LogiCORE IP SMPTE 2022-1/2 Video over IP Transmitter v1.0

Product Guide for Vivado Design Suite

PG180 October 2, 2013

XILINX®

XILINX

Table of Contents

Appendix A: Verification, Compliance, and Interoperability

Appendix B: Debugging

Appendix C: Additional Resources

Introduction

The Xilinx LogiCORE™ IP SMPTE 2022-1/2 Video over IP Transmitter is a module for broadcast applications that require bridging between transport stream packets and 1 Gbps IP networks. The core maps the transport stream packets to IP packets and generates forward error correction packets by adding systematically generated redundant data. This allows the receiver to correct a limited number of packet errors without requesting the transmitter to retransmit lost packets. This core is used to develop IP-based systems that reduce the overall cost of distributing and routing audio and video data.

Features

- Transport stream packets encapsulation from up to 16 inputs as defined in SMPTE 2022-2
- Per-stream basis forward error correction as defined in SMPTE 2022-1
- 1-7 transport stream packets per IP packet, and transport stream packet length of 188/204 bytes
- Level A and Level B FEC operations support
- Block-aligned and non block-aligned FEC operations support
- Dynamic switching of L and D values in FEC matrix over AXI4-Lite interface
- VLAN support
- AXI4-Stream data interfaces
- AXI4-Lite control interface
- User-configurable Ethernet , IP, UDP, and RTP headers over AXI4-Lite interface

LogiCORE IP Facts Table	
Core Specifics	
Supported Device Family(1)	Zynq-7000®, Virtex-7®, Kintex-7®, Artix-7®
Supported User Interfaces	AXI4-Lite, AXI4-Stream, AXI-4
Resources	See Table 2-1 through Table 2-4
Provided with Core	
Design Files	Encrypted HDL
Example Design	Not Provided
Test Bench	Not Provided
Constraints File	Not Provided
Simulation Model	Encrypted RTL
Supported S/W Driver	N/A
Tested Design Flows(2)	
Design Entry	Vivado® Design Suite IP Integrator
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 Vivado IP catalog.
2. For the supported versions of the tools, see the Xilinx Design Tools: Release Notes Guide.

Overview

As broadcast and communications markets converge, broadcasters and telecommunication companies increasingly use IP networks for video stream transport. Xilinx devices bridge the broadcast and the communications industries by providing highly integrated real-time video interfaces that help broadcasters reduce costs and the time it takes to acquire, edit, and produce content.

Now that video can be delivered reliably over Ethernet, broadcasters can replace expensive mobile infrastructures that support outside live broadcasts, as well as enable remote production from existing fixed studios. This dramatically reduces both capital expenditure and operating expenses. As a result, using Ethernet to transmit multiple compressed media streams is a major customer requirement. The industry implements primarily the SMPTE 2022 set of standards to create an open and interoperable way to transmit video over Ethernet, ensure quality of service (QoS), and minimize packet loss.

As shown in Figure 1-1, SMPTE 2022-1/2 transmitter core primarily targets distribution networks where multiple transport streams are carried over 1 Gb/s Ethernet networks. The core includes Forward Error Correction (FEC), which protects transport streams over IP

networks. With FEC, the transmitter adds systematically generated redundant data that allows the receiver to detect and correct a limited number of packet errors.

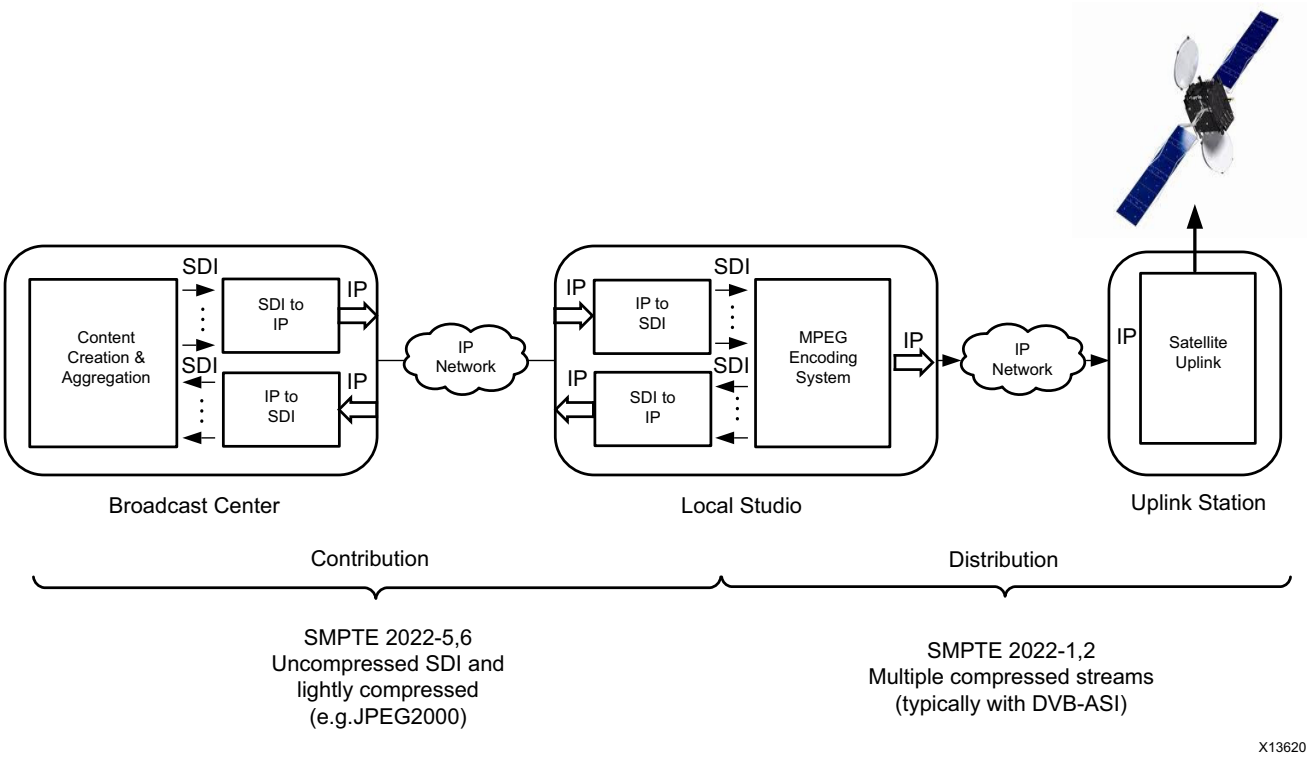

Figure 1-1: **SMPTE 2022-1/2 Deployed in Distribution Networks**

Video packets are lost for a variety of reasons, including thermal noise, storage system defects, and transmission noise introduced by the environment. FEC enables the receiver to correct these errors without using a reverse channel to request retransmission, which is not feasible in real-time systems because the latency is too great.

Feature Summary

The SMPTE 2022-1/2 Video over IP Transmitter core encapsulates transport stream packets into Ethernet packets as defined in SMPTE2022-2. The core generates the Forward Error Correction packets in accordance with SMPTE2022-1 for recovery of IP packets lost to network transmission errors. The core supports VLAN, operating seamlessly when receiving VLAN-tagged Ethernet packets.

You can configure and instantiate the core from the CORE Generator™ tool and control core functionality dynamically through an AXI4-Lite interface.

Applications

The SMPTE 2022-1/2 Video over IP Transmitter core is used to transport compressed constant bit rate video streams over an IP network.

Licensing and Ordering Information

This Vivado® Design Suite IP module is provided under the terms of the Xilinx Core License Agreement. The module is shipped as part of the Vivado 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, see the SMPTE 2022-1/2 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.

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 tools:

- Vivado design tools: Vivado Synthesis, Vivado Implementation, write_bitstream (Tcl command)

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

Product Specification

Standards

The SMPTE 2022-1/2 Video over IP Transmitter 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) [Ref 1] for additional information. The function of the core is compliant with SMPTE 2022-1/2.

Performance

Maximum Frequencies

The maximum achievable clock frequency and all resource counts can be affected by other tool options, additional logic in the FPGA device, different versions of Xilinx tools, and other factors. See Table 2-1 through Table 2-4 for device family–specific information.

Resource Utilization

Resources required for this core have been estimated for Zynq®-7000, Kintex-7®, Virtex-7®, and Artix®-7 devices. These values were generated using Xilinx Vivado® Design Suite. They are derived from post-synthesis reports, and might change during MAP and PAR.

Zynq-7000 Devices

Table 2-1 provides approximate resource counts for the various core options on Zynq-7000 devices.

Table 2-1: **Resource Utilization for Zync-7000 Devices (xc7z045 Speed -1)**

CHANNEL	FEC INCLUDE	HITLESS INCLUDE	FFs	LUTs	Slices	LUT FF Pairs	36k BRAMs	18k BRAMs	DSP48E1s	Fmax (MHz)
1	0	0	7315	3953	2522	7272	12	0	0	219
4	0	0	14411	9661	5187	14459	18	0	0	214
8	0	0	24095	15478	9401	24642	28	0	0	212
16	0	0	43619	27852	13197	42334	44	0	0	204
1	1	0	11254	6940	3806	11020	35	0	0	196
4	1	0	19549	13735	5875	18853	41	0	0	196
8	1	0	30651	20612	10482	30464	51	0	0	196
16	1	0	52845	35726	17105	52833	67	0	0	188
1	0	1	8944	4482	2797	8229	12	0	0	204
4	0	1	17637	10599	6065	16756	18	0	0	204
8	0	1	29453	16671	9714	27209	28	0	0	204
16	0	1	53181	30072	16326	49871	44	0	0	204
1	1	1	12882	7470	4425	12263	35	0	0	212
4	1	1	22776	14707	7728	22138	41	0	0	212
8	1	1	36006	21898	11914	34215	51	0	0	212
16	1	1	62472	37714	18046	57816	67	0	0	188

Virtex-7 FPGAs

Table 2-2 provides approximate resource counts for the various core options on Virtex-7 FPGAs.

Table 2-2: **Resource Utilization for Virtex-7 FPGAs (xcv7vx690t Speed -1)**

CHANNEL	FEC INCLUDE	HITLESS INCLUDE	FFs	LUTs	Slices	LUT FF Pairs	36k BRAMs	18k BRAMs	DSP48E1s	Fmax (MHz)
1	0	0	7315	3953	2440	7033	12	0	0	212
4	0	0	14411	9656	4787	14258	18	0	0	204
8	0	0	24095	15482	7451	23063	28	0	0	180
16	0	0	43619	27846	13464	42992	44	0	0	180
1	1	0	11254	6932	3959	11179	35	0	0	219
4	1	0	19549	13731	6243	19282	41	0	0	219
8	1	0	30651	20607	9255	29275	51	0	0	212

Table 2-2: **Resource Utilization for Virtex-7 FPGAs (xcv7vx690t Speed -1) *(Cont'd)***

CHANNEL	FEC INCLUDE	HITLESS INCLUDE	FFs	LUTs	Slices	LUT FF Pairs	36k BRAMs	18k BRAMs	DSP48E1s	Fmax (MHz)
16	1	0	52845	35735	16330	52178	67	0	0	164
1	0	1	8944	4479	2688	8073	12	0	0	219
4	0	1	17637	10604	6184	16798	18	0	0	219
8	0	1	29453	16679	9174	26742	28	0	0	212
16	0	1	53181	30080	15006	48191	44	0	0	164
1	1	1	12882	7462	4434	12498	35	0	0	212
4	1	1	22776	14704	8573	22734	41	0	0	180
8	1	1	36006	21895	11462	33611	51	0	0	180
16	1	1	62472	37732	19719	58446	67	0	0	180

Kintex-7 FPGAs

Table 2-3 provides approximate resource counts for the various core options on Kintex-7 FPGAs.

Table 2-3: **Resource Utilization for Kirtex-7 FPGAs (xc7k325t Speed -1)**

CHANNEL	FEC INCLUDE	HITLESS INCLUDE	FFs	LUTs	Slices	LUT FF Pairs	36k BRAMs	18k BRAMs	DSP48E1s	Fmax (MHz)
1	0	0	7315	3950	2560	7131	12	0	0	219
4	0	0	14411	9658	5226	14651	18	0	0	219
8	0	0	24095	15476	8161	23835	28	0	0	196
16	0	0	43619	27854	13078	42482	44	0	0	188
1	1	0	11254	6931	3826	11105	35	0	0	212
4	1	0	19549	13730	6770	19830	41	0	0	204
8	1	0	30651	20611	10936	30934	51	0	0	196
16	1	0	52845	35738	17221	53011	67	0	0	180
1	0	1	8944	4480	2879	8268	12	0	0	212
4	0	1	17637	10592	5722	16465	18	0	0	219
8	0	1	29453	16679	9495	27029	28	0	0	212
16	0	1	53181	30083	16070	49262	44	0	0	196
1	1	1	12882	7464	4137	12106	35	0	0	212
4	1	1	22776	14709	7536	21824	41	0	0	204
8	1	1	36006	21886	12060	34192	51	0	0	196
16	1	1	62472	37730	19072	58424	67	0	0	180

Artix-7 FPGAs

Table 2-4 provides approximate resource counts for the various core options on Artix-7 FPGAs.

Table 2-4: **Resource Utilization for Artix-7 FPGAs (xc7a200t Speed -1)**

CHANNEL	FEC INCLUDE	HITLESS INCLUDE	FFs	LUTs	Slices	LUT FF Pairs	36k BRAMs	18k BRAMs	DSP48E1s	Fmax (MHz)
1	0	0	7315	3955	2450	7084	12	0	0	132
4	0	0	14411	9664	5286	14585	18	0	0	132
8	0	0	24095	15439	7591	23076	28	0	0	132
16	0	0	43619	27870	14843	43771	44	0	0	132
1	1	0	11254	6865	3571	10841	35	0	0	132
4	1	0	19549	13758	6959	19810	41	0	0	132
8	1	0	30651	20590	10027	30136	51	0	0	125
16	1	0	52845	35694	17294	53059	67	0	0	125
1	0	1	8944	4491	3025	8497	12	0	0	125
4	0	1	17637	10594	5646	16298	18	0	0	125
8	0	1	29453	16687	9342	26780	28	0	0	125
16	0	1	53181	30076	16749	49886	44	0	0	125
1	1	1	12882	7389	4203	12141	35	0	0	132
4	1	1	22776	14705	7874	22091	41	0	0	132
8	1	1	36006	21860	12263	34213	51	0	0	132
16	1	1	62472	37700	18360	57787	67	0	0	110

The maximum clock frequency results were obtained by double-registering input and output ports to reduce dependence on I/O placement. The inner level of registers used a separate clock signal to measure the path from the input registers to the first output register through the core. The results are post-implementation, using tool default settings except for high effort.

The resource usage results do not include the "characterization" registers and represent the true logic used by the core. LUT counts include SRL16s or SRL32s.

Clock frequency does not take clock jitter into account and should be derated by an amount appropriate to the clock source jitter specification. The maximum achievable clock frequency and the resource counts might also be affected by other tool options, additional logic in the FPGA, using a different version of Xilinx tools, and other factors.

Port Descriptions

The SMPTE 2022-1/2 Video over IP Transmitter 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 2-1 provides an I/O diagram of the core. The S_AXIS transport stream interface pins depend on the number of channels configured via the GUI.

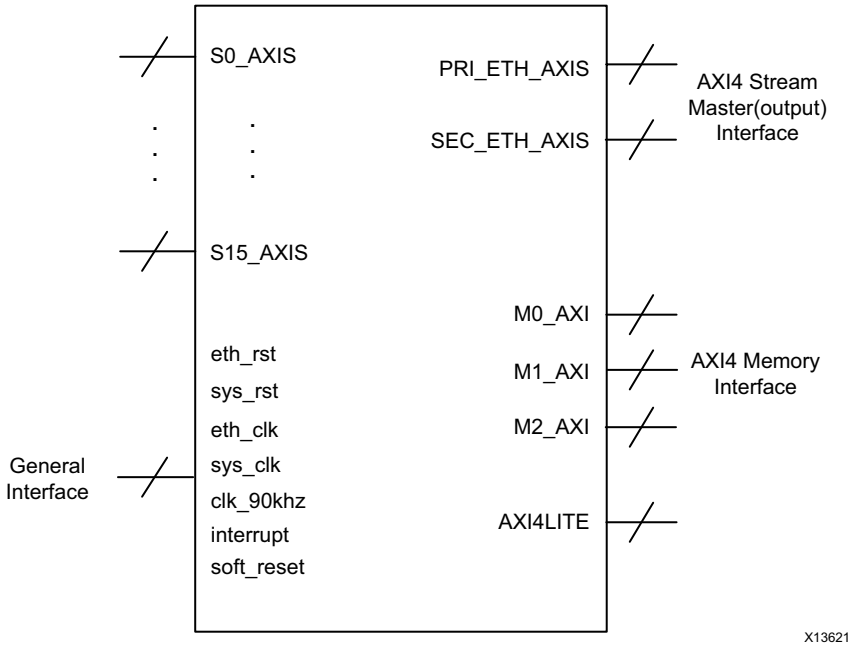

Figure 2-2: **SMPTE 2022-1/2 Video over IP Transmitter Core Top-Level Signaling Interface**

General Interface Signals

Table 2-5 summarizes the signals which are either shared by, or not part of, the AXI4-Stream, AXI-4, or AXI4-Lite control interfaces.

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

Signal Name	Direction	Width	Description
eth_rst	In	1	Ethernet Clock domain reset.
eth_clk	In	1	125 MHz Ethernet clock.
sys_rst	In	1	System clock domain reset.
sys_clk	In	1	System clock.
clk_90khz	In	1	90 KHz RTP timestamp clock.

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

Signal Name	Direction	Width	Description
Interrupt	Out	1	Interrupt from processor.
Soft_reset	Out	1	Reset from processor.

AXI4 Memory Interface

The core uses an AXI4 interface to connect to the AXI4 interconnects. The AXI4 interconnects provide access to the external memory through the AXI DDR controller. See the *LogiCORE IP AXI Interconnect Product Guide* (PG059) [Ref 2] 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_awport	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	128	Write Data Channel Data.
m0_axi_wstrb	Out	16	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_axis_bready	Out	1	Write Response Channel Ready.
m0_axi_arid	Out	1	Read Address Channel Transaction ID.

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

Signal Name	Direction	Width	Description
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.
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	128	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_awport	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	128	Write Data Channel Data.
m1_axi_wstrb	Out	16	Write Data Channel Data Byte Strobes.

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

Signal Name	Direction	Width	Description
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.
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	In	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	128	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_arid	Out	1	Read Address Channel Transaction ID.
m2_axi_araddr	Out	32	Read Address Channel Address
m2_axi_arlen	Out	8	Read Address Channel Burst Length code.
m2_axi_arsize	Out	3	Read Address Channel Transfer Size code.
m2_axi_arburst	Out	2	Read Address Channel Burst Type.
m2_axi_arlock	Out	2	Read Address Channel Atomic Access Type.

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

Signal Name	Direction	Width	Description
m2_axi_arcache	Out	4	Read Address Channel Cache Characteristics.
m2_axi_arprot	Out	3	Read Address Channel Protection Bits.
m2_axi_arqos	Out	4	Read Address Channel Quality of Service.
m2_axi_arvalid	In	1	Read Address Channel Valid.
m2_axi_arready	In	1	Read Address Channel Ready.
m2_axi_rid	In	1	Read Data Channel Transaction ID.
m2_axi_rdata	In	128	Read Data Channel Data.
m2_axi_rresp	In	2	Read Data Channel Response Code.
m2_axi_rlast	In	1	Read Data Channel Last Data Beat.
m2_axi_rvalid	In	1	Read Data Channel Valid.
m2_axi_rready	Out	1	Read Data Channel Ready.

AXI4 Stream Master Interface: Transmit

See the *LogiCORE Tri-Mode Ethernet MAC Product Guide* (PG051) [Ref 3] for more information.

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

Signal Name	Direction	Width	Description
pri_tx_axis_aresetn	Out	1	AXI4-Stream Active-Low reset for Transmit path TEMAC.
pri_tx_axis_tdata[7:0]	Out	8	AXI4-Stream Data to TEMAC.
pri_tx_axis_tvalid	Out	1	AXI4-Stream Data Valid input to TEMAC.
pri_tx_axis_tlast	Out	1	AXI4-Stream last Data input to TEMAC.
pri_tx_axis_tready	In	1	AXI4-Stream acknowledges signals from TEMAC to indicate to start the data transfer.
sec_tx_axis_aresetn	Out	1	AXI4-Stream Active-Low reset for Transmit path TEMAC.
sec_tx_axis_tdata[7:0]	Out	8	AXI4-Stream Data to TEMAC.
sec_tx_axis_tvalid	Out	1	AXI4-Stream Data Valid input to TEMAC.
sec_tx_axis_tlast	Out	1	AXI4-Stream last Data input to TEMAC.
sec_tx_axis_tready	In	1	AXI4-Stream acknowledges signals from TEMAC to indicate to start the data transfer.

AXIS Transport Stream Interface

Table 2-8: **AXIS Transport Stream Interface Signals**

Signal Name	Direction	Width	Description
s_axis_aresetn	In	1	AXI4-Stream Active-Low reset.
s_axis_clk	In	1	AXI-4 Stream clock input to core.
s_axis_data	In	8	AXI-4 Stream data input to core.
s_axis_tvalid	In	1	AXI4-Stream data valid input to core.
s_axis_tuser	In	1	AXI4-Stream tuser input to core indicating the hex 47 synchronizing byte.
s_axis_tlast	In	1	AXI4-Stream data last input to core. Set to 0.
s_axis_tready	out	1	AXI4-Stream acknowledges signals from core. Always 1.

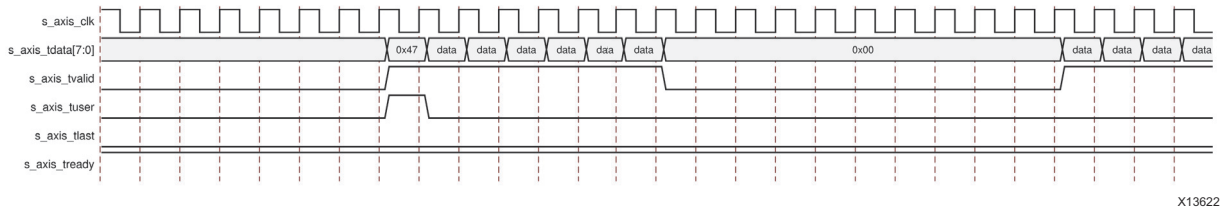

Figure 2-3: **SMPTE 2022-1/2 Video over IP Transmitter Core Transport Stream AXIS Interface**

AXI4-Lite Interface

The AXI4-Lite interface allows user to dynamically control parameters within the SMPTE 2022-1/2 Video over IP Transmitter core. You can configure the core using an embedded ARM or soft system processor such as MicroBlaze.

You can control the core through the AXI4-Lite interface using read and write transactions to the SMPTE 2022-1/2 Video over IP Transmitter 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, connecting through the AXI4-Lite interface to an AXI4-Lite master.

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

Signal Name	Direction	Width	Description
s_axi_clk	In	1	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.

Table 2-9: **AXI4-Lite Interface Signals *(Cont'd)***

Signal Name	Direction	Width	Description
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_awready	Out	1	AXI4-Lite Write Address Channel Write Address Ready. Indicates DMA ready to accept the write address.
s_axi_wready	Out	1	AXI4-Lite Write Data Channel Write Data Ready. Indicates DMA 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_bready	In	1	AXI4-Lite Write Response Channel Ready. Indicates target is ready to receive response.
s_axi_arvalid	In	1	AXI4-Lite Read Address Channel Read Address Valid
s_axi_arready	Out	1	Ready. Indicates DMA is ready to accept the read address.
s_axi_araddr	In	9	AXI4-Lite Read Address Bus
s_axi_rready	In	1	AXI4-Lite Read Data Channel Read Data Ready. Indicates target is ready to accept the read data.
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

Register Space

The SMPTE 2022-1/2 Video over IP Transmitter register space is partitioned to general and channel-specific registers.

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

Address (hex) BASEADDR +	Register Name	Access Type	Default Value	Register Description
General Registers				
0x0000	CONTROL	R/W	0x00000000	Bit 0: SW_ENABLE Bit 1: REG_UPFDATE 32-2: Reserved

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

Address (hex) BASEADDR +	Register Name	Access Type	Default Value	Register Description
0x0004	RESET	R/W	0x00000000	Bit 0: RESET 31-1: Reserved
0x000C	CHANNEL_ACCESS	R/W	0x00000000	Bit 31: 0-primary, 1-secondary 30-8: Reserved 7-0: The channel number to access its register.
0x0010	PRI_MAC_ADDR_LOW	R/W	0x00000000	31-0: Primary Link Source MAC address [31:0]
0x0014	PRI_MAC_ADDR_HIGH	R/W	0x00000000	15-0: Primary Link Source MAC address [47:32] 31-16: Reserved
0x0018	SEC_MAC_ADDR_LOW	R/W	0x00000000	31-0: Secondary Link Source MAC address [31:0]
0x001C	SEC_MAC_ADDR_HIGH	R/W	0x00000000	15-0: Secondary Link Source MAC address [47:32] 31-16: Reserved
0x0020	SYS_CONF	R		Bit 31: Hitless supported Bit 30: FEC supported 29-8: reserved 7-0: Numbers of channels supported
0x0024	VERSION	R	0x01000000	31-24: version major 23-16: version minor 15-12: version revision 11-8: patch ID 7-0: revision number
0x0030	HITLESS_CONFIG	R/W	0x00000000	31-1: reserved Bit 0: 0=enable, 1=disable
Channel Registers				
0x0080	IP_HEADER	R/W	0x00000000	7-0: Time To Live (TTL) 15-8: Type of Service (TOS) 31-16: Reserved
0x0084	VLAN_TAG_INFO	R/W	0x00000000	15-0: VLAN tag info 30-16: Reserved Bit 31: 0=disable VLAN, 1=enable VLAN
0x0088	DST_MAC_LOW_ADDR	R/W	0x00000000	31-0: Destination MAC address [31:0]
0x008C	DST_MAC_HIGH_ADDR	R/W	0x00000000	15-0: Destination MAC address [47:32] 31-16: Reserved

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

Address (hex) BASEADDR +	Register Name	Access Type	Default Value	Register Description
0x0090	DST_IP_ADDR	R/W	0x00000000	31-0: Destination IP address
0x00A0	SRC_IP_ADDR	R/W	0x00000000	31-0: Source IP address
0x00B0	SRC_UDP_PORT	R/W	0x00000000	15-0: Source UDP port 31-16: Reserved
0x00B4	DST_UDP_PORT	R/W	0x00000000	15-0: Destination UDP port 31-16: Reserved
0x0100	CHANNEL_ENABLE	R/W	0x00000000	Bit 0: 1–enable, 0–disable 31-1: Reserved
0x0110	TS_CONFIG	R/W	0x00000000	Bit 0: Reserved 3-1: Transport stream packet per IP [1-7] 31-4: Reserved
0x0114	TS_STATUS	R		Bit 0: Sync byte lock Bit 1: Transport stream packet size 0–188, 1–2-4 31-2: Reserved
0x0118	SSRC	R/W	0x00000000	31-0: Synchronization source
0x011C	FEC_CONFIG	R/W	0x00000000	Bit 0: Reserved Bit 1: 1-row FEC enable Bit 2: 1-column FEC enable 31-3: Reserved
0x0120	FEC_OFFSET	R/W	0x00000000	4-0: Offset for non-block align FEC 31-5: Reserved
0x0124	FEC_L	R/W	0x00000000	4-0: FEC L value 31-5: Reserved
0x0128	FEC_D	R/W	0x00000000	4-0: FEC D value 31-5: Reserved
0x012C	FEC_BASE_ADDR	R/W	0x00000000	31-0: DDR base address for FEC packets

CONTROL (0x0000) 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 will get copied over to the active set if and only if register update bit is set. Setting the bit to 0 before updating multiple registers, then setting the bit to 1 when updates are completed ensures all channel space registers are updated simultaneously.

RESET (0x0004) Register

Bit 0 facilitate software reset. When '1' all registers and the core is held at reset.

CHANNEL_ACCESS (0x000C) Register

Bit 31 select between Primary and Secondary links and Bit 7-0 set the channel number to read and write to/from registers in channel space. All the channels share the same set of register address in the channel space.

PRI_MAC_ADDR_LOW (0x0010) Register

This register configures the third, fourth, fifth and sixth bytes of the source Ethernet MAC Address that is inserted into the Primary Link Ethernet header of the packet.

PRI_MAC_ADDR_HIGH (0x0014) Register

This register configures the first and second bytes of the source Ethernet MAC Address that is inserted into the Primary Link Ethernet header of the packet.

SEC_MAC_ADDR_LOW (0x0018) Register

This register configures the third, fourth, fifth and sixth bytes of the source Ethernet MAC Address that is inserted into the Secondary Link Ethernet header of the packet.

SEC_MAC_ADDR_HIGH (0x001C) Register

This register configures the first and second bytes of the source Ethernet MAC Address that is inserted into the Secondary Link Ethernet header of the packet.

SYS_CONF (0x0020) Register

This register indicates the status of the core.

VERSION (0x0024) Register

Bit fields of the Version 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.

HITLESS_CONFIG (0x0020) Register

Bit 0 configures the bit to enable/disable secondary link.

IP_HEADER (0x0080) Register

This register configures the Time to Live (TTL) and Type of Service (TOS) of the packet.

VLAN_TAG_INFO (0x0084) Register

VLAN_TAG_INFO register configures weather the Ethernet packet contains a VLAN and the tag control information to insert into packets. The tag protocol; identifier is set to 0x8100.

DST_MAC_ADDR_LOW (0x0088) Register

This register configures the third, fourth, fifth and sixth bytes of the destination Ethernet MAC Address that is inserted into the Ethernet header of the packet.

DST_MAC_ADDR_HIGH (0x008C) Register

This register configures the first and second bytes of the destination Ethernet MAC Address that is inserted into the Ethernet header of the packet.

DST_IP_ADDR (0x0090) Register

This register configures the destination IP address that is inserted into the IP Header of the packets.

SRC_IP_ADDR (0x00A0) Register

This register configures the source IP address that is inserted into the IP Header of the packets.

SRC_UDP_PORT (0x00B0) Register

This register configures the UDP source port that is inserted into the IP Header of the packets.

DST_UDP_PORT (0x00B0) Register

This register configures the destination UDP port that is inserted into the IP Header of the packets.

CHAN_EN (0x100) Register

Enable the channel to work by setting to '1'.

TS_CONFIG (0x0110) Register

This register configures the transport stream packet per IP packet. The range of transport stream packets per IP packet is from 1 to 7.

TS_STATUS (0x0114) Register

Bit 0 indicates the sync byte lock and bit 1 indicates the size of the transport stream packet.

SSRC (0x0118) Register

This register configures the SSRC value that is inserted to the RTP header of the packet.

FEC_CONFIG (0x011C) Register

This register configures the FEC level. For level B set both Bit 1 and Bit 2. For level A set Bit 2.

FEC_OFFSET (0x0120) Register

The FEC_OFFSET register is for turning on/off the non block-aligned feature of FEC engine.

FEC_L (0x0124) Register

The FEC_L register is for configuring L value of FEC matrix.

Level A FEC $1 \leq L \leq 20$ and

Level B FEC $4 \leq L \leq 20$.

FEC_D (0x0128) Register

The FEC_D register is for configuring D value of FEC matrix.

Both level A and level B FEC $4 \leq D \leq 20$.

$L \times D$ shall be ≤ 100

FEC_BASE_ADDRESS (0x0128) Register

This register configures the FEC base address of the channel.

Designing with the Core

This chapter includes guidelines and additional information to facilitate designing with the core.

General Design Guidelines

The SMPTE 2022-1/2 Video over IP Transmitter core is designed for broadcast applications that require bridging between the transport stream and 1 Gbit Ethernet. The core takes transport stream packets from the AXIS interface, encapsulates the data into an IP packet by adding UDP and RTP headers, generates forward error correction (FEC) packets, and sends them over the AXI4-Stream interface to Ethernet MAC. The core uses the AXI4 interface to transfer data between the core and buffer in external DDR memory. The register interface is compliant with the AXI4-Lite interface.

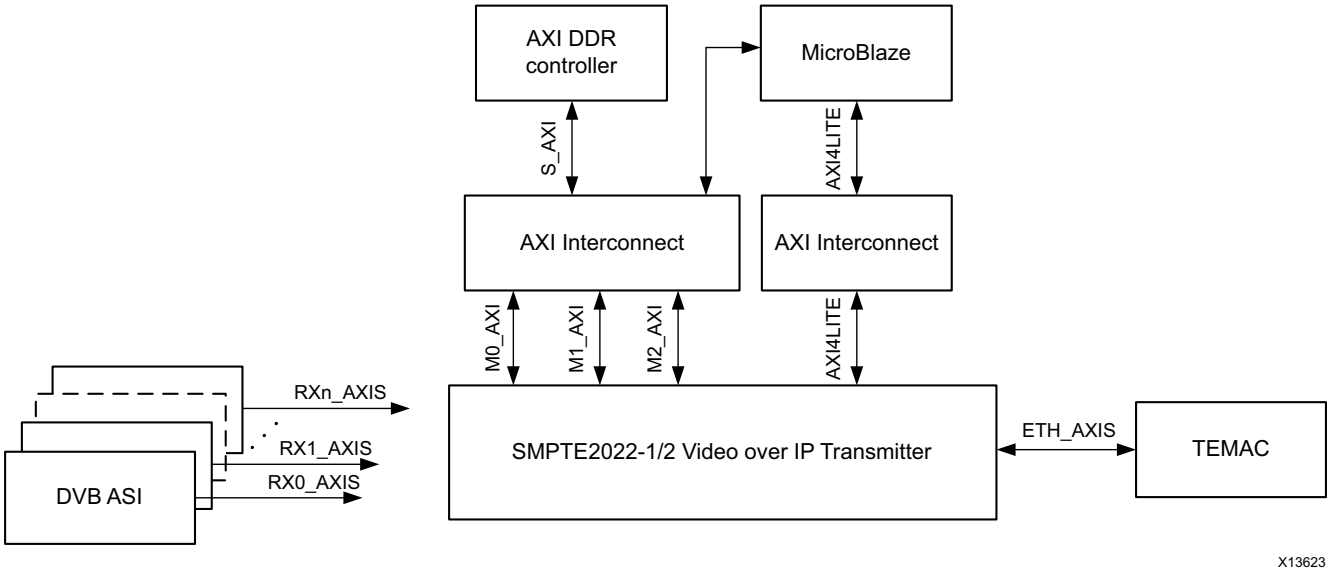

Figure 3-1: **SMPTE 2022-1/2 Video over IP Transmitter System Built with Other Xilinx IP**

Note: The SMPTE 2022-1/2 Video over IP Transmitter core provides two options: include the FEC engine, and enable seamless switching. Enabling the FEC engine allows the transmitter to add systematically generated redundant data so that the receiver can detect and correct a limited number of packet errors to ensure the quality of the compressed video. However, this increases 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. It increases the FPGA resource count.

Clocking

The core has four clock domains:

- Transport stream clock domain, s_axis_clk (recommended 148.5MHz)
- System clock domain, sys_clk (see Fmax data)
- Ethernet clock domain, eth_clk (125MHz)
- AXI4-Lite clock domain, s_axi_aclk (recommended 100MHz)

Resets

The core has three main resets:

- Ethernet reset, eth_rst
- System domain reset, sys_rst
- AXI4-Lite domain reset, s_axi_arestn

The reset must be synchronous to its individual clock domain. A minimum of 16 clock cycles is recommended for the reset assertion. The sys_rst reset must be de-asserted last.

Memory Requirements

The amount of DDR memory required by the SMPTE 2022-1/2 Video over IP Transmitter core is determined by the number and size of transport stream packets in IP packets for each channel.

Table 3-1 shows how to calculate the amount of DDR memory required by the core, when there are 4 channels and the FEC engine is included.

For an SMPTE 2022-2 packet with seven transport stream packets and a transport stream packet size of 188 bytes, the memory required to store one FEC packet is 1408 bytes. The

number of FEC packets to buffer for each channel is [{FEC_L x 2} + 2]. Therefore, memory utilization for each channel would be 59136 bytes..

Table 3-1: **Calculation of Memory Requirement for SMPTE 2022-1/2 Video over IP FEC Packets**

Channel	TS per IP	TS size	TS Packet Size (Bytes)	FEC L	Memory Utilization (Bytes)	Base Address (Hex)
0	7	0	1408	20	59136	00000000
1	7	0	1408	20	59136	0000E700
2	7	0	1408	20	59136	0001CE00
3	7	0	1408	20	59136	0002B500

Customizing and Generating the Core

This chapter includes information about using Xilinx tools to customize and generate the core in the Vivado® Design Suite.

If you are customizing and generating the core in the Vivado IP Integrator, see the *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994) [Ref 11] for detailed information.

Vivado Integrated Design Environment (IDE)

You can customize the IP for use in your design by specifying values for the various parameters associated with the IP core using the following steps:

1. Select the IP from the IP catalog.
2. Double-click the selected IP or select the Customize IP command from the toolbar or popup menu.

For details, see the sections, "Working with IP" and "Customizing IP for the Design" in the *Vivado Design Suite User Guide: Designing with IP* (UG896) and the "Working with the Vivado IDE" section in the *Vivado Design Suite User Guide: Getting Started* (UG910).

Note: Figures in this chapter are illustrations of the Vivado IDE. This layout might vary from the current version.

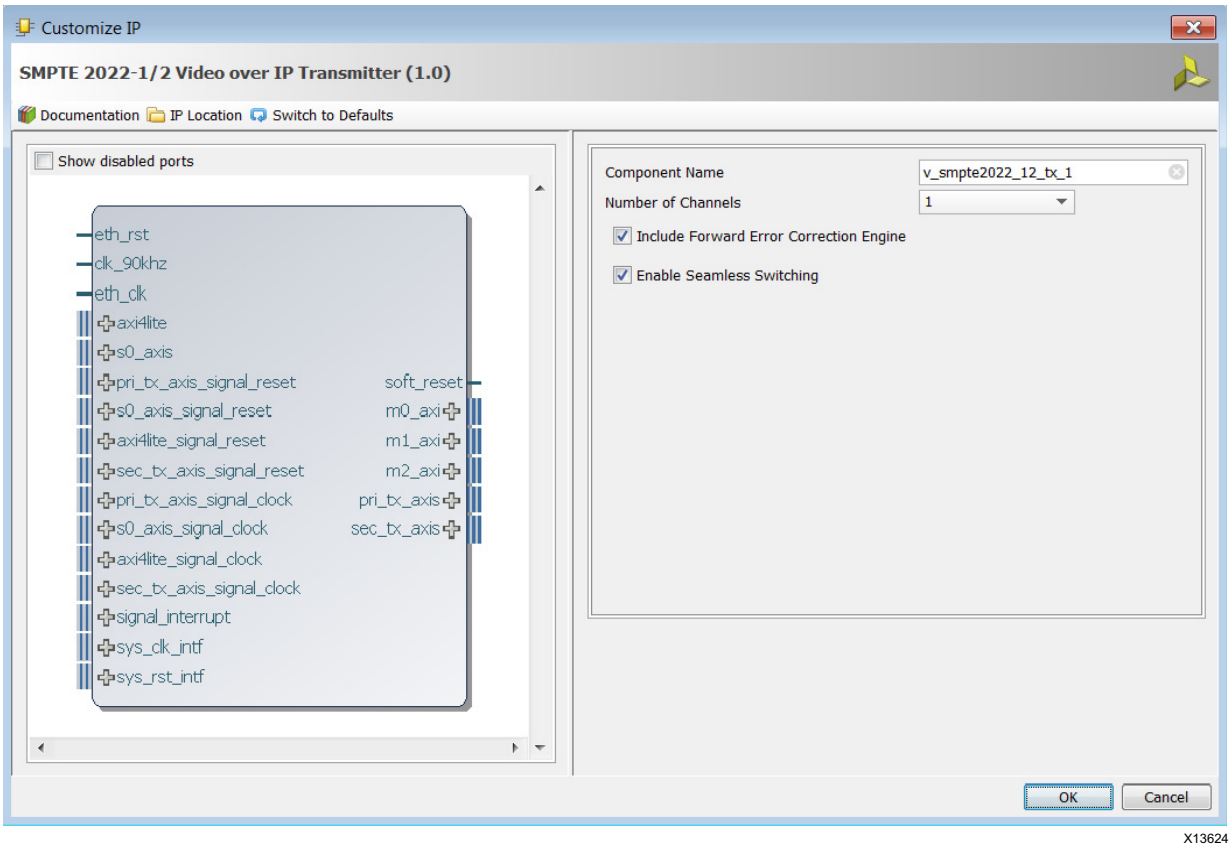

Figure 4-1: **SMPTE 2022-1/2 Video over IP Transmitter Vivado Graphical User Interface**

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

- **Component Name**: The base name of output files generated for the module. Names must begin with a letter and must be composed of characters a to z, 0 to 9 and "_". The name v_smpte2022_12_tx_v1_0 cannot be used as a component name.
- **Number of Channels**: Specifies the number of channels.
- **Include Forward Error Correction Engine**: When checked, the core is generated with FEC.
- **Enable Seamless Switching**: When checked, the core is generated with Secondary AXIS Ethernet Link to support hitless operation.

Output Generation

For details, see "Generating IP Output Products" in the *Vivado Design Suite User Guide: Designing with IP* (UG896). The Vivado design tools generate the files necessary to build the core and place those files in the <*project*>/<*project*>.srcs/sources_1/ip/<*core*> directory.

Constraining the Core

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

Required Constraints

The constraints required for the core are clock frequency constraints described in Clocking in Chapter 3. Paths between two clock domains should be constrained with the 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 Placement

There is no specific clock placement requirement for this core.

Banking

There is no specific banking rule for this core.

I/O Standard and Placement

There is no specific I/O standards and placement requirement for this core.

XILINX

Appendix A

Verification, Compliance, and Interoperability

The SMPTE 2022-1/2 Video over IP Transmitter core has been validated using the Xilinx Kintex®-7 FPGA Broadcast Connectivity Kit.

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 may 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 SMPTE 2022-1/2 Video over IP Transmitter, 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 SMPTE 2022-1/2 Video over IP Transmitter. 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 for this core can also 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

AR: 54533

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 file(s) 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.

Debug Tools

There are many tools available to address SMPTE 2022-1/2 Video over IP Transmitter design issues. It is important to know which tools are useful for debugging various situations.

Vivado Lab Tools

Vivado® lab tools insert 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 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).

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_aready` 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.

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. *AXI Reference Guide* (UG761)
2. *LogiCORE IP AXI Interconnect Product Guide* (PG059)
3. *LogiCORE Tri-Mode Ethernet MAC Product Guide* (PG051)
4. *Vivado Design Suite User Guide - Logic Simulation* (UG900)
5. *Vivado Design Suite User Guide - Implementation* (UG904)
6. *Vivado Design Suite User Guide: Designing with IP* (UG896)
7. *ISE to Vivado Design Suite Migration Guide* (UG911)
8. *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
10/02/2013	1.0	Initial Xilinx release.

Notice of Disclaimer