and AXI4-Lite interconnect standards. See the *Video IP: AXI Feature Adoption* section of the *AXI Design Reference Guide* (UG761) [Ref 2] for additional information. The function of the core is compliant with SMPTE 2022-5/6 standard.

Maximum Frequencies

The maximum achievable clock frequency can vary. The maximum achievable clock frequency and all resource counts can be affected by other tool options, additional logic in the FPGA, using a different version of Xilinx tools and other factors. See the resource utilization tables for device family specific information.

Resource Utilization

Resources required for the this core have been estimated for the devices shown in Table 2-1, Table 2-2, and Table 2-3. These values were generated using the Vivado® Design Suite.

Table 2-1: **Resource Utilization for Zynq-7000 Devices (xc7z045, speed -1)**

SDI CHANNEL	FEC INCLUDE	FFs	LUTs	Slices	LUT FF Pairs	36k BLock RAMs	18k BLock RAMs	DSP48E1s
1	0	7,698	6,549	2,745	8,456	14	2	0
2	0	10,585	8,676	3,624	11,351	21	3	0
3	0	13,434	10,104	4,836	14,329	28	4	0
4	0	16,305	10,717	5,502	16,571	35	5	0
5	0	19,145	12,132	6,194	19,046	42	6	0
6	0	22,016	13,429	7,564	21,972	49	7	0
7	0	24,876	14,480	8,786	25,012	56	8	0
8	0	27,743	15,253	9,827	28,044	63	9	0
1	1	12,900	9,238	3,959	12,649	50	7	0
2	1	16,398	11,503	5,239	16,011	57	9	0
3	1	20,042	13,654	6,607	19,601	78	12	0
4	1	23,656	14,913	7,770	22,798	85	15	0
5	1	27,284	16,973	8,821	26,194	120	21	0
6	1	30,893	19,140	10,677	30,144	127	21	0
7	1	34,491	20,447	10,434	32,144	134	24	0
8	1	38,105	21,569	12,152	36,419	141	27	0

Table 2-2: **Resource Utilization for Virtex-7 FPGAs (xc7vx690t, Speed -1)**

SDI CHANNEL	FEC INCLUDE	FFs	LUTs	Slices	LUT FF Pairs	36k BLock RAMs	18k BLock RAMs	DSP48E1s
1	0	7,698	6,546	2,531	8,380	14	2	0
2	0	10,585	8,665	3,643	11,341	21	3	0
3	0	13,434	10,103	4,756	14,258	28	4	0
4	0	16,305	10,702	5,637	16,702	35	5	0
5	0	19,145	12,140	6,501	19,299	42	6	0
6	0	22,016	13,418	7,516	21,975	49	7	0

Table 2-2: **Resource Utilization for Virtex-7 FPGAs (xc7vx690t, Speed -1) *(Cont'd)***

SDI CHANNEL	FEC INCLUDE	FFs	LUTs	Slices	LUT FF Pairs	36k BLock RAMs	18k BLock RAMs	DSP48E1s
7	0	24,876	14,492	9,012	25,161	56	8	0
8	0	27,743	15,233	8,710	26,901	63	9	0
1	1	12,900	9,248	4,109	12,826	50	7	0
2	1	16,398	11,506	5,364	16,000	57	9	0
3	1	20,042	13,649	6,973	19,947	78	12	0
4	1	23,656	14,892	8,093	23,159	85	15	0
5	1	27,284	16,976	10,833	27,784	120	21	0
6	1	30,893	19,142	11,032	30,589	127	21	0
7	1	34,491	20,451	13,137	34,748	134	24	0
8	1	38,105	21,571	13,310	37,788	141	27	0

Table 2-3: **Resource Utilization for Kintex-7 FPGAs (xc7k325t, speed -1)**

SDI CHANNEL	FEC INCLUDE	FFs	LUTs	Slices	LUT FF Pairs	36k BLock RAMs	18k BLock RAMs	DSP48E1s
1	0	7,698	6,542	2,780	8,551	14	2	0
2	0	10,585	8,675	3,725	11,479	21	3	0
3	0	13,434	10,103	4,767	14,278	28	4	0
4	0	16,305	10,714	5,415	16,390	35	5	0
5	0	19,145	12,133	6,556	19,221	42	6	0
6	0	22,016	13,420	6,951	21,446	49	7	0
7	0	24,876	14,476	7,967	24,300	56	8	0
8	0	27,743	15,249	9,890	28,081	63	9	0
1	1	12,900	9,241	3,959	12,643	50	7	0
2	1	16,380	11,476	5,440	16,129	57	9	0
3	1	20,042	13,639	6,830	19,740	78	12	0
4	1	23,656	14,905	8,023	23,092	85	15	0
5	1	27,284	16,967	8,338	25,812	120	21	0
6	1	30,893	19,134	11,145	30,509	127	21	0
7	1	34,491	20,446	11,446	33,424	134	24	0
8	1	38,105	21,577	12,322	36,598	141	27	0

Port Descriptions

The core uses industry-standard control and data interfaces to connect to other system components. The following sections describe the various interfaces available with the core. Figure 2-2 shows an I/O Diagram of the core. The SDI_TX interface pins depend on the number of channels configured through the Vivado Integrated Design Environment (IDE).

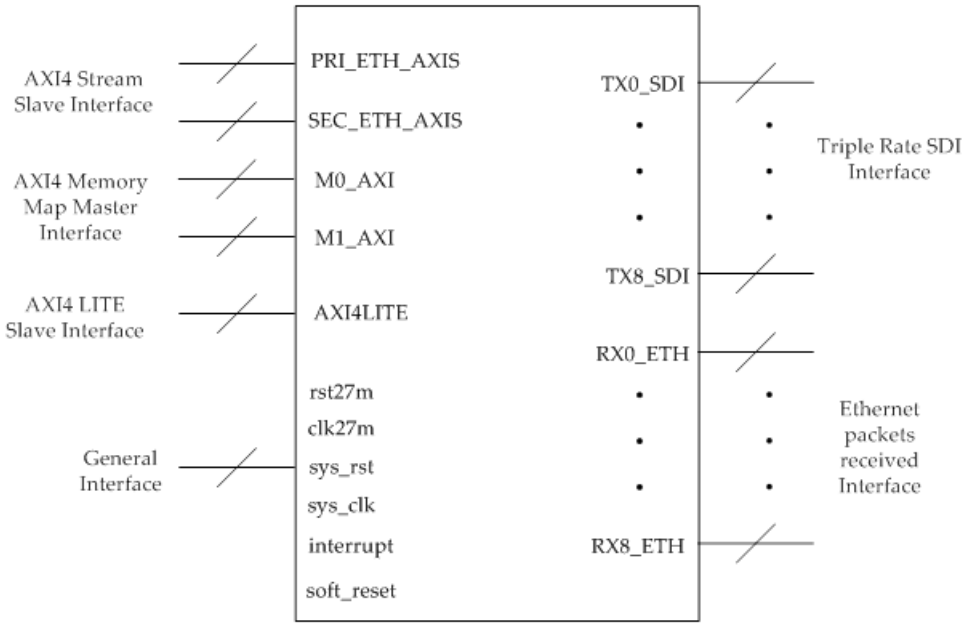

Figure 2-2: **SMPTE 2022-5/6 Video over IP Receiver Core Interface**

Common Interface

Table 2-4 summarizes the signals which are either shared by or are not part of the dedicated SDI, AXI4-Stream, AXI4, or AXI4-Lite control interfaces.

Table 2-4: **Common Interface Signals**

Signal Name	Direction	Width	Description
rst27m	In	1	27 Mhz domain reset
clk27m	In	1	27 Mhz clock and is used for timekeeping
sys_rst	In	1	System domain reset.

Table 2-4: **Common Interface Signals *(Cont'd)***

Signal Name	Direction	Width	Description
sys_clk	In	1	System clock
interrupt	Out	1	Reserved
soft_reset	Out	1	Core reset generated from specific control register bit

AXI4 Memory Master Interface

The core uses an AXI4 interface to connect to the AXI4 interconnect. The AXI4 Interconnect provides the access to the external memory through the AXI Double Data Rate (DDR) controller. See the *LogiCORE IP AXI Interconnect Product Guide* (PG059) [Ref 3] for more information.

Table 2-5: **AXI4 Memory Interface Signals**

Signal Name	Direction	Width	Description
m0_axi_awid	Out	1	Write Address Channel Transaction ID
m0_axi_awaddr	Out	32	Write Address Channel Address
m0_axi_awlen	Out	8	Write Address Channel Burst Length code
m0_axi_awsize	Out	3	Write Address Channel Transfer Size code
m0_axi_awburst	Out	2	Write Address Channel Burst Type
m0_axi_awlock	Out	2	Write Address Channel Atomic Access Type
m0_axi_awcache	Out	4	Write Address Channel Cache Characteristics
m0_axi_awprot	Out	3	Write Address Channel Protection Bits
m0_axi_awqos	Out	4	Write Address Channel Quality of Service
m0_axi_awvalid	Out	1	Write Address Channel Valid
m0_axi_awready	In	1	Write Address Channel Ready
m0_axi_wdata	Out	256	Write Data Channel Data
m0_axi_wstrb	Out	32	Write Data Channel Data Byte Strobes
m0_axi_wlast	Out	1	Write Data Channel Last Data Beat
m0_axi_wvalid	Out	1	Write Data Channel Valid
m0_axi_wready	In	1	Write Data Channel Ready
m0_axi_bid	In	1	Write Response Channel Transaction ID
m0_axi_bresp	In	2	Write Response Channel Response Code
m0_axi_bvalid	In	1	Write Response Channel Valid
m0_axi_bready	Out	1	Write Response Channel Ready
m0_axi_arid	Out	1	Read Address Channel Transaction ID
m0_axi_araddr	Out	32	Read Address Channel Address
m0_axi_arlen	Out	8	Read Address Channel Burst Length code
m0_axi_arsize	Out	3	Read Address Channel Transfer Size code
m0_axi_arburst	Out	2	Read Address Channel Burst Type

Table 2-5: **AXI4 Memory Interface Signals *(Cont'd)***

Signal Name	Direction	Width	Description
m0_axi_arlock	Out	2	Read Address Channel Atomic Access Type
m0_axi_arcache	Out	4	Read Address Channel Cache Characteristics
m0_axi_arprot	Out	3	Read Address Channel Protection Bits
m0_axi_arqos	Out	4	AXI4 Read Address Channel Quality of Service
m0_axi_arvalid	Out	1	Read Address Channel Valid
m0_axi_arready	In	1	Read Address Channel Ready
m0_axi_rid	In	1	Read Data Channel Transaction ID
m0_axi_rdata	In	256	Read Data Channel Data
m0_axi_rresp	In	2	Read Data Channel Response Code
m0_axi_rlast	In	1	Read Data Channel Last Data Beat
m0_axi_rvalid	In	1	Read Data Channel Valid
m0_axi_rready	Out	1	Read Data Channel Ready
m1_axi_awid	Out	1	Write Address Channel Transaction ID
m1_axi_awaddr	Out	32	Write Address Channel Address
m1_axi_awlen	Out	8	Write Address Channel Burst Length code
m1_axi_awsize	Out	3	Write Address Channel Transfer Size code
m1_axi_awburst	Out	2	Write Address Channel Burst Type
m1_axi_awlock	Out	2	Write Address Channel Atomic Access Type
m1_axi_awcache	Out	4	Write Address Channel Cache Characteristics
m1_axi_awprot	Out	3	Write Address Channel Protection Bits
m1_axi_awqos	Out	4	Write Address Channel Quality of Service
m1_axi_awvalid	Out	1	Write Address Channel Valid
m1_axi_awready	In	1	Write Address Channel Ready
m1_axi_wdata	Out	256	Write Data Channel Data
m1_axi_wstrb	Out	32	Write Data Channel Data Byte Strobes
m1_axi_wlast	Out	1	Write Data Channel Last Data Beat
m1_axi_wvalid	Out	1	Write Data Channel Valid
m1_axi_wready	In	1	Write Data Channel Ready
m1_axi_bid	In	1	Write Response Channel Transaction ID

Table 2-5: **AXI4 Memory Interface Signals *(Cont'd)***

Signal Name	Direction	Width	Description
m1_axi_bresp	In	2	Write Response Channel Response Code
m1_axi_bvalid	In	1	Write Response Channel Valid
m1_axis_bready	Out	1	Write Response Channel Ready
m1_axi_arid	Out	1	Read Address Channel Transaction ID
m1_axi_araddr	Out	32	Read Address Channel Address
m1_axi_arlen	Out	8	Read Address Channel Burst Length code
m1_axi_arsize	Out	3	Read Address Channel Transfer Size code
m1_axi_arburst	Out	2	Read Address Channel Burst Type
m1_axi_arlock	Out	2	Read Address Channel Atomic Access Type
m1_axi_arcache	Out	4	Read Address Channel Cache Characteristics
m1_axi_arprot	Out	3	Read Address Channel Protection Bits
m1_axi_arqos	Out	4	AXI4 Read Address Channel Quality of Service
m1_axi_arvalid	Out	1	Read Address Channel Valid
m1_axi_arready	In	1	Read Address Channel Ready
m1_axi_rid	In	1	Read Data Channel Transaction ID
m1_axi_rdata	In	256	Read Data Channel Data
m1_axi_rresp	In	2	Read Data Channel Response Code
m1_axi_rlast	In	1	Read Data Channel Last Data Beat
m1_axi_rvalid	In	1	Read Data Channel Valid
m1_axi_rready	Out	1	Read Data Channel Ready

AXI4-Stream Slave Interface

See the *LogiCORE IP 10-Gigabit Ethernet MAC Product Guide* (PG072) [Ref 4] for more information.

Table 2-6: **AXI4-Stream Interface Signals**

Signal Name	Direction	Width	Description
pri/sec_eth_rst	In	1	Active-High reset from core
pri/sec_eth_clk	In	1	Recovered clock from XGMAC
pri/sec_s_axis_tdata[63:0]	In	64	AXI4-Stream Data from XGMAC
pri/sec_s_axis_tkeep[7:0]	In	8	AXI4-Stream Data Control from XGMAC

Table 2-6: **AXI4-Stream Interface Signals *(Cont'd)***

Signal Name	Direction	Width	Description
pri/sec_s_axis_tvalid	In	1	AXI4-Stream Data Valid from XGMAC
pri/sec_s_axis_tlast	In	1	AXI4-Stream signal from XGMAC indicating an end of packet
pri/sec_s_axis_tuser	In	1	AXI4-Stream User Sideband Interface from XGMAC • 1 indicates that a good packet has been received. • 0 indicates that a bad packet has been received.

SMPTE SD/HD/3G-SDI Interface

See the *Society of Motion Picture and Television Engineers (SMPTE) SD/HD/3G-SDI Product Guide* (PG071) [Ref 5] for more information.

Table 2-7: **SMPTE SD/HD/3G-SDI Interface Signals**

Signal Name(1)	Direction	Width	Description
tx[0-7]_rst	In	1	Reset
tx[0-7]_clk	In	1	Clock input. It must have a frequency of 74.25 MHz or 74.25/1.001 MHz for HD-SDI, 148.5 MHz or 148.5/1.001 MHz for 3G-SDI, and 148.5 MHz for SD-SDI mode.
tx[0-7]_ce	Out	3	To tx[0-7]_ce of SMPTE SD/HD/3G-SDI
tx[0-7]_din_rdy	Out	1	To tx[0-7]_din_rdy of SMPTE SD/HD/3G-SDI
tx[0-7]_ds1a	Out	10	To tx[0-7]_ds1a of SMPTE SD/HD/3G-SDI
tx[0-7]_ds1b	Out	10	To tx[0-7]_ds1b of SMPTE SD/HD/3G-SDI
tx[0-7]_ds2a	Out	10	To tx[0-7]_ds2a of SMPTE SD/HD/3G-SDI
tx[0-7]_ds2b	Out	10	To tx[0-7]_ds2b of SMPTE SD/HD/3G-SDI
tx[0-7]_level_b_3g	Out	1	To tx[0-7]_level_b_3g of SMPTE SD/HD/3G-SDI
tx[0-7]_mode	Out	1	To tx[0-7]_mode of SMPTE SD/HD/3G-SDI
tx[0-7]_m	Out	1	In HD-SDI and 3G-SDI modes, this output indicates which bit rate is received. If this output is Low, it indicates a bit rate of 1.485 Gb/s in HD-SDI mode and 2.97 Gb/s in 3G-SDI mode. If this output is High, it indicates a bit rate of 1.485/1.001 Gb/s in HD-SDI mode and 2.97/1.001 Gb/s in 3G-SDI mode.

1. [0-7] is index that represent up to 8 channels support for SDI streams.

Ethernet Packets Received Interface

See the *SMPTE 2022-5/6* reference design for more information.

Table 2-8: **Ethernet Packets Received Interface Signals**

Signal Name[(1)]	Direction	Width	Description
rx[0-7]_pri/sec_rtp_pkt_recv	Out	1	Pulse indicating receiving of RTP packet from primary/secondary link. (synchronous to pri/sec_eth_clk)
rx[0-7]_pri/sec_rtp_seq_num	Out	16	Sequence number of RTP packet received from primary/secondary link. (Synchronous to pri/sec_eth_clk)
rx[0-7]_rtp_pkt_buffered	Out	16	Amount of RTP packets buffered. (Synchronous to pri_eth_clk)
rx[0-7]_rtp_pkt_transmit	Out	1	Pulse indicating consumption of RTP packet for SDI output. (Synchronous to pri_eth_clk)
rx[0-7]_pkt_lock	Out	1	Indication of channel locking to certain payload. (Synchronous to pri_eth_clk)
rx[0-7]_pri/sec_vid_ts	Out	32	Video timestamp of the RTP packet received from primary/secondary link. (Synchronous to pri/sec_eth_clk)
rx[0-7]_pri/sec_rtp_ts	Out	32	RTP timestamp of the RTP packet received from primary/secondary link. (Synchronous to pri/sec_eth_clk)
rx[0-7]_playout_ready	Out	1	Indication of channel ready for playing out the TS data. (Synchronous to pri_eth_clk)

1. [0-7] is index that represent up to 8 channels support for SDI streams.

AXI4-Lite Control Interface

The AXI4-Lite interface allows you to dynamically control parameters within the core. Core configuration can be accomplished using an embedded ARM® or soft system processor such as MicroBlaze™.

The core can be controlled through the AXI4-Lite interface using read and write transactions to the SMPTE 2022-5/6 Video over IP Receiver register space.

The AXI4-Lite slave interface facilitates integrating the core into a processor system, or along with other video or AXI4-Lite compliant IP, connected through the AXI4-Lite interface to an AXI4-Lite master. See the *LogiCORE IP AXI Interconnect Product Guide* (PG059) [Ref 3] for more information.

Table 2-9: **AXI4-Lite Interface Signals**

Signal Name	Direction	Width	Description
s_axi_aclk	In	1	AXI4-Lite clock
s_axi_aresetn	In	1	AXI4-Lite active-Low reset
s_axi_awaddr	In	9	AXI4-Lite Write Address Bus
s_axi_awvalid	In	1	AXI4-Lite Write Address Channel Write Address Valid
s_axi_wdata	In	32	AXI4-Lite Write Data Bus
s_axi_wstrb	In	4	AXI4-Lite Write Data Channel Data Byte Strobes
s_axi_wvalid	In	1	AXI4-Lite Write Data Channel Write Data Valid
s_axi_bready	In	1	AXI4-Lite Write Response Channel Ready. Indicates target is ready to receive response.
s_axi_araddr	In	9	AXI4-Lite Read Address Bus
s_axi_arvalid	In	1	AXI4-Lite Read Address Channel Read Address Valid
s_axi_rready	In	1	AXI4-Lite Read Data Channel Read Data Ready. Indicates target is ready to accept the read data.
s_axi_arready	Out	1	AXI4-Lite Read Address Channel Read Address Ready. Indicates target is ready to accept the read address.
s_axi_rdata	Out	32	AXI4-Lite Read Data Bus
s_axi_rresp	Out	2	AXI4-Lite Read Response Channel Response. Indicates results of the read transfer.
s_axi_rvalid	Out	1	AXI4-Lite Read Data Channel Read Data Valid
s_axi_wready	Out	1	AXI4-Lite Write Data Channel Write Data Ready. Indicates target is ready to accept the write data.
s_axi_bresp	Out	2	AXI4-Lite Write Response Channel. Indicates results of the write transfer.
s_axi_bvalid	Out	1	AXI4-Lite Write Response Channel Response Valid. Indicates response is valid.
s_axi_awready	Out	1	AXI4-Lite Write Address Channel Write Address Ready.

Register Space

The SMPTE 2022-5/6 Video over IP Receiver register space is partitioned to General and Channel specific registers. See the *SMPTE 2022-5/6* reference design for more information on register usage.

Table 2-10: **AXI4-Lite Register Map**

Address (Hex)	Register Name	Access Type	Default Value(HEX)	Description	
				Bit Range	Value
General					
0x0000	control	R/W	0x00000000	Control	
				31:2	Reserved
				1	Channel Register Update. 0 – Host processor actively updating the channel registers 1 – Updating process completed
				0	Reserved
0x0004	reset	R/W	0x00000000	Reset	
				31:1	Reserved
				0	1 – Reset the configuration registers and set soft_reset signal High
0x000C	channel_access	R/W	0x00000000	Channel Access	
				31	0 - primary 1 - secondary
				30:8	Reserved
				7:0	The channel number to access the channel space registers
0x0020	sys_cfg	R	Based on configured generics.	System Configuration	
				31	Seamless switching supported
				30	FEC recovery supported
				29:8	Reserved
				7:0	Number of channels supported

Table 2-10: **AXI4-Lite Register Map *(Cont'd)***

Address (Hex)	Register Name	Access Type	Default Value(HEX)	Description	
				Bit Range	Value
0x0024	version	R	0x04000000	Version	
				31:24	Version major
				23:16	Version minor
				15:12	Version revision
				11:8	Patch ID
				7:0	Revision number
0x0028	network_path_differential	R/W	0x00000000	Network Path Differential	
				31:0	max accepted delay between 2 streams in hitless, value based on 27MHz clock ticks
0x0034	fec_buf_base_addr	R/W	0x00000000	FEC Buffer Base Address	
				31:0	Base address on where the buffer begins in the DDR
0x0038	fec_buf_pool_size	R/W	0x00000000	FEC Buffer Pool Size	
				31:0	No. of Bytes of memory space to cater for FEC buffer
0x003C	pri_recv_pkt_cnt	R	0x00000000	Primary Received Packet Count	
				31:0	Number of packets received in the primary stream
0x0040	sec_recv_pkt_cnt	R	0x00000000	Secondary Received Packet Count	
				31:0	Number of packets received in the secondary stream
0x0044	pri_err_pkt_cnt	R	0x00000000	Primary Errored Packet Count	
				31:0	Number of errored packets received in the primary stream
0x0048	sec_err_pkt_cnt	R	0x00000000	Secondary Errored Packet Count	
				31:0	Number of errored packets received in the secondary stream

Table 2-10: **AXI4-Lite Register Map *(Cont'd)***

Address (Hex)	Register Name	Access Type	Default Value(HEX)	Description	
				Bit Range	Value
0x004C	pri_discard_pkt_cnt	R	0x00000000	Primary Discarded Packet Count	
				31:0	Number of not matching pkts received in the primary stream
0x0050	sec_discard_pkt_cnt	R	0x00000000	Secondary Discarded Packet Count	
				31:0	Number of not matching pkts received in the secondary stream
0x0054	gen_stat_reset	R/W	0x00000000	General Statistics Reset	
				31:6	Reserved
				5	Reset sec_discarded_pkts_cnt
				4	Reset pri_discarded_pkts_cnt
				3	Reset sec_err_pkts_cnt
				2	Reset pri_err_pkts_cnt
				1	Reset sec_recv_pkts_cnt
				0	Reset pri_recv_pkts_cnt
Channel					
0x0084	ip_hdr_param	R	0x00000000	IP Header Parameter	
				31:16	Reserved
				15:8	type of service(TOS)
				7:0	time to live(TTL)
0x0088	match_vlan	R/W	0x00000000	Match VLAN	
				31	VLAN filtering 0 - Filter stream without VLAN, 1 - Filter stream with VLAN having tag info in bit 15:0
				30:16	Reserved
				15:0	16-bit VLAN tag info

Table 2-10: **AXI4-Lite Register Map *(Cont'd)***

Address (Hex)	Register Name	Access Type	Default Value(HEX)	Description	
				Bit Range	Value
0x008C	match_dest_ip_addr	R/W	0x00000000	Match Destination IP Address	
				31:0	32-bit IP host low address
0x009C	match_src_ip_addr	R/W	0x00000000	Match Source IP Address	
				31: 0	32-bit source IP address
0x00AC	match_src_port	R/W	0x00000000	Match UDP Source Port	
				31:16	Reserved
				15:0	16-bit UDP source port address
0x00B0	match_dest_port	R/W	0x00000000	Match UDP Destination Port	
				31:16	Reserved
				15:0	16-bit UDP destination port address used to filter
0x00B4	match_sel	R/W	0x00000000	Matching Selection	
				31:6	Reserved
				5	To match SSRC
				4	To match UDP dest port
				3	To match UDP src port
				2	To match Destination IP
				1	To match Source IP
				0	To match VLAN
0x00B8	link_reordered_ pkt_cnt	R	0x00000000	Link Reordered Packet Count	
				31:0	Number of reordered packets
0x00BC	link_stat_reset	R/W	0x00000000	Link Statistics Reset	
				31:2	Reserved
				1	Reset valid_pkts_cnt
				0	Reset reordered_ pkts_cnt
0x00C0	link_valid_media_pkt_cnt	R	0x00000000	Link Valid Media Packet Count	
				31:0	Number of valid media packets received in the link per channel
Channel [Shared]					

Table 2-10: **AXI4-Lite Register Map *(Cont'd)***

Address (Hex)	Register Name	Access Type	Default Value(HEX)	Description	
				Bit Range	Value
0x0100	chan_en	R/W	0x00000000	Channel Enable	
				31:2	Reserved
				1	RTP timestamp bypass 1 - operate on RTP stream with no timestamp 0 - operate on RTP stream with timestamp
				0	Channel enable 0 - disable channel. 1 - enable channel
0x010C	chan_stat_reset	R/W	0x00000000	Channel Statistics Reset	
				31:6	Reserved
				5	Reset oor_ pkt_cnt
				4	Reset unrec_pkt_cnt
				3	Reset media_buffer_ov
				2	Reset dup_pkt_cnt
				1	Reset corr_ pkt_cnt
				0	Reset chan_valid_media_pkt_cnt
0x0110	match_ssrc	R/W	0x00000000	Match SSRC	
				31:0	32-bit SSRC value used to match between primary and secondary links to the channel
0x0114	sdi_pkt_status	R	0x00000000	SDI Packet Status	
				1	1 - SDI frame out of sync. Packet count for the SDI frame does not match the video format
				0	Packet size locked indicator 0 - Not locked 1 - Locked

Table 2-10: **AXI4-Lite Register Map *(Cont'd)***

Address (Hex)	Register Name	Access Type	Default Value(HEX)	Description	
				Bit Range	Value
0x0118	vid_src_fmt	R	0x00000000	Video Source Format	
				31:28	MAP(refer to SMPTE 2022-6 spec)
				27:20	FRAME(refer to SMPTE 2022-6 spec)
				19:12	FRATE(refer to SMPTE 2022-6 spec)
				11:8	SAMPLE(refer to SMPTE 2022-6 spec)
				7:0	Reserved
0x011C	playout_delay	R/W	0x00000000	Playout delay	
				31:0	Set wait time to SDI playout upon incoming stream packet size lock and first detection of end of SDI frame. Value based on 27MHz clock ticks
0x0124	fec_param	R	0x00000000	FEC parameter	
				31:22	Reserved
				21	FEC protect level. 1 - level B. 0 - level A.
				20	1 - FEC parameters locked
				19:10	10-bit D value from the received header
				9:0	10-bit L value from the received header
0x0128	seamless_protect	R	0x00000000	Seamless Protect	
				31	Seamless status 0- not protected. 1 - protected
				30:0	RTP timestamp difference between incoming primary and secondary stream packets

Table 2-10: **AXI4-Lite Register Map *(Cont'd)***

Address (Hex)	Register Name	Access Type	Default Value(HEX)	Description	
				Bit Range	Value
0x012C	media_buf_base_addr	R/W	0x00000000	Media Buffer Base Address	
				31:0	Base address on where the buffer begins in the DDR
0x0130	media_pkt_buf_size	R/W	0x00000000	Media Packet Buffer Size	
				31:16	Reserved
				15:0	The number of RTP packets to hold in the DDR for the channel
0x0134	chan_valid_media_pkt_cnt	R	0x00000000	Channel Valid Media Packet Count	
				31:0	Number of valid media packets received in the channel
0x0138	rec_ pkt_cnt	R	0x00000000	Recovered Packet Count	
				31:0	Number of FEC recovered packets
0x013C	dup_ pkt_cnt	R	0x00000000	Duplicated Packet Count	
				31:0	Number of duplicated packets in the channel
0x0144	pkt_interval	R	0x00000000	Packet interval	
				31:0	Timestamp difference between 2 consecutive sequence number packets in the merge stream
0x0154	media_buffer_ov	R	0x00000000	Media Buffer Overflow	
				31:1	Reserved
				0	1 - indicate that media buffer overflowed
0x0158	unrec_ pkt_cnt	R	0x00000000	Unrecoverable Packet Count	
				31:0	Number of unrecoverable packets
0x0160	oor_ pkt_cnt	R	0x00000000	Out-Of-Range Packet Count	
				31:0	Number of out-of-range packets

CONTROL (0x000) Register

Bit 1 of the CONTROL register is a write-done semaphore for the host processor, which facilitates committing all user register updates in the channel space simultaneously. One set of registers (the processor registers) is directly accessed by the processor interface, while the other set (the active set) is actively used by the core. New values written to the processor registers are copied over to the active set if and only if the register update bit is set. Setting the bit to 0 before updating multiple registers and then setting the bit to 1 when updates are completed ensures all channel space registers are updated simultaneously.

RESET (0x004) Register

Bit 0 is software reset. When High, the configuration registers are held at reset state. At the same time, the soft_reset signal at the core interface is held High.

CHANNEL_ACCESS (0x00C) Register

Set the channel to access. All the primary link and secondary link channels share the same set of register address in the channel space. To access secondary link channels, set bit 31 to 1. Only 0x084 - 0x0C0 registers are available for secondary link. Bit 7-0 represent the channel number in standard binary count.

SYS_CFG (0x020) Register

System configuration of the core.

Bit 31 High indicates seamless switching support.

Bit 30 High indicates FEC engine is included.

Bit 7-0 gives the number of channels available to use.

VERSION (0x024) Register

Bit fields of the register facilitate software identification of the exact version of the hardware peripheral incorporated into a system. The core driver can take advantage of this read-only value to verify that the software is matched to the correct version of the hardware.

NETWORK_PATH_DIFFERENTIAL (0x028) Register

Set the maximum delay between primary and secondary link for the core to operate in seamless switching mode. The value is based on a 27MHz clock tick.

FEC_BUF_BASE_ADDR (0x034) Register

This sets the base address of the memory allocated in the DDR to store the FEC packets for recovery.

FEC_BUF_POOL_SIZE (0x038) Register

This register allocates the memory buffer size in the DDR for FEC packets storage. The value is in terms of bytes.

PRI_RECV_PKT_CNT (0x03C) Register

Primary received packet counter increments when a packet is filtered into the channels in the primary link. Register 0x054, bit 0 resets it.

SEC_RECV_PKT_CNT (0x040) Register

Secondary received packet counter increments when a packet is filtered into the channels in the secondary link. Register 0x054, bit 1 resets it.

PRI_ERR_PKT_CNT (0x044) Register

Primary error packet counter increments when a packet is identified as bad frame from the MAC core in the primary link. Register 0x054, bit 2 resets it.

SEC_ERR_PKT_CNT (0x048) Register

Secondary error packet counter increments when a packet is identified as bad frame from the MAC core in the secondary link. Register 0x054, bit 3 resets it.

PRI_DISCARD_PKT_CNT (0x04C) Register

Primary discard packet counter increments when a packet is not accepted for any of the channels in the primary link. Register 0x054, bit 4 resets it.

SEC_DISCARD_PKT_CNT (0x050) Register

Secondary discard packet counter increments when a packet is not accepted for any of the channels in the secondary link. Register 0x054, bit 5 resets it.

GEN_STAT_RESET (0x054) Register

A High to Bit 5 resets secondary discarded packet counter (register 0x050).

A High to Bit 4 resets primary discarded packet counter (register 0x04C).

A High to Bit 3 resets secondary error packet counter (register 0x048).

A High to Bit 2 resets primary error packet counter (register 0x044).

A High to Bit 1 resets secondary received packet counter (register 0x040).

A High to Bit 0 resets primary received packet counter (register 0x03C).

IP_HDR_PARAM (0x084) Register

Read-only status on the IP header fields.

Bit 15-8 is Type of service (TOS).

Bit 7-0 is Time to live (TTL).

MATCH_VLAN (0x088) Register

This parameter is used in filtering the packets for the channel. Configure Bit 15-0 for VLAN tag matching. Set Bit 31 for valid VLAN tag.

If Bit 31 is High, packets with VLAN tag and matching information is filtered into the channel.

If Bit 31 is Low, packets without VLAN tag is filtered into the channel.

Table 2-11: **Usage of Match VLAN (0x88) with Match Select (0xB4)**

Match Select (0xB4) Bit '0' To Match VLAN	Match VLAN (0x88) Bit '31' VLAN Filtering	Incoming Packet with VLAN	Incoming Packet without VLAN
0	0	Not applicable	
	1		
1	0	Not Match	Match
	1	Match if VLAN TAG ID = Bit [15:0] of Match VLAN	Not Match

MATCH_DEST_IP_ADDR (0x08C) Register

This parameter is used in filtering the packets for the channel. Configure Bit 31-0 for Destination IP address matching.

MATCH_SRC_IP_ADDR (0x09C) Register

This parameter is used in filtering the packets for the channel. Configure Bit 31-0 for Source IP address matching.

MATCH_SRC_PORT (0x0AC) Register

This parameter is used in filtering the packets for the channel. Configure Bit 15-0 for UDP Source Port matching.

MATCH_DEST_PORT (0x0B0) Register

This parameter is used in filtering the packets for the channel. Configure Bit 15-0 for UDP Destination Port matching.

MATCH_SEL (0x0B4) Register

This register sets the parameters to match the incoming packet and filter it to the channel.

- Bit 5 matches the SSRC field in the RTP header of the packet to the configured register 0x110.
- Bit 4 matches the Destination Port field in the UDP header of the packet to the configured register 0x0B0.
- Bit 3 matches the Source Port field in the UDP header of the packet to the configured register 0x0AC.
- Bit 2 matches the Destination address field in the IP header of the packet to the configured register 0x08C.
- Bit 1 matches the Source address field in the IP header of the packet to the configured register 0x09C.
- Bit 0 matches the IEEE 802.1Q tag in the Ethernet frame to the configured register 0x088.

Note that each link media packet can only be filtered into 1 particular channel.

LINK_REORDERED_PKT_CNT (0x0B8) Register

This counter tracks the number of incoming media packets that are reordered in the link channel (primary or secondary). A packet is considered reorder when its sequence number is less than the previous packet. Register 0x0BC, bit 0 resets the counter.

LINK_STAT_RESET (0x0BC) Register

- A High to Bit 1 resets link valid media packet counter (register 0x0C0).

- A High to Bit 0 resets link reordered packet counter (register 0x0B8).

LINK_VALID_MEDIA_PKT_CNT (0x0C0) Register

This counter tracks the number of incoming media packets that match to the channel in the link (primary or secondary). Register 0x0BC, bit 1 resets the counter.

CHANNEL_ENABLE (0x100) Register

- Set Bit 0 to enable the channel operation.
- Set Bit 1 to bypass media packet RTP timestamp check for the channel. Out-of-range counter is not active. Out-of-range packets are not discarded.

CHAN_STAT_RESET (0x10C) Register

- A High to Bit 5 resets out-of-range counter (register 0x160).
- A High to Bit 4 resets unrecoverable packet counter (register 0x158).
- A High to Bit 3 resets media buffer overflow (register 0x154).
- A High to Bit 2 resets duplicated packet counter (register 0x13C).
- A High to Bit 1 resets recovered packet counter (register 0x138).
- A High to Bit 0 resets channel valid media packet counter (register 0x134).

MATCH_SSRC (0x110) Register

This parameter is used in filtering the packets for the channel. Configure Bit 31-0 for RTP Synchronization Source identifier matching.

SDI_PKT_STATUS (0x114) Register

Read only status on the media packet size detected by the channel.

Bit 0 is High after detection of 32 consecutive media packets with the same video source format for the channel. Register 0x118 shows the video source format of the SDI video. Change of video source format thereafter will reset this bit and restart the whole process of detection.

Bit 1 High indicates the received amount of packets per frame based on the locked video source format is incorrect. The channel needs to be reset to ensure proper operation.

VID_SRC_FMT (0x118) Register

- Bit 31-28 refers to MAP parameter in SMPTE 2022-6 specification

- Bit 27-20 refers to FRAME parameter in SMPTE 2022-6 specification
- Bit 19-12 refers to FRATE parameter in SMPTE 2022-6 specification
- Bit 11-8 refers to SAMPLE parameter in SMPTE 2022-6 specification

PLAYOUT_DELAY (0x11C) Register

Sets the wait time before the buffered SDI data is ready for play out after packet size lock (register 0x120, bit 0) and marker bit packet detected. The value is based on 27MHz clock tick.

FEC_PARAM (0x124) Register

Read-only status on the FEC parameters detected by the channel.

Bit 20 is High upon first detection of FEC packet for the channel. Detection will not start before register 0x120, bit 0 is High.

When Bit 20 is High:

- Bit 9-0 shows the L value of FEC matrix.
- Bit 19-10 shows the D value of FEC matrix.
- Bit 21 shows the FEC protection level. Bit 21 is Low for Level A stream and High for Level B.

SEAMLESS_PROTECT (0x128) Register

Bit 30-0 samples the RTP timestamp difference between incoming packets from the primary and secondary links of a channel.

Bit 31 indicates whether the channel is under seamless protection. The channel is considered protected when the RTP timestamps of the media packets from both primary and secondary links fall within the range defined by NETWORK_PATH_DIFFERRENTIAL and PLAYOUT_DELAY.

MEDIA_PKT_BASE_ADDR (0x12C) Register

This register sets the base address of the memory buffer allocated in the DDR to store the media packets for FEC recovery and play out.

MEDIA_PKT_BUF_SIZE (0x130) Register

This register sets the maximum number of media packets to be stored in the DDR for the channel. It has a limitation of 16 bits (Bit 15-0) and has to be in written in the value of (2^n-1).

CHAN_VALID_MEDIA_PKT_CNT (0x134) Register

This counter increases when a media packet gets written into the memory buffer. It is per channel and register 0x10C, bit 0 resets it.

REC_PKT_CNT (0x138) Register

Recovered packet counter increases when a media packet is being recovered by the FEC engine. This counter is per channel and register 0x10C, bit 1 resets it.

DUP_PKT_CNT (0x13C) Register

Duplicated packet counter increases when the combined primary and secondary link of the channel has received a media packet with a sequence number that is already in the memory buffer. This incoming packet is discarded and not processed further. The counter is per channel and register 0x10C, bit 2 resets it.

PKT_INTERVAL (0x144) Register

The difference in RTP timestamps between 2 consecutive sequence numbers from the media packets stream. The value is in terms of 27MHz clock tick.

MEDIA_BUF_OV (0x154) Register

Bit 0 of this register flags High when incoming packets fill up the memory buffer faster than the outgoing packets can clear. This status is per channel and register 0x10C, bit 3 resets it.

UNREC_PKT_CNT (0x158) Register

Unrecoverable packet counter increases when the outgoing packet is not available. There is no previous incoming packet or the FEC engine is not able to recover the missing packet from the matrix. This counter is per channel and register 0x10C, bit 4 resets it.

OOR_PKT_CNT (0x160) Register

Out-of-range packet counter increases when the difference between the incoming and outgoing packets' RTP timestamp is greater than the value of (NETWORK_PATH_DIFFERRENTIAL + PLAYOUT_DELAY). The incoming packet is discarded and not processed further. This counter is per channel and register 0x10C, bit 5 resets it.

Note that massive out-of-range packets may cause the core to stop working.

Designing with the Core

The core is for broadcast applications that require bridging between SMPTE video connectivity standards SD/HD/3G-SDI and 10Gb/s Ethernet. The core takes in Ethernet packets encapsulated in accordance with SMPTE 2022-5/6 and maps them in uncompressed SD/HD/3G-SDI streams to the SMPTE SD/HD/3G-SDI core. It receives Ethernet packets through the AXI4-Stream interface from the 10 Gb/s Ethernet MAC. The core uses the AXI4 memory interface to transfer data between the core and external DDR memory. The register control interface is compliant with AXI4-Lite interface. See *SMPTE 2022-5/6 High Bit Rate Media Transport Over IP Networks with Forward Error Correction* (XAPP1199) [Ref 1] for more information.

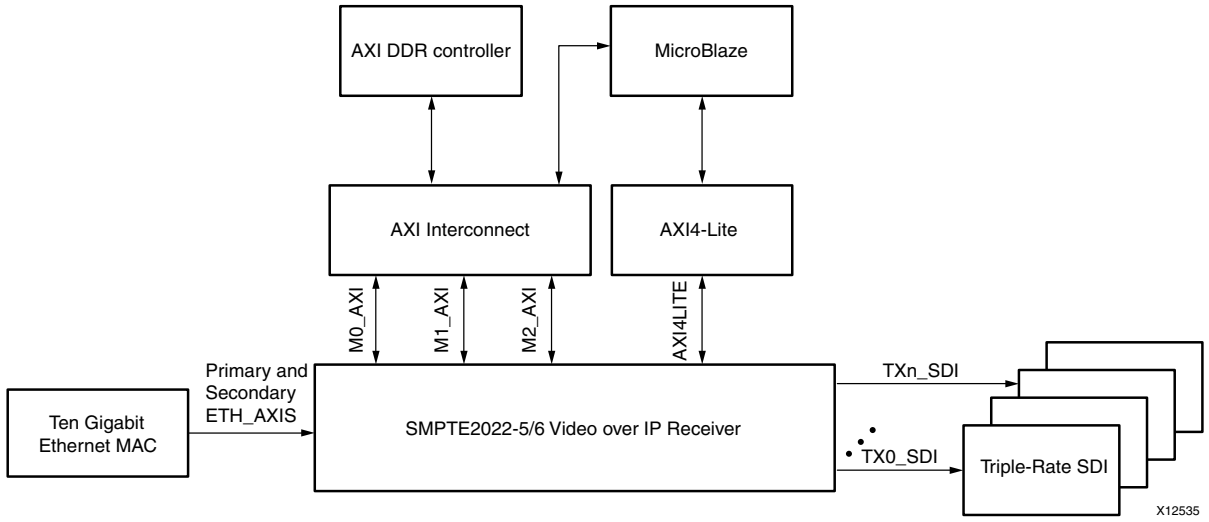

Figure 3-1: **SMPTE 2022-5/6 Video over IP Receiver System Built with Other Xilinx IP Cores**

1. Primary and Secondary ETH_AXIS exist only when seamless switching enabled.

Note: There is an option to include Forward Error Correction engine in the SMPTE 2022-5/6 Video over IP Receiver core. Adding this enables the receiver to recover IP packets lost to the network transmission errors and hence ensure the quality of the uncompressed video. However, it will increase the resource count in the FPGA as well as the usage of external memory. Enabling seamless switching adds a redundancy protection link for packets lost to network transmission errors, which also increases device resource count. Reset individual channel can be achieved by setting Bit 0 Low in `chan_en` register (register offset 0x100). To reset the core, all active individual channels must be reset and following by setting Bit 0 Low in reset register (register offset 0x004).

Clocking

The core has six clock domains:

- 27MHz clock domain
- SDI video clock domain
- System clock domain recommended running at 200 MHz
- Primary Ethernet clock domain at 156.25 MHz
- Secondary Ethernet clock domain at 156.25 MHz
- AXI4-Lite clock domain recommended at 100 MHz

Resets

The SMPTE 2022-5/6 Video over IP Receiver core has five (or six when Seamless Switching enabled) main resets:

- Primary Ethernet link reset, pri_eth_rst
- Secondary Ethernet link reset, sec_eth_rst
- System domain reset, sys_rst
- 27Mhz domain reset, rst27m
- SDI domain reset, tx<port_num>_rst
- AXI4-Lite domain reset, s_axi_aresetn

Reset Requirements

- The resets must be synchronous to their individual clock domains.
- A minimum of 16 clocks assertion is recommended.
- The ordering of reset de-assertion is not important except the `pri_eth_rst` and `sec_eth_rst` have to be the last.

Refer to *SMPTE 2022-5/6 High Bit Rate Media Transport Over IP Networks with Forward Error Correction* (XAPP1199) [Ref 1].

Memory Requirement and Register Configurations

Base Address, Packet Buffer and Playout Delay Registers Configurations

Follow the computation below to get the configuration values to obtain the Media Packet Buffer Size and play out delay (**in bold**). These configuration values must be set before enabling the channel.

*notes, all computation below are per-channel basis.

SMPTE 2022-567 Computation is based with this SDI rate:

Table 3-1: **SDI Rate**

SDI Format	Maximum SDI Bitrate (Mbps)
SD	270
HD	1485
3G	2970

$$Video\ Packet_{rate} = \frac{SDI_{rate}}{1376\ Bytes * 8}$$

*SMPTE 2022-567 Payload Size: 1376 Bytes

Then compute for packet buffered for packet delay;

Then compute for packet buffered for packet delay;

*$Network\ Path\ Differential_{in\ seconds}$ is set when Seamless Switching is being used.

$$Packet\ Buffered_{Network\ Path\ Differential} = Network\ Path\ Differential_{in\ seconds} * Video\ Packet_{rate}$$

Then compute packets buffered for FEC correction;

$$Packet\ Buffered_{FEC\ Correction} = (FEC_L * FEC_D * 2) + Packet\ Margin$$

Packet margin is set to 64 as to give some time for FEC recovery process.

The summed packet buffered is computed as;

$$Summed\ Packet\ Buffered = (Packet\ Buffered_{Network\ Path\ Differential} * 2) + Packet\ Buffered_{FEC\ Correction}$$

1. To calculate Media Packet Buffer Size,

*the maximum Media Packet Buffer Size is 65535 (0xFFFF)

$$\boldsymbol{Media\ Packet\ Buffer\ Size} = [(\log_2[RoundUp(Summed\ Packet\ Buffered), 1)]]^2 - 1$$

2. To calculate the play out delay based on the obtained information.

$$Playout\ Delay(seconds) = \frac{Packet\ Buffered_{Network\ Path\ Differential} + Packet\ Buffered_{FEC\ Correction}}{Video\ Packet_{rate}}$$

To convert Playout Delay to 27 M*Hz* clock ticks,

$$\boldsymbol{Playout\ Delay\ (ticks)} = Playout\ Delay(seconds) * 27\ 000\ 000\ Hz$$

Follow the computation below to determine the FEC Base Address in the general space and RTP Base Address per channel in the channel space (in **bold**).

Table 3-2: **Look up Table for Size allocated Per Packet in the DDR**

SMPTE 2022-5/6 Packet	Media/FEC
Size allocated per Packet (Bytes)	1472

Based on look up table (Table 3-2) for size allocated per packet, compute size that allocated in the DDR for each channel (RTP).

$$Size\ per\ Channel_{media} = Size\ allocated\ per\ Packet\ (Bytes)_{media} * (Media\ Packet\ Buffer\ Size + 1)$$

3. Media Buffer Base Address for each channel can be set consecutively by,

$$\boldsymbol{Media\ Buffer\ Base\ Address_{Channel\ n}} = Media\ Buffer\ Base\ Address_{Channel\ [n-1]} + Size\ per\ Channel_{media\ [n-1]}$$

4. FEC Buffer Base Address can be set after the media buffer allocation by,

$$\boldsymbol{FEC\ Buffer\ Base\ Address} = Media\ Buffer\ Base\ Address_{Channel\ [last]} + Size\ per\ Channel_{media\ [last]}$$

Finally set FEC Buffer Pool Size (in **bold**) by computing:

For each channel,

$$FEC\ Pool\ Size = Size\ allocated\ per\ Packet\ (Bytes)_{FEC} * Packet\ Buffered_{FEC\ Correction}$$

5. Summing the each pool size,

$$\boldsymbol{FEC\ Buffer\ Pool\ Size} = FEC\ Pool\ Size_{Channel\ [0]} + \cdots + FEC\ Pool\ Size_{Channel\ [last]}$$

AXI Memory Map Bandwidth Requirements

The memory bandwidth is calculated based on maximum input of 10Gbps per link to the SMPTE 2022-5/6 RX including RTP and FEC packet regardless of Channel Number and SDI Format.

The values on the table are based on worst case per port scenario.

Table 3-3: **Receiver AXI-MM Port Bandwidth Consumption**

Port	MaximumBandwidth (Gbps)
M0_AXIMM WR	21.5
M0_AXIMM RD	10.5
M1_AXIMM WR	2.5
M1_AXIMM RD	10.5

Receiver Output Data Behavior

Figure 3-2 illustrates the receiver output data behavior.

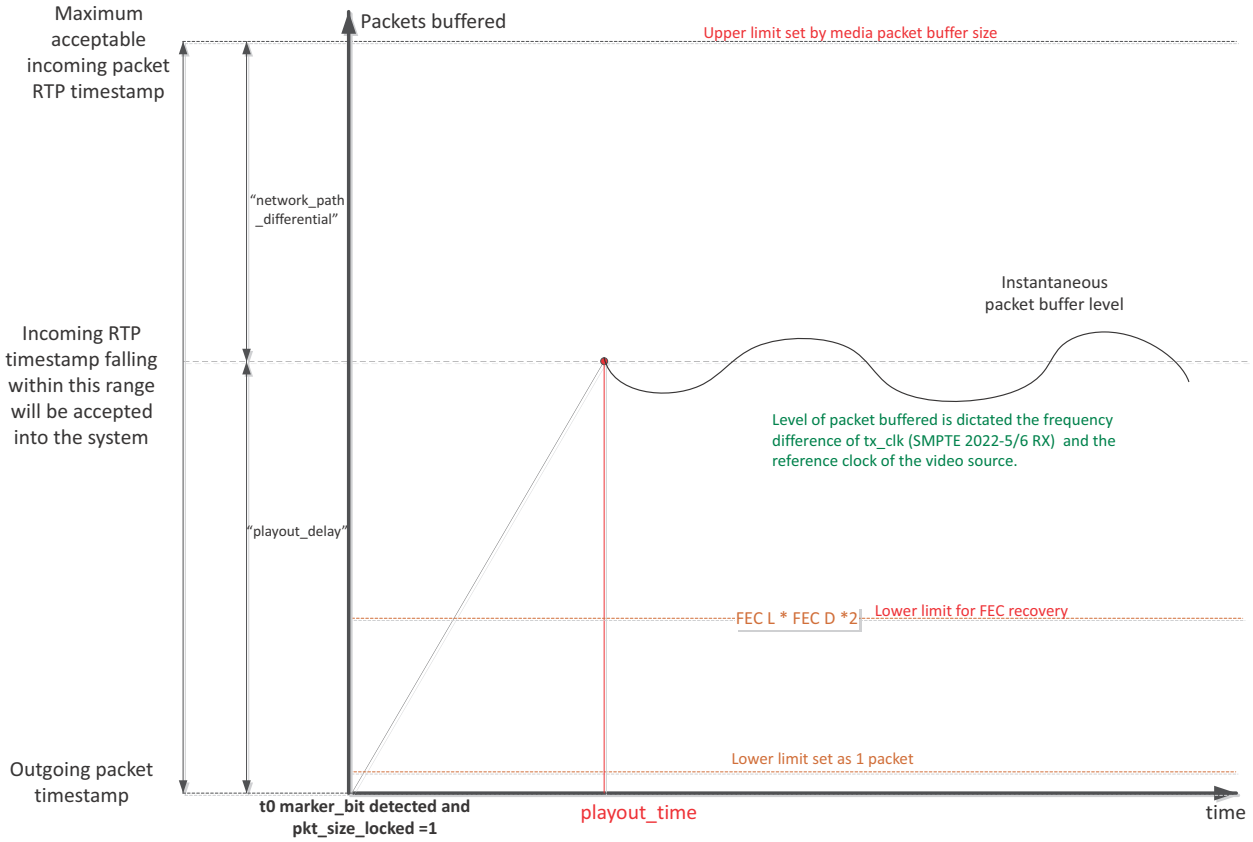

Figure 3-2: **Receiver Output Data Behavior**

Design Flow Steps

This chapter describes customizing and generating the core, constraining the core, and the simulation, synthesis and implementation steps that are specific to this IP core. More detailed information about the standard Vivado® design flows in the IP Integrator can be found in the following Vivado Design Suite user guides:

- *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994) [Ref 11]
- *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 6]
- *Vivado Design Suite User Guide: Getting Started* (UG910) [Ref 8]
- *Vivado Design Suite User Guide: Logic Simulation* (UG900) [Ref 7]

Customizing and Generating the Core

The core is configured to meet the developer's specific needs before instantiation through the Vivado IDE. This section provides a quick reference to parameters that can be configured at generation time.

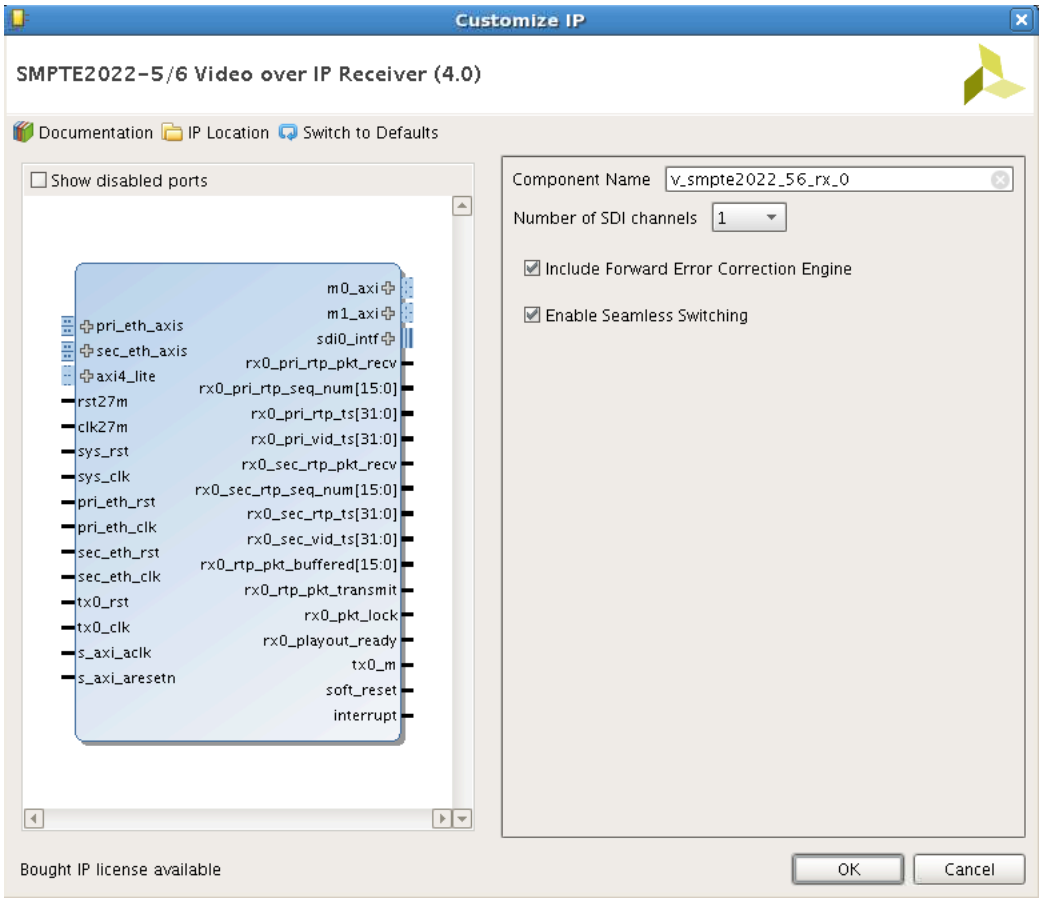

Figure 4-1: **Vivado IDE**

The Vivado IDE displays a representation of the IP symbol on the left side, and the parameter assignments on the right side, which are described as follows:

- **Component Name:** The component name is used as the base name of output files generated for the module. Names must begin with a letter and must be composed from characters: a to z, 0 to 9 and "_". The name v_smpte2022_56_rx cannot be used as a component name.
- **Number of SDI Channels:** Specifies the number of SDI channels.
- **Include Forward Error Correction engine:** When checked, SMPTE 2022-5 Forward Error Correction engine is generated in the core. The core is capable of recovering IP packets lost to network transmission errors.
- **Enable Seamless Switching**: When checked, the core is generated with Secondary AXIS Ethernet Link to support seamless operation.

User Parameters

Table 4-1 shows the relationship between the GUI fields in the Vivado IDE and the User Parameters (which can be viewed in the Tcl console).

Table 4-1: **GUI Parameter to User Parameter Relationship**

GUI Parameter/Value[1]	User Parameter/Value[1]	Default Value
Number of SDI channels	C_CHANNELS	1
include Forward Error Correction Engine	C_INCLUDE_FEC	FALSE
Enable Seamless Switching	C_INCLUDE_HITLESS	FALSE

1. Parameter values are listed in the table where the GUI parameter value differs from the user parameter value. Such values are shown in this table as indented below the associated parameter.

Output Generation

For details, see "Generating IP Output Products" in the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 6].

Constraining the Core

Required Constraints

Constraints required for the core are clock frequency constraints for the clock domains described in Clocking in Chapter 3, Designing with the Core. Paths between the clock domains are constrained with a max_delay constraint and use the datapathonly flag, causing setup and hold checks to be ignored for signals that cross clock domains. These constraints are provided in the XDC constraints file included with the core.

Device, Package, and Speed Grade Selections

There are no device, package or speed grade requirements for this core. This core has not been characterized for use in low-power devices.

Clock Frequencies

See Maximum Frequencies in Chapter 2.

Clock Management

See Clocking in Chapter 3.

Clock Placement

There are no specific clock placement requirements for this core.

Banking

There are no specific Banking rules for this core.

Transceiver Placement

There are no transceiver placement requirements for this core.

I/O Standard and Placement

There are no specific I/O standards and placement requirements for this core.

Simulation

For comprehensive information about Vivado simulation components, as well as information about using supported third-party tools, see the *Vivado Design Suite User Guide: Logic Simulation* (UG900) [Ref 7].

Synthesis and Implementation

For details about synthesis and implementation, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 6].

Test Bench

This chapter contains information about the provided test bench in the Vivado® Design Suite.

Demonstration Test Bench

When the core is generated using the Vivado IP catalog, a demonstration test bench is optionally created. This is a simple SV test bench that exercises the core. The demonstration test bench source code created from mixed verilog/vhdl and systemverilog files under demo_tb/ directory in the Vivado Design Suite output directory. The testbench top file namely as tb_<component_name>.sv

Using the Demonstration Test Bench

The demonstration test bench instantiates the generated SMPTE2022-56-RX core. Either the behavioral model or the netlist can be simulated within the demonstration test bench. Run demonstration test bench using the following steps:

1. Generate the core using IP catalog and set it as the top level.
2. Go to **Simulation Setting** and append prefix "tb_" at the component name stated in the Simulation top module name field. Then click ok.
3. Click **Run Simulation** to start the behavioral simulation.

The test bench generates data for the 3G-A mode by default and simulation will stop once the data stream checker detected output from SMPTE2022-56-RX receiver side. Any mismatch data happen will get displayed by the SDI stream data checker module on the Vivado® IDE console.

Demonstration Test Bench Architecture

Figure 5-1 shows the test bench architecture.

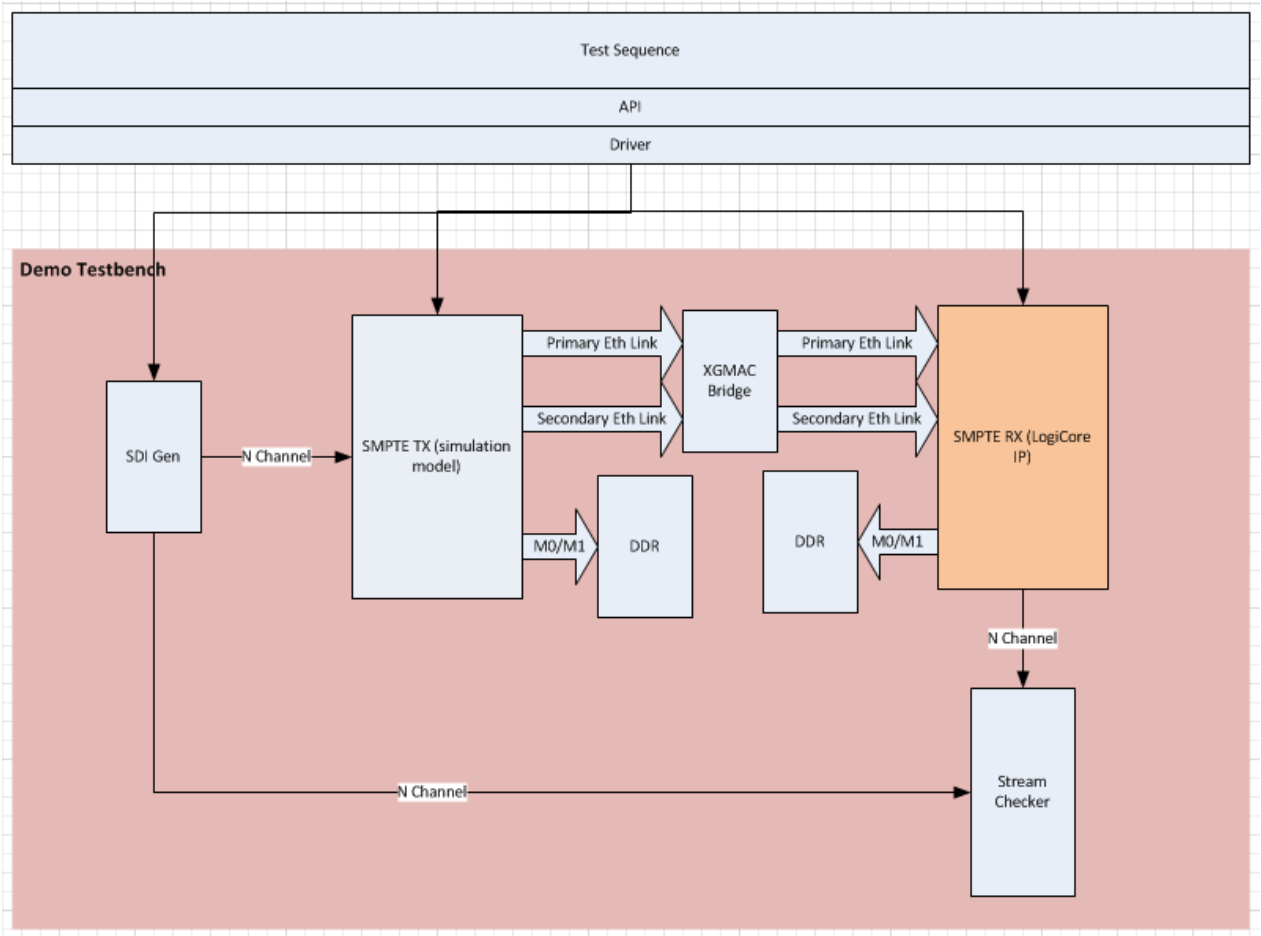

Figure 5-1: **Test Bench Architecture**

The test bench for the SMPTE cores (TX and RX) consists of the major test bench components listed in Table 5-1:

Table 5-1: **Test Bench Components and Descriptions**

Test Bench Components	Description
SDI Video Generator	Provides the video inputs to the SMPTE TX Core based on the SDI format selection (3G/HD/SD) configured by the user via SDI Video Generator Configuration Module setup.
Dummy DDR	Acts as dummy external storage that used by the cores and network emulator during the pkt transaction.
XGMAC Bridge	Acts as a dummy 10G Ethernet MAC Bridge
SDI Stream Data Checker	Compares the raw data output received by the SMPTE RX core with the originally output data from SDI Video Generator. Assert error if there is any data mismatched. Also detects SOF on the input and output streams to signal successful data transmission and reception by the TX and RX core respectively.

Table 5-1: **Test Bench Components and Descriptions *(Cont'd)***

Test Bench Components	Description
Configuration Modules	Can be used to configure the TX, RX, SDI generator and network emulator. Consists of following sub-components: API layer, Driver Layer, HAL layer, AXI4 Lite Master and Slave Decode Logic.
SMPTE TX	Simulation model which is an encrypted version of the VOIP Transceiver core in a loopback mode in the test bench. You cannot view the encrypted model.

XILINX
ALL PROGRAMMABLE™

Appendix A

Verification, Compliance, and Interoperability

The SMPTE 2022-5/6 Video over IP Receiver core has been validated using the Xilinx Kintex®-7 FPGA Connectivity Kit. See *SMPTE 2022-5/6 High Bit Rate Media Transport Over IP Networks with Forward Error Corrections* (XAPP1199) [Ref 1] for more information.

Migrating and Upgrading

This appendix contains information about migrating a design from ISE® to the Vivado® Design Suite, and for upgrading to a more recent version of the IP core. For customers upgrading in the Vivado Design Suite, important details (where applicable) about any port changes and other impact to user logic are included.

Migrating to the Vivado Design Suite

The SMPTE 2022 5/6 Receiver core version v3.0 has been updated to comply with the latest SMPTE 2022 5/6 Standards Specification. Migration from the older ISE version (v2.1) is supported and appropriate warning messages will be displayed during migration process.

For information about migrating to the Vivado Design Suite, see the *ISE to Vivado Design Suite Migration Guide* (UG911) [Ref 9].

Upgrading in the Vivado Design Suite

SMPTE 2022-5/6 Video Over IP Receiver version 4.0 had the following changes implemented, and may not be compatible with previous versions cores (v2.0, v2.1 and v3.0).

Parameter Changes

Table B-1 shows the details of changes involved.

Table B-1: **Parameter Changes**

Version		Note
v2.0/1 and v3.0	**v4.0**	
Component Name	Component Name	Unchanged
Number of SDI channels	Number of SDI channels	Unchanged

Table B-1: **Parameter Changes *(Cont'd)***

Version		Note
v2.0/1 and v3.0	**v4.0**	
Include FEC Engine	Include FEC Engine	Unchanged
	Enable Seamless Switching	New to version v4.0. Refer Customizing and Generating the Core for description.

Port Changes

There are ports changes between older core versions (v2.0/2.1 and v3.0) compare to v4.0. Table B-2 shows the details of changes involved.

Table B-2: **Port Changes**

Version		Note
v2.0/1 and v3.0	**v4.0**	
	rst27m	Newly added
	clk27m	Newly added
eth_rst	pri_eth_rst	Renamed
eth_clk	pri_eth_clk	Renamed
	sec_eth_rst	Newly added for seamless switching purpose
	sec_eth_clk	Newly added for seamless switching purpose
s_axis_aresetn		Removed
s_axis_tdata	pri_s_axis_tdata	Renamed
s_axis_tkeep	pri_s_axis_tkeep	Renamed
s_axis_tvalid	pri_s_axis_tvalid	Renamed
s_axis_tuser	pri_s_axis_tuser	Renamed
s_axis_tlast	pri_s_axis_tlast	Renamed
	sec_s_axis_tdata	Newly added for seamless switching purpose
	sec_s_axis_tkeep	Newly added for seamless switching purpose
	sec_s_axis_tvalid	Newly added for seamless switching purpose
	sec_s_axis_tuser	Newly added for seamless switching purpose
	sec_s_axis_tlast	Newly added for seamless switching purpose
m2_axi_*		Removed
rx[0-7]_rtp_pkt_recv	rx[0-7]_pri_rtp_pkt_recv	Renamed
rx[0-7]_rtp_seq_num	rx[0-7]_pri_rtp_seq_num	Renamed
rx[0-7]_rtp_vid_ts	rx[0-7]_pri_vid_ts	Renamed
rx[0-7]_rtp_ts	rx[0-7]_pri_rtp_ts	Renamed

Functionality Changes

Added Seamless Switching support, which can be enabled thru XGUI option. Refer to Customizing and Generating the Core in Chapter 4 for more information.

Instructions for Minimum Change Migration

SMPTE 2022-5/6 RX Migration from v3.0 to v4.0

Port Changes

In SMPTE 2022-5/6 RX v4.0, a new feature was added which supports SMPTE 2022-7, where the receiver core receives two identical/seamless AXI-Streams (Primary and Secondary) with different header as configured by the user. The port changes related to the new feature are shown in Table B-3.

The secondary AXI-Stream Slave Interface and Ethernet Packets Received Interface signal is visible only when Seamless Switching feature is enabled in the core setting. By disabling the Seamless Switching, only primary AXI-Stream is visible and is similar to SMPTE 2022-5/6 RX v3.0 core which uses single Ethernet Link.

*prefix pri_ is for Primary and sec_ for Secondary

*prefix N is indicating channel number

Table B-3: **Port Changes as supporting SMPTE2022-7 Feature**

SMPTE 2022-5/6 TX v3.0 Ports	SMPTE 2022-5/6 TX v4.0 Ports
s_axis_aresetn	pri_s_axis_aresetn
	sec_s_axis_aresetn
s_axis_tdata[63:0]	pri_s_axis_tdata[63:0]
	sec_s_axis_tdata[63:0]
s_axis_tkeep[7:0]	pri_s_axis_tkeep[7:0]
	sec_s_axis_tkeep[7:0]
s_axis_tvalid	pri_s_axis_tvalid
	sec_s_axis_tvalid
s_axis_tlast	pri_s_axis_tlast
	sec_s_axis_tlast
s_axis_tready	pri_s_axis_tready
	sec_s_axis_tready
rxN_rtp_pkt_recv	rxN_pri_rtp_pkt_recv
	rxN_sec_rtp_pkt_recv

Table B-3: **Port Changes as supporting SMPTE2022-7 Feature *(Cont'd)***

SMPTE 2022-5/6 TX v3.0 Ports	SMPTE 2022-5/6 TX v4.0 Ports
rxN_rtp_seq_num	rxN_pri_rtp_seq_num
	rxN_sec_rtp_seq_num
rxN_rtp_vid_ts	rxN_pri_rtp_vid_ts
	rxN_sec_rtp_vid_ts
rxN_rtp_rtp_ts	rxN_pri_tp_rtp_ts
	rxN_sec_rtp_rtp_ts

For minimum change migration, disable the Seamless Switching Feature and remap the renamed ports accordingly by referring to Table B-2. Refer Port Descriptions in Chapter 2 for more details on ports added in v4.0.

Register Setting

New registers to take note of which were added to support new functionality and features as shown in below table Table B-4.

These are crucial registers that need to take note during migration.

Table B-4: **Added Register in Register Map**

Address Offset	General/ Channel	Register Name		Register Description
		Bit	Name	
0x028	General	[31:0]	network_path_differential	Setting for maximum accepted delay between 2 streams based on 27MHz clock tick
0x034	General	[31:0]	fec_buf_base_addr	DDR base address to store FEC packets
0x038	General	[31:0]	fec_buf_pool_size	No. of Bytes of memory space to cater for FEC buffer
0x11C	Channel	[31:0]	playout_delay	Set time to SDI playout upon incoming stream packet size lock. Value based on 27MHz clock ticks
0x12C	Channel	[31:0]	media_buf_base_addr	Starting base address in the DDR to store packets for each channels [make sure doesn't overlap]
0x130	Channel	[15:0]	media_pkt_buf_size	Maximum number of packets that can hold in the DDR for each channel

1. Primary and Secondary registers are configured separately.

When configuring these new added register, refer to Memory Requirement and Register Configurations in Chapter 3.

For minimum change migration, disable the Seamless Switching and set the primary registers accordingly. Refer to Core Debug in Appendix C for register settings information.

Debugging

This appendix includes details about resources available on the Xilinx Support website and debugging tools.

TIP: *If the IP generation halts with an error, there might be a license issue. See License Checkers in Chapter 1 for more details.*

Finding Help on Xilinx.com

To help in the design and debug process when using the core, the Xilinx Support web page (www.xilinx.com/support) contains key resources such as product documentation, release notes, answer records, information about known issues, and links for obtaining further product support.

Documentation

This product guide is the main document associated with the core. This guide, along with documentation related to all products that aid in the design process, can be found on the Xilinx Support web page (www.xilinx.com/support) or by using the Xilinx Documentation Navigator.

Download the Xilinx Documentation Navigator from the Design Tools tab on the Downloads page (www.xilinx.com/download). For more information about this tool and the features available, open the online help after installation.

Answer Records

Answer Records include information about commonly encountered problems, helpful information on how to resolve these problems, and any known issues with a Xilinx product. Answer Records are created and maintained daily ensuring that users have access to the most accurate information available.

Answer Records can be located by using the Search Support box on the main Xilinx support web page. To maximize your search results, use proper keywords such as

- Product name
- Tool message(s)
- Summary of the issue encountered

A filter search is available after results are returned to further target the results.

Master Answer Record for the SMPTE 2022-5/6 RX Core

AR 54534.

Contacting Technical Support

Xilinx provides technical support at www.xilinx.com/support for this LogiCORE™ IP product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support of product if implemented in devices that are not defined in the documentation, if customized beyond that allowed in the product documentation, or if changes are made to any section of the design labeled DO NOT MODIFY.

To contact Xilinx Technical Support:

1. Navigate to www.xilinx.com/support.
2. Open a WebCase by selecting the WebCase link located under Additional Resources.

When opening a WebCase, include:

- Target FPGA including package and speed grade.
- All applicable Xilinx Design Tools and simulator software versions.
- Additional files based on the specific issue might also be required. See the relevant sections in this debug guide for guidelines about which files to include with the WebCase.

Note: Access to WebCase is not available in all cases. Please login to the WebCase tool to see your specific support options.

Vivado Lab Tools

Vivado® lab tools inserts logic analyzer and virtual I/O cores directly into your design. Vivado lab tools allows you to set trigger conditions to capture application and integrated block port signals in hardware. Captured signals can then be analyzed. This feature represents the functionality in the Vivado IDE that is used for logic debugging and validation of a design running in Xilinx FPGA devices in hardware.

The Vivado lab tools logic analyzer is used to interact with the logic debug LogiCORE IP cores, including:

- ILA 2.0 (and later versions)
- VIO 2.0 (and later versions)

See *Vivado Design Suite User Guide: Programming and Debugging* (UG908) [Ref 10].

Interface Debug

AXI4-Lite Interfaces

Read from a register that does not have all 0s as a default to verify that the interface is functional. Output `s_axi_arready` asserts when the read address is valid, and output `s_axi_rvalid` asserts when the read data/response is valid. If the interface is unresponsive, ensure that the following conditions are met:

- The `s_axi_aclk and aclk` inputs are connected and toggling.
- The interface is not being held in reset, and `s_axi_areset` is an active-Low reset.
- The interface is enabled, and `s_axi_aclken` is active-High (if used).
- The main core clocks are toggling and that the enables are also asserted.
- If the simulation has been run, verify in simulation and/or a Vivado Lab Tools capture that the waveform is correct for accessing the AXI4-Lite interface.

AXI4-Stream Interfaces

If data is not being transmitted or received, check the following conditions:

- If transmit `<interface_name>_tready` is stuck Low following the `<interface_name>_tvalid` input being asserted, the core cannot send data.
- If the receive `<interface_name>_tvalid` is stuck Low, the core is not receiving data.
- Check that the `aclk` inputs are connected and toggling.
- Check that the AXI4-Stream waveforms are being followed.
- Check core configuration.
- Add appropriate core specific checks.

Core Debug

Crucial Register Settings

1. Ensure bits [31:0] of `network_path_differential` (0x028) and bits [31:0] `playout_delay` (0x11C) registers are set accordingly.
2. Ensure `media_pkt_buf_size` (0x130) is enough based on media buffer size formula showed in Memory Requirement and Register Configurations in Chapter 3.
3. Ensure the `fec_buf_base_addr` (0x034) register and `media_buf_base_addr` (0x12C) register are set accordingly and not overlapping.
4. Ensure the `match_sel` (0x0B4) register is set properly to match the Ethernet Header (SSRC, IP Source, IP Destination, UDP Source, UDP Destination and VLAN TAG ID) which is set in the core and indicates which channel it belongs.

Debug Operation

1. If `pri_recv_pkt_cnt` (0x03C) register and `sec_recv_pkt_cnt` (0x040) register is incrementing, it indicates the core is receiving packets.
2. If `pri_discard_pkt_cnt` (0x04C) register and `sec_discard_pkt_cnt` (0x050) register is incrementing, it indicates the incoming packets is dropped due it doesn't match with the settings of `match_sel` (0x0B4) register.
3. Ensure the bit [0] packet lock of `sdi_pkt_status` (0x114) register is high which indicates the core has locked to a packet.
4. If `link_valid_media_pkt_cnt` (0x0C0) register and `chan_valid_media_pkt_cnt` (0x134) register is incrementing, it indicate valid packet is coming into the core.
5. If `oor_pkts_cnt` (0x160) register is incrementing, it indicates dropped packet due the incoming packet timestamp falls outside the window as set by bits [31:9] of `network_path_differential` (0x028) and bits [31:9] of `playout_delay` (0x11C) registers.
6. Ensure that the `curr_pkts_buffered` (0x140) is not greater than the `media_pkt_buf_size` (0x130) register.

Additional Resources and Legal Notices

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see Xilinx Support.

For a glossary of technical terms used in Xilinx documentation, see the Xilinx Glossary.

References

These documents provide supplemental material useful with this product guide.

1. *SMPTE 2022-5/6 High Bit Rate Media Transport Over IP Networks with Forward Error Correction* (XAPP1199)
2. *AXI Design Reference Guide* (UG761)
3. *LogiCORE IP AXI Interconnect Product Guide* (PG059)
4. *LogiCORE IP 10-Gigabit Ethernet MAC Product Guide* (PG072)
5. *Society of Motion Picture and Television Engineers (SMPTE) SD/HD/3G-SDI 2.0 Product Guide* (PG071)
6. *Vivado Design Suite User Guide: Designing with IP* (UG896)
7. *Vivado Design Suite User Guide - Logic Simulation* (UG900)
8. *Vivado Design Suite User Guide: Getting Started* (UG910)
9. *ISE to Vivado Design Suite Migration Guide* (UG911)
10. *Vivado Design Suite User Guide: Programming and Debugging* (UG908)
11. *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994)

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
10/01/2014	4.0	• Revision number advanced to 4.0 with design architecture improvement. • Updated the demonstration test bench. • Updated GUI screens. • Updated tables in Chapter 2, Product Specification. • Updated memory requirements for the core. • Updated migrating and upgrading section.
10/02/2013	3.0	• Added XDC and module level constraints to core. • Added demonstration test bench. • Changed all signals to lowercase.
03/20/2013	3.0	• Revision number advanced to 3.0 to align with core version number • Updated to core version 3.0 and Vivado Design Suite. • Removed all material related to Virtex-6 devices, ISE Design Suite, CORE Generator™ tools, and UCF. • Updated GUIs. • Updated Table 2-8 and 2-10.
12/18/2012	2.1	• Updated to core version 2.1. • Updated to ISE® design tools 14.4 and Vivado® Design Suite 2012.4. • Updated Debug appendix. • Updated design to support the latest SMPTE 2022-5/6 draft change. • Removed MAC_LOW _ADDR, MAC_HIGH _ADDR, and IP_HOST_ADDR registers. • Updated screen captures in Chapter 4 and Chapter 6.
10/16/2012	2.0.1	Updated with memory requirements for the core.
07/25/2012	2.0	Updated to core version 2.0. Added Vivado Design Suite material and support for Virtex-7 device.
04/24/2012	1.0	Initial Xilinx release.

Please Read: Important Legal Notices

Xilinx's Terms of Sale which can be viewed at http://www.xilinx.com/legal.htm#tos; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx's Terms of Sale which can be viewed at http://www.xilinx.com/legal.htm#tos.