LogiCORE IP SMPTE2022-5/6 Video over IP Receiver v2.0

Product Guide

PG033 October 16, 2012

XILINX®

XILINX®

Table of Contents

SECTION I: SUMMARY

IP Facts

Chapter 1: Overview

Feature Summary 8
Applications 8
Operating System Requirements 8
Licensing and Ordering Information 8

Chapter 2: Product Specification

Standards 9
Performance 9
Resource Utilization 9
Port Descriptions 11
Register Space 18

Chapter 3: Designing with the Core

Clocking 24
Resets 24
Memory Requirement 24

SECTION II: VIVADO DESIGN SUITE

Chapter 4: Customizing and Generating the Core

GUI 26
Output Generation 27

Chapter 5: Constraining the Core

Required Constraints 28
Device, Package, and Speed Grade Selections 28
Clock Frequencies 28
Clock Management 28
Clock Placement 29
Banking 29
Transceiver Placement 29
I/O Standard and Placement 29

SECTION III: ISE DESIGN SUITE

Chapter 6: Customizing and Generating the Core

GUI 31
Parameter Values in the XCO File 32
Output Generation 33

Chapter 7: Constraining the Core

Required Constraints 34
Device, Package, and Speed Grade Selections 34
Clock Frequencies 34
Clock Management 34
Clock Placement 35
Banking 35
Transceiver Placement 35
I/O Standard and Placement 35

Chapter 8: Detailed Example Design

SECTION IV: APPENDICES

Appendix A: Verification, Compliance, and Interoperability

Hardware Testing 38

Appendix B: Migrating

Appendix C: Debugging

Solution Centers 40

Appendix D: Additional Resources

Xilinx Resources 41
References 41
Technical Support 41
Revision History 42
Notice of Disclaimer. 42

SECTION I: SUMMARY

IP Facts

Overview

Product Specification

Designing with the Core

Introduction

The Xilinx LogiCORE™ IP SMPTE2022-5/6 Video over IP Receiver is a module for broadcast applications that requires bridging between SMPTE video connectivity standards (SD/HD/3G-SDI) and 10Gb/s networks. The module is capable of recovering IP packets lost to network transmission errors and ensure the picture quality of uncompressed, high bandwidth professional video. The core is for developing internet protocol-based systems to reduce overall cost in broadcast facility for distribution and routing of audio video data.

Features

- Handle up to 8 channels of SD/HD/3G-SDI streams (3 for the case of 3G-SDI) according to SMPTE2022-6.
- Per stream basis Forward Error Correction (FEC) in accordance to SMPTE2022-5
- Supports Level A and Level B FEC operations
- Supports block-aligned and non block-aligned FEC operations
- Supports Virtual Local Area Network (VLAN)
- AXI4-Stream data interfaces
- AXI4-Lite control interface
- Configurable channel selection based on IP source address, User Datagram Protocol (UDP) destination port, and Real-time Transport Protocol (RTP) Synchronization Source (SSRC) identifier over AXI4-Lite interface
- Supports SD-SDI, HD-SDI, 3G-SDI Level-A, 3G-SDI Level-B single stream and 3G-SDI Level-B dual stream

LogiCORE IP Facts Table	
Core Specifics	
Supported Device Family(1)	Virtex®-7, Kintex™-7, Virtex-6
Supported User Interfaces	AXI4-Lite, AXI4-Stream, AXI4
Resources	See Table 2-1, Table 2-2, Table 2-3 and Table 2-4
Provided with Core	
Design Files	ISE®: NGC netlist Vivado™: Encrypted HDL
Example Design	Not Provided
Test Bench	Not Provided
Constraints File	Not Provided
Simulation Model	VHDL or Verilog Structural
Supported S/W Driver	N/A
Tested Design Flows(2)	
Design Entry	ISE Design Suite 14.2 Vivado Design Suite 2012.2(3)
Simulation	Mentor Graphics ModelSim
Synthesis	Xilinx Synthesis Technology (XST) Vivado Synthesis
Support	
Provided by Xilinx @ www.xilinx.com/support	

Notes:

1. For a complete listing of supported devices, see the release notes for this core.
2. For the supported versions of the tools, see the Xilinx Design Tools: Release Notes Guide.
3. Supports only 7 series devices.

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 SMPTE2022 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 SMPTE2022-5/6 is aimed at contribution networks (for example, between broadcast center and regional studio).

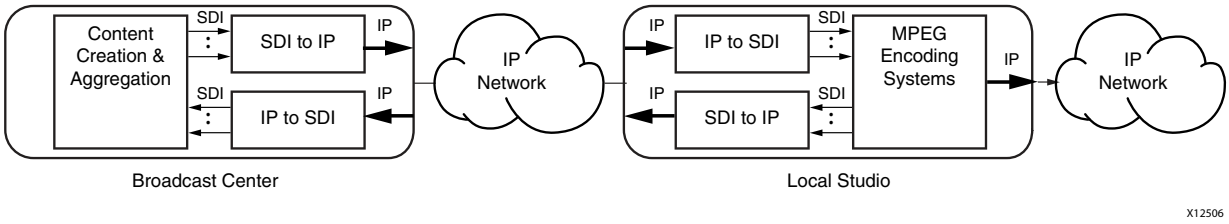

Figure 1-1: **High Bit Rate SMPTE2022-5/6 between Broadcast Center and Local Studio**

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's 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 CORE Generator™ or the Vivado™ tool. 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

Operating System Requirements

For operating system requirements, see the Xilinx Design Tools: Release Notes Guide.

Licensing and Ordering Information

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/ISE Design Suite. 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 SMPTE2022-5/6 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 Reference Guide* (UG761) for additional information. The function of the core is compliant with SMPTE 2022-5/6 working draft.

Performance

The following sections detail the performance characteristics of the core.

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, Table 2-3, and Table 2-4. These values were generated using Xilinx CORE Generator™ Tools, v14.2. They are derived from post-synthesis reports, and might change during MAP and PAR.

Note: Resource numbers for Virtex-7 FPGAs are expected to be similar to those for Kintex-7 FPGAs.

Table 2-1: **Resource Utilization for Kintex-7 Families (FEC not included)**

No. of channels	FFs	LUTs	Slices	BRAM18	BRAM36
1	8121	6499	2331	3	14
3	13709	11016	3573	7	28
6	22019	15061	6685	13	49

Table 2-2: **Resource Utilization for Kintex-7 Families (FEC included)**

No. of channels	FFs	LUTs	Slices	BRAM18	BRAM36
1	11460	8887	3822	7	44
3	18402	14225	5857	15	72
6	28669	20929	9658	33	121

Table 2-3: **Resource Utilization for Virtex-6 Families (FEC not included)**

No. of channels	FFs	LUTs	Slices	BRAM18	BRAM36
1	8127	6036	2799	3	14
3	13715	9996	4864	7	28
6	22019	14451	7274	13	49

Table 2-4: **Resource Utilization for Virtex-6 Families (FEC included)**

No. of channels	FFs	LUTs	Slices	BRAM18	BRAM36
1	11481	8486	4557	7	44
3	18407	13838	6582	15	72
6	28655	20849	10757	33	121

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 GUI.

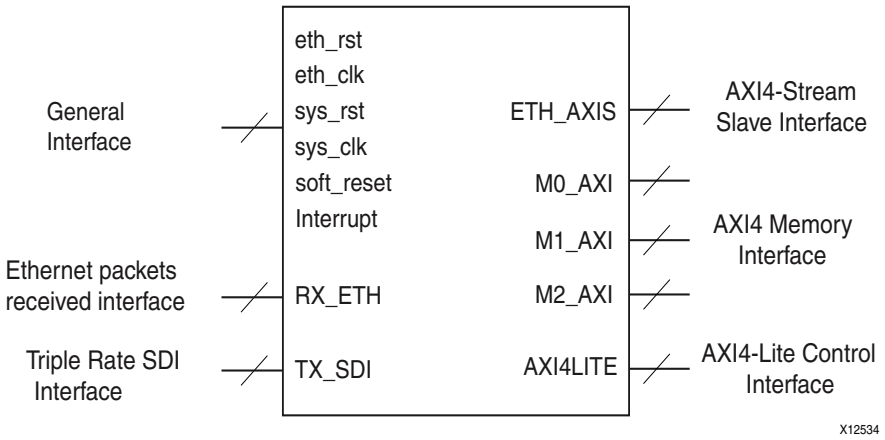

Figure 2-2: **SMPTE2022-5/6 Video over IP Receiver Core Top Level Signaling Interface**

General Interface

Table 2-5 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 2-5: **General Interface Signals**

Signal Name	Direction	Width	Description
eth_rst	In	1	Ethernet domain reset.
eth_clk	In	1	156.25Mhz Ethernet clock.
sys_rst	In	1	System domain reset.
sys_clk	In	1	200MHz system clock.
interrupt	Out	1	Reserved
soft_reset	Out	1	Core reset from the control register

AXI4 Memory 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 IP Product Specification* (DS768) for more information.

Table 2-6: **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 2-6: **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 2-6: **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
m2_axi_awid	Out	1	Write Address Channel Transaction ID
m2_axi_awaddr	Out	32	Write Address Channel Address
m2_axi_awlen	Out	8	Write Address Channel Burst Length code
m2_axi_awsize	Out	3	Write Address Channel Transfer Size code
m2_axi_awburst	Out	2	Write Address Channel Burst Type
m2_axi_awlock	Out	2	Write Address Channel Atomic Access Type
m2_axi_awcache	Out	4	Write Address Channel Cache Characteristics
m2_axi_awprot	Out	3	Write Address Channel Protection Bits
m2_axi_awqos	Out	4	Write Address Channel Quality of Service

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

Signal Name	Direction	Width	Description
m2_axi_awvalid	Out	1	Write Address Channel Valid
m2_axi_awready	In	1	Write Address Channel Ready
m2_axi_wdata	Out	256	Write Data Channel Data.
m2_axi_wstrb	Out	32	Write Data Channel Data Byte Strobes
m2_axi_wlast	Out	1	Write Data Channel Last Data Beat
m2_axi_wvalid	Out	1	Write Data Channel Valid
m2_axi_wready	In	1	Write Data Channel Ready
m2_axi_bid	In	1	Write Response Channel Transaction ID
m2_axi_bresp	In	2	Write Response Channel Response Code
m2_axi_bvalid	In	1	Write Response Channel Valid
m2_axi_bready	Out	1	Write Response Channel Ready

AXI4-Stream Slave Interface

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

Table 2-7: **AXI4-Stream Interface Signals**

Signal Name	Direction	Width	Description
s_axis_aresetn	Out	1	AXI4-Stream active-Low reset for Receive path - 10 Gigabit Ethernet MAC (XGMAC)
s_axis_tdata[63:0]	In	64	AXI4-Stream Data from XGMAC
s_axis_tkeep[7:0]	In	8	AXI4-Stream Data Control from XGMAC
s_axis_tvalid	In	1	AXI4-Stream Data Valid from XGMAC
s_axis_tlast	In	1	AXI4-Stream signal from XGMAC indicating an end of packet
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.

Triple Rate SDI Interface

See the *LogiCORE IP Virtex-6 FPGA Triple-Rate SDI User Guide* (UG823) for more information.

Table 2-8: **Triple Rate SDI Interface Signals**

Signal Name	Direction	Width	Description
tx_rst	In	1	Reset.
tx_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_ce	Out	3	To tx_ce of Triple-Rate SDI
tx_din_rdy	Out	1	To tx_din_rdy of Triple-Rate SDI
tx_ds1a	Out	10	To tx_ds1a of Triple-Rate SDI
tx_ds1b	Out	10	To tx_ds1b of Triple-Rate SDI
tx_ds2a	Out	10	To tx_ds2a of Triple-Rate SDI
tx_ds2b	Out	10	To tx_ds2b of Triple-Rate SDI
tx_level_b_3g	Out	1	To tx_level_b_3g of Triple-Rate SDI
tx_mode	Out	1	To tx_mode of Triple-Rate SDI
tx_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.

Ethernet Packets Received Interface

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

Table 2-9: **Ethernet Packets Received Interface Signals**

Signal Name	Direction	Width	Description
rx_rtp_pkt_recv	Out	1	Pulse indicating receiving of RTP packets
rx_rtp_seq_num	Out	16	Sequence number of RTP packet received
rx_rtp_pkt_buffered	Out	16	Amount of RTP packets buffered
rx_rtp_pkt_transmit	Out	1	Pulse indicating consumption of RTP packet for SDI output
rx_vid_lock	Out	1	Indication of channel locking to certain video payload

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 SMPTE2022-5/6 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* (DS768) for more information.

Table 2-10: **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.

Table 2-10: **AXI4-Lite Interface Signals *(Cont'd)***

Signal Name	Direction	Width	Description
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 SMPTE2022-5/6 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 2-11: **AXI4-Lite Register Map**

Address (hex)	Register Name	Access Type	Default Value	Register Description
General Registers				
0x000	CONTROL	R/W	0	Bit 0: Reserved Bit 1: Register update Bit 31-2: Reserved
0x004	RESET	R/W	0	Bit 0: Soft reset Bit 31-1: Reserved
0x030	CHANNEL	R/W	0	Bit 31-0: Access channel
0x03C	VERSION	R	0x02000000	Bit 7-0: Revision number Bit 11-8: Patch ID Bit 15-12: Version revision Bit 23-16: Version minor Bit 31-24: Version major
0x050	AXI_MM_ADDR_MSB	R/W	0	Bit 2-0: Most significant three bits of the 32-bit AXI memory map address to access the DDR through the AXI interconnect Bit 31-3: Reserved
0x060	MAC_LOW _ADDR	R/W	0	Bit 31-0: Media Access Controller (MAC) address [31:0]
0x064	MAC_HIGH _ADDR	R/W	0	Bit 15-0: MAC address [47:32] Bit 31-16: Reserved
0x068	IP_HOST_ADDR	R/W	0	Bit 31-0: IP address
0x0A0	NUM_CHAN	R	0	Bit 10-0: Number of channels Bit 31-11: Reserved

Table 2-11: **AXI4-Lite Register Map *(Cont'd)***

Address (hex)	Register Name	Access Type	Default Value	Register Description
Channel Registers				
0x100	CHAN_EN	R/W	0	Bit 0: Channel Enable Bit 31-1: Reserved
0x110	FIREWALL_SEL	R/W	0	Bit 1- 0: Select which firewall parameters to be used to filter the Ethernet packets. "00"- dest_port, "01"- dest_port and src_ip, "10"- dest_port and ssrc, "11"- dest_port, src_ip and ssrc" Bit 31-2: Reserved
0x114	DEST_PORT	R/W	0	Bit 15-0: UDP destination port Bit 31-16: Reserved
0x118	SSRC	R/W	0	Bit 31-0: Synchronization Source (SSRC) value
0x11C	SRC_IP_HOST_ADDR	R/W	0	Bit 31-0: Source IP address
0x12C	START_BUFFER_SIZE	R/W	0	Bit 31-0: The number of RTP packets to buffer before starting SDI output
0x144	VID_SRC_FMT	R	0	Bit 31-0: Video source format value
0x148	VID_LOCK_PARAM	R	0	Bit 0: Video locked Bit 31-1: Reserved
0x154	FEC_L	R	0	Bit 9-0: L value Bit 31-10: Reserved
0x158	FEC_D	R	0	Bit 9-0: D value Bit 31-10: Reserved
0x15C	FEC_LOCK_PARAM	R	0	Bit 0: FEC locked Bit 1: FEC protect level. '0' - Level A, '1' - Level B. Bit 31-2: Reserved
0x160	PACKETS_BUFFERED	R	0	Bit 15-0: Number of RTP packets buffered
0x180	SDI_STATUS	R	0	Bit 0: Frame error Bit 31-1: Reserved

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, all the other registers and the core are held at reset state.

CHANNEL (0x030) Register

Set the channel's registers to access. All the channels share the same set of register address in the channel space.

VERSION (0x03C) 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.

MAC_LOW _ADDR (0x060) Register

This register contains the third, fourth, fifth, and sixth byte of the receiver's Ethernet MAC Address. Not used in the design.

MAC_HIGH _ADDR (0x064) Register

This register contains the first byte and second byte of the receiver's MAC Address. Not used in the design.

IP_HOST_ADDR (0x068) Register

This register contains the IP address of the receiver. Not used in the design.

NUM_CHAN (0x0A0) Register

This register indicates the number of channels in the design.

CHAN_EN (0x100) Register

Set high to turn on the channel operation.

FIREWALL_SEL (0x110) Register

Configures the channel to filter the Ethernet packets based on DEST_PORT, SSRC or SRC_IP_HOST_ADDR registers.

DEST_PORT (0x114) Register

Configures the UDP destination port, a parameter that is used to filter the Ethernet packets for the channel.

SSRC (0x118) Register

Configures the Synchronization Source identifier, a parameter that is used to filter the Ethernet packets for the channel.

SRC_IP_HOST_ADDR (0x11C) Register

Configures the source IP address, a parameter that is used to filter the Ethernet packets for the channel.

START_BUFFER_SIZE (0x12C) Register

Configures the latency of the output SDI based on the number of RTP packets to accumulate before starting.

VID_SRC_FMT (0x144) Register

This register contains the video payload identifier of the SDI video format received based on SMPTE 352M standard. It is valid when the video locked bit is high.

VID_LOCKED_PARAM (0x148) Register

Channel is locked to certain video source payload when video locked bit is high.

FEC_L (0x154) Register

This register contains the L value of FEC matrix. It is valid when FEC locked bit is high.

FEC_D (0x158) Register

This register contains the D value of FEC matrix. It is valid when FEC locked bit is high.

FEC_LOCKED_PARAM (0x15C) Register

FEC locked bit high indicates receiver has received FEC packets with certain L and D configuration.

FEC protection level bit indicates if the channel uses one FEC stream (Level A) or two FEC stream (Level B).

PACKETS_BUFFERED (0x160) Register

Read back on the number of packets currently being buffered in the external DDR memory.

SDI_STATUS (0x180) Register

Received incorrect amount of packets per frame when frame error bit is high. Reset the core to ensure proper operation.

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 SMPTE2022-5/6 and maps them in uncompressed SD/HD/3G-SDI streams to the Triple-Rate 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 the XAPP590 *High Bit Rate Media Transport over IP Networks with Forward Error Correction* reference design for more information.

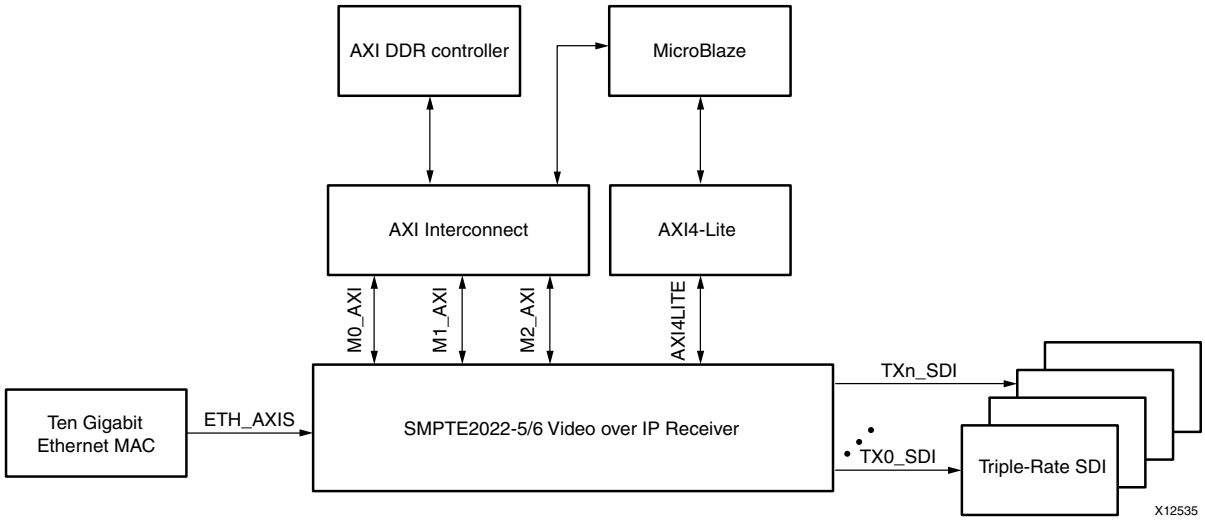

Figure 3-1: **SMPTE2022-5/6 Video over IP Receiver System Built with other Xilinx IP Cores**

Note: There is an option to include Forward Error Correction engine in the SMPTE2022-5/6 Video over IP Receiver core. Adding this will enable 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.

Clocking

The core has three clock domains:

- SDI video clock domain
- System clock domain recommended running at 200 MHz
- Ethernet clock domain at 156.25 MHz for 10Gb/s bandwidth.

Resets

See the XAPP590 *High Bit Rate Media Transport over IP Networks with Forward Error Correction* reference design.

Memory Requirement

Table 3-1 shows tabulation of the amount of DDR memory required by the SMPTE2022-5/6 Video over IP Receiver core based on the number of channels instantiated in the design.

Table 3-1: **Memory Requirement for the SMPTE2022-5/6 Video over IP Receiver Core**

Number of Channels Instantiated	Size of DDR Memory Needed (MB)
1	128
2	256
3	384
4	512
5	320
6	384
7	448
8	512

SECTION II: VIVADO DESIGN SUITE

Customizing and Generating the Core

Constraining the Core

Customizing and Generating the Core

This chapter includes information on using Xilinx tools to customize and generate the core using the Vivado™ Design Suite environment. For more information about the Vivado™ Design Suite, see the Vivado Design Suite - 2012.2 User Guides web page.

GUI

The core is configured to meet the developer's specific needs before instantiation through the Vivado design tools Graphical User Interface (GUI). This section provides a quick reference to parameters that can be configured at generation time.

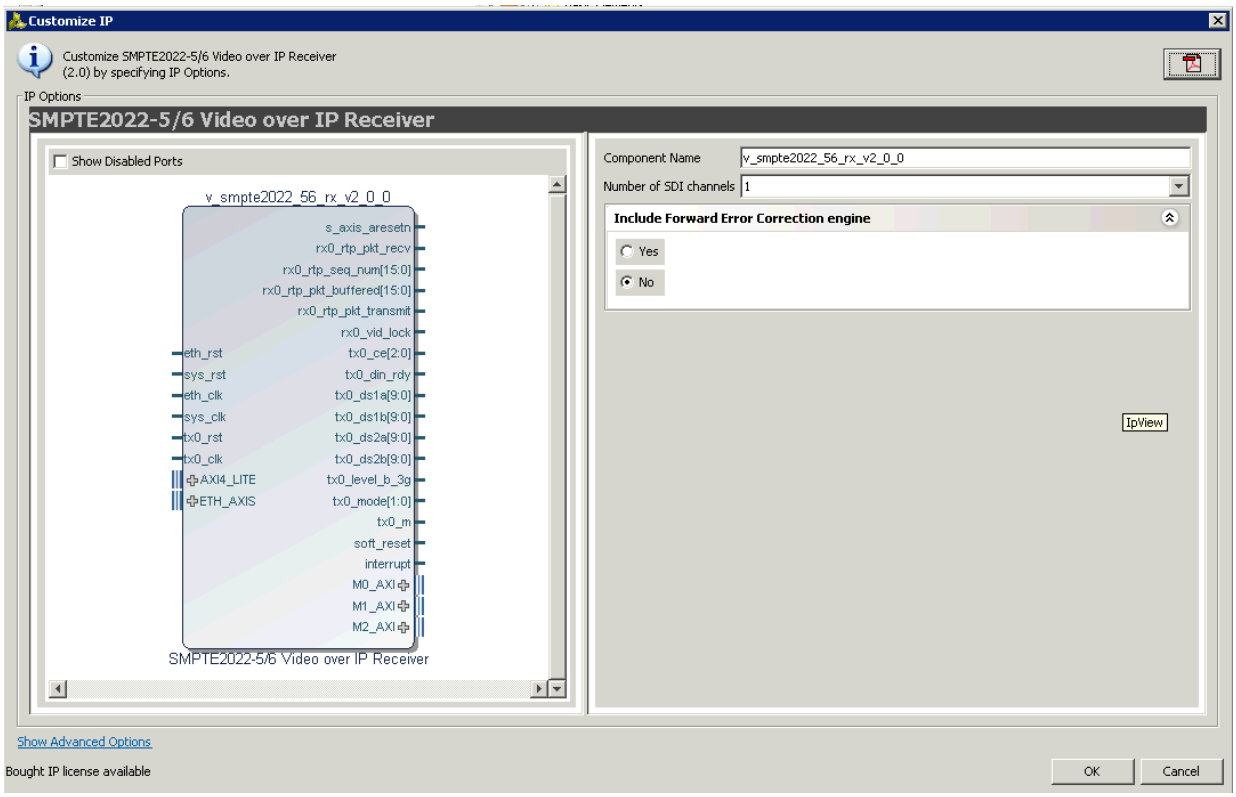

Figure 4-1: **Vivado tools Graphical User Interface**

The GUI 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 Yes is selected, 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.

Output Generation

The Vivado design tools generate the files necessary to build the core and places those files in the <project>/<project>.srcs/sources_1/ip/<core> directory. The Vivado tools output consists of some or all of the following files.

Table 4-1: **File Details**

Name	Description
<component_name>	Library directory for the v_smpte2022_56_rx core that contains the encrypted source files.
<blk_mem_gen_v7_1>	Library directory for the helper core that contains the encrypted source files.
<blk_mem_gen_v7_2>	Library directory for the helper core that contains the encrypted source files.
<fifo_generator_v9_2>	Library directory for the helper core thaT contains the encrypted source files.
<component_name>.vho <component_name>.veo	The HDL template for instantiating the core.
<component_name>.xci	IP-XACT file describing which options were used to generate the core. An XCI file can also be used as a source file for Vivado designs.
<component_name>.xml	IP-XACT XML file describing how the core is constructed so Vivado design tools can properly build the core.

Constraining the Core

This chapter contains information about constraining the core in the Vivado™ Design Suite environment.

Required Constraints

There are no required constraints for this 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

This core has three clock domains.

- SDI clock domain
- System clock domain recommended running at 200 MHz
- Ethernet clock domain at 156.25 MHz.

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.

SECTION III: ISE DESIGN SUITE

Customizing and Generating the Core

Constraining the Core

Detailed Example Design

Customizing and Generating the Core

This chapter includes information on using Xilinx tools to customize and generate the core in the ISE® Design Suite environment.

GUI

The core is configured to meet the developer's specific needs through the CORE Generator™ Graphical User Interface (GUI). This section provides a quick reference to parameters that can be configured at generation time.

Figure 6-1: **Main GUI**

The GUI 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:** Select the number of SDI channels
- I**nclude Forward Error Correction engine:** When **Yes** is selected, 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.

Parameter Values in the XCO File

Table 6-1 defines valid entries for the Xilinx CORE Generator tool (XCO) parameters. Xilinx strongly suggests that XCO parameters are not manually edited in the XCO file; instead, use the CORE Generator system GUI to configure the core and perform range and parameter value checking. The XCO parameters are helpful in defining the interface to other Xilinx tools

Table 6-1: **XCO Parameters**

XCO Parameter	Default	Valid Value
component_name	smpte2022_56_voip_rx	ASCII text using characters: a..z, 0..9 and"_" starting with a letter. ***Note:*** "v_smpte2022_56_rx" is not allowed.
c_sdi_channels	1	1-8
c_include_fec	0	0, 1
c_chan_buf_size	14	13, 14
c_phy	0	0
c_sim_mode	0	0
c_debug_mode	0	0

Output Generation

The Xilinx® CORE Generator tool for the SMPTE2022-5/6 Video over IP Receiver core outputs the core as a netlist that can be instantiated directly in an HDL design. The output is placed in the <project directory>.

Table 6-2: **File Details**

Name	Description
<component_name>_readme.txt	Readme file for the core
<component_name>.ngc	The netlist for the core
<component_name>.vho	The HDL template for instantiating the core
<component_name>.vhd	The structural simulation model for the core. It is used for functionally simulating the core.
<component_name>.xco	Log file from CORE Generator tool describing which options were used to generate the core. An XCO file can also be used as an input to the CORE Generator tool.
<component_name>_flist.txt	A text file listing all of the output files produced when the customized core was generated in the CORE Generator tool.
<component_name>.asy	IP symbol file.
<component_name>.gise <component_name>.xise	ISE® design tools subproject files for use when including the core in ISE designs.

Constraining the Core

This chapter contains information about constraining the core in the ISE® Design Suite environment.

Required Constraints

There are no required constraints for this 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

This core has three clock domains.

- SDI clock domain
- System clock domain recommended running at 200 MHz
- Ethernet clock domain at 156.25 MHz.

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.

Detailed Example Design

No example design is available for the v2.0 core. See the XAPP590 *High Bit Rate Media Transport over IP Networks with Forward Error Correction* reference design for more information.

SECTION IV: APPENDICES

Verification, Compliance, and Interoperability

Migrating

Debugging

Additional Resources

XILINX®

Appendix A

Verification, Compliance, and Interoperability

Hardware Testing

The SMPTE2022-5/6 Video over IP Receiver core has been validated using Xilinx Virtex-6 FPGA Broadcast Connectivity Kit. See the XAPP590 *High Bit Rate Media Transport over IP Networks with Forward Error Correction* reference design for more information.

Appendix B

Migrating

See the *Vivado Design Suite Migration Methodology Guide* (UG911).

For more information about the Vivado Design Suite, see the Vivado Design Suite - 2012.2 User Guides web page.

Appendix C

Debugging

See XAPP590 *High Bit Rate Media Transport over IP Networks with Forward Error Correction* reference design for more information.

Solution Centers

See the Xilinx Solution Centers for support on devices, software tools, and intellectual property at all stages of the design cycle. Topics include design assistance, advisories, and troubleshooting tips.

Appendix D

Additional Resources

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see the Xilinx Support website at www.xilinx.com/support.

For a glossary of technical terms used in Xilinx documentation, see:

www.xilinx.com/company/terms.htm.

References

These documents provide supplemental material useful with this product guide.

1. Vivado™ Design Suite user documentation (www.xilinx.com/cgi-bin/docs/rdoc?v=2012.2;t=vivado+docs)
2. *AXI Reference Guide* (UG761)
3. *LogiCORE IP 10-Gigabit Ethernet MAC Product Guide* (PG072)
4. *LogiCORE IP Virtex-6 FPGA Triple-Rate SDI User Guide* (UG823)
5. *Vivado Design Suite Migration Methodology Guide* (UG911)

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.

See the *IP Release Notes Guide* (XTP025) for more information on this core. For each core, there is a master Answer Record that contains the Release Notes and Known Issues list for the core being used. The following information is listed for each version of the core:

- New Features
- Resolved Issues
- Known Issues

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
04/24/12	1.0	Initial Xilinx release.
07/25/12	2.0	Updated to core version 2.0. Added Vivado Design Suite material and support for Virtex-7 device.
10/16/12	2.0.1	Updated with memory requirements for the core.

Notice of Disclaimer