

SDI II IP Core User Guide



Subscribe



Send Feedback

Last updated for Quartus Prime Design Suite: 16.1

UG-01125
2016.12.20

101 Innovation Drive
San Jose, CA 95134
www.altera.com



Contents

SDI II IP Core Quick Reference.....	1-1
SDI II IP Core Overview.....	2-1
General Description.....	2-1
Resource Utilization.....	2-2
SDI II IP Core Getting Started.....	3-1
Installation and Licensing.....	3-1
Design Walkthrough.....	3-2
Creating a New Quartus Prime Project.....	3-2
Launching IP Catalog.....	3-2
Parameterizing the IP Core.....	3-3
Generating a Design Example and Simulation Testbench.....	3-3
SDI II IP Core Component Files.....	3-4
Compiling the SDI II IP Core Design.....	3-4
Programming an FPGA Device.....	3-4
SDI II IP Core Parameters.....	4-1
SDI II IP Core Functional Description.....	5-1
Protocol.....	5-2
Transmitter.....	5-2
Receiver.....	5-6
Transceiver.....	5-12
Submodules.....	5-14
Insert Line.....	5-14
Insert/Check CRC.....	5-15
Insert Payload ID.....	5-15
Match TRS.....	5-17
Scrambler.....	5-17
TX Sample.....	5-17
Clock Enable Generator.....	5-17
RX Sample.....	5-18
Detect Video Standard.....	5-18
Detect 1 and 1/1.001 Rates.....	5-18
Transceiver Controller.....	5-19
Descrambler.....	5-19
TRS Aligner.....	5-19
3Gb Demux.....	5-20

Extract Line.....	5-20
Extract Payload ID.....	5-21
Detect Format.....	5-21
Sync Streams.....	5-21
Convert SD Bits.....	5-21
Insert Sync Bits.....	5-22
Remove Sync Bits.....	5-23
Optional Features.....	5-23
HD-SDI Dual Link to 3G-SDI (Level B) Conversion.....	5-23
3G-SDI (Level B) to HD-SDI Dual Link Conversion.....	5-25
SMPTE RP168 Switching Support.....	5-27
SD 20-Bit Interface for Dual/Triple Rate.....	5-27
Dynamic TX Clock Switching for Arria V, Cyclone V, and Stratix V Devices.....	5-28
SDI II IP Core Signals.....	6-1
SDI II Core Resets and Clocks.....	6-1
Transmitter Protocol Signals.....	6-9
Image Mapping	6-13
Receiver Protocol Signals.....	6-14
rx_format.....	6-21
Transceiver Signals	6-22
SDI II IP Core Design Considerations.....	7-1
Transceiver Handling Guidelines.....	7-1
Handling Transceiver in Arria V, Cyclone V, and Stratix V Devices.....	7-1
Handling Transceiver in Arria 10 Devices.....	7-4
Timing Violation.....	7-7
Design Example for Arria V, Cyclone V, and Stratix V Devices.....	8-1
Design Example Components.....	8-5
Video Pattern Generator.....	8-5
Transceiver Reconfiguration Controller (Arria V, Cyclone V, and Stratix V Devices).....	8-5
Reconfiguration Management.....	8-6
Reconfiguration Router.....	8-7
Avalon-MM Translators.....	8-8
Design Reference.....	8-8
Video Pattern Generator Signals.....	8-8
Transceiver Reconfiguration Controller Signals.....	8-10
Reconfiguration Management Parameters.....	8-10
Reconfiguration Router Signals.....	8-11
Simulating the SDI II IP Core Design.....	8-12
Simulation Run Time.....	8-13
SDI II IP Core User Guide Archives.....	A-1

Revision History for SDI II IP Core User Guide.....B-1

2016.12.20

UG-01125



Subscribe



Send Feedback

The Altera® Serial Digital Interface (SDI) II MegaCore® function is the next generation SDI intellectual property (IP).

The SDI II IP core is part of the MegaCore IP Library, which is distributed with the Quartus® Prime software and downloadable from the Altera website at www.altera.com.

Note: For system requirements and installation instructions, refer to *Altera Software Installation & Licensing*.

Table 1-1: Brief Information About the SDI II IP Core

Information		Description
Release Information	Version	16.1 (Only available in Quartus Prime Standard Edition)
	Release Date	October 2016
	Ordering Code	IP-SDI-II
IP Core Information	SDI Data Rate Support	<ul style="list-style-type: none"> • 270-Mbps SD-SDI, as defined by <i>SMPTE259M</i> specification • 1.485-Gbps or 1.4835-Gbps HD-SDI, as defined by <i>SMPTE292M</i> specification • 2.97-Gbps or 2.967-Gbps 3G-SDI, as defined by <i>SMPTE424M</i> specification • 5.94-Gbps or 5.934-Gbps 6G-SDI, as defined by <i>SMPTEST2081</i> specification • 11.88-Gbps or 11.868-Gbps 12G-SDI, as defined by <i>SMPTEST2082</i> specification
	Features	<ul style="list-style-type: none"> • Automatic detection of SDI standards and video transport formats • Payload identification packet (ST 352) insertion and extraction • Cyclical redundancy check (CRC) encoding and decoding (except SD) • Line number (LN) insertion and extraction (except SD) • Framing and extraction of video timing signals • Dual link HD-SDI data stream synchronization (except SD)

© 2016 Intel Corporation. All rights reserved. Intel, the Intel logo, Altera, Arria, Cyclone, Enpirion, MAX, NIOS, Quartus and Stratix words and logos are trademarks of Intel Corporation in the US and/or other countries. Other marks and brands may be claimed as the property of others. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO
9001:2008
Registered

ALTERA
now part of Intel

Information	Description
	<ul style="list-style-type: none"> • 3G-SDI with data mapped by ST 425-<i>x</i> mapping • 6G-SDI with data mapped by ST 2081-<i>x</i> mapping • 12G-SDI with data mapped by ST 2082-<i>x</i> mapping • 20-bit interface support for SD-SDI • Dynamic TX clock switching to support fractional and non-fractional video frame rates
Applications	<ul style="list-style-type: none"> • Digital video equipment • Mixing and recording equipment
Device Family Support	Arria® 10, Arria V GX, Arria V GZ, Cyclone® V, and Stratix® V FPGA device families.
Design Tools	<ul style="list-style-type: none"> • IP Catalog in the Quartus Prime software for design creation and compilation • ModelSim®-Altera, Riviera-Pro, NCSIM, and VCS/VCS MX software for design simulation or synthesis

Related Information

- [Altera Software Installation and Licensing](#)
- [SDI II IP Core Design Example User Guide](#)
Provides the design examples for Arria 10 devices.
- [SDI II IP Core User Guide Archives](#) on page 9-1
Provides a list of user guides for previous versions of the SDI II IP core.

2016.12.20

UG-01125



Subscribe



Send Feedback

The SDI II IP core implements a transmitter, receiver, or full-duplex SDI at standard definition (SD), high definition (HD), or 3 gigabits per second (3G) to 12G rate as defined by the Society of Motion Picture and Television Engineers (SMPTE). The SDI II IP core supports dual rates (SD-SDI and HD-SDI), triple rates (SD-SDI, HD-SDI, and 3G-SDI) and multi rates (SD-SDI, HD-SDI, 3G-SDI, 6G-SDI, and 12G-SDI). These modes provide automatic receiver rate detection and transceiver dynamic reconfiguration.

The SDI II IP core highlights the following new features:

- Supports 28 nm devices and beyond.
 - Arria V GX and Stratix V from Quartus II version 12.1 onwards
 - Arria V GZ and Cyclone V from Quartus II version 13.0 onwards
 - Arria 10 from Quartus II version 14.0A10 onwards
- Improved integration with Altera tools (`hw.tcl`).
- IEEE encryption for functional simulation.

General Description

The SMPTE defines a SDI standard that is widely used as an interconnect between equipment in video production facilities. The SDI II IP core can handle the following SDI data rates:

- 270 megabits per second (Mbps) SD-SDI, as defined by *SMPTE259M-1997 10-Bit 4:2:2 Component Serial Digital Interface*
- 1.485 gigabits per second (Gbps) or 1.4835-Gbps HD-SDI, as defined by *SMPTE292M-1998 Bit-Serial Digital Interface for High Definition Television Systems*
- 2.97-Gbps or 2.967-Gbps 3G SDI, as defined by *SMPTE424M*
- 5.94-Gbps or 5.934-Gbps 6G-SDI, as defined by *SMPTEST2081*
- 11.88-Gbps or 11.868-Gbps 12G-SDI, as defined by *SMPTEST2082*
- Dual link HD-SDI, as defined by *SMPTE372M-Dual Link 1.5Gb/s Digital Interface for 1920×1080 and 2048×1080 Picture Formats*
- Dual-rate support for SD-SDI and HD-SDI
- Triple-rate support for SD-SDI, HD-SDI, and 3G-SDI
- Multi-rate support for SD-SDI, HD-SDI, 3G-SDI, 6G-SDI, and 12G-SDI

© 2016 Intel Corporation. All rights reserved. Intel, the Intel logo, Altera, Arria, Cyclone, Enpirion, MAX, NIOS, Quartus and Stratix words and logos are trademarks of Intel Corporation in the US and/or other countries. Other marks and brands may be claimed as the property of others. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO
9001:2008
Registered

ALTERA
now part of Intel

- *SMPTE425M* HD-SDI dual link support (direct source image formatting)
- *SMPTE425M* 3G-SDI (level B) support (dual link mapping)
- *SMPTE RP168* switching support

Table 2-1: SDI II Support

Table below lists the SDI standard support for various FPGA devices.

Device Family	SDI Video Standard						
	Single Rate				Multiple Rates		
	SD-SDI	HD-SDI	3G-SDI	Dual Link HD-SDI	Dual Rate (up to HD)	Triple Rate (up to 3G)	Multi Rate (up to 12G)
Arria V GX	Yes	Yes	Yes	Yes	Yes	Yes	No
Arria V GZ	Yes	Yes	Yes	Yes	Yes	Yes	No
Stratix V	Yes	Yes	Yes	Yes	Yes	Yes	No
Cyclone V	Yes	Yes	Yes	Yes	Yes	Yes	No
Arria 10	No	Yes	Yes	No	No	Yes	Yes

Resource Utilization

The tables below list the typical resource utilization for the SDI II IP core with the Quartus Prime software, version 16.1.

Note: The resource utilization data was obtained by using the most common configurations for each video standard and from one specific family device.

Table 2-2: Resource Utilization for Each Video Standard (Arria V, Cyclone V, and Stratix V Devices)

Standard	ALM Needed	Primary Logic Registers	Secondary Logic Registers	Block Memory Bits
SD-SDI TX	91	147	10	0
SD-SDI RX	498	628	58	60
HD-SDI TX	139	175	17	0
HD-SDI RX	542	854	70	0
HD Dual Link TX	449	518	42	0
HD Dual Link RX	1,247	1,974	181	4,608
3G-SDI TX	440	434	18	0
3G-SDI RX	833	1,338	109	0
Dual Rate TX	248	258	2	0
Dual Rate RX	929	1224	107	0
Triple Rate TX	499	552	5	0

Standard	ALM Needed	Primary Logic Registers	Secondary Logic Registers	Block Memory Bits
Triple Rate RX	1,059	1,600	108	0

Table 2-3: Resource Utilization for Each Video Standard (Arria 10 Devices)

Standard	ALM Needed	Primary Logic Registers	Secondary Logic Registers	Block Memory Bits
HD-SDI TX	108	120	25	0
HD-SDI RX	514	707	153	0
3G-SDI TX	420	362	35	0
3G-SDI RX	799	1,202	212	0
Triple Rate TX	491	500	32	0
Triple Rate RX	1,042	1,443	211	0
Multi Rate (Up to 12G-SDI) TX	1,975	2,169	144	0
Multi Rate (Up to 12G-SDI) RX	4,296	4,982	735	0

2016.12.20

UG-01125



Subscribe



Send Feedback

Installation and Licensing

To evaluate the SDI II IP core using the OpenCore Plus feature, follow these steps in your design flow:

1. Install the SDI II IP core.
2. Create a custom variation of the SDI II IP core.
3. Implement the rest of your design using the design entry method of your choice.
4. Use the IP functional simulation model to verify the operation of your design.
5. Compile your design in the Quartus Prime software. You can also generate an OpenCore Plus time-limited programming file, which you can use to verify the operation of your design in hardware.

Note: For more information on IP functional simulation models, refer to the *Simulating Altera Designs* chapter in volume 3 of the *Quartus Prime Handbook*.

The default installation directory for the SDI II IP core on Windows is `c:\altera\<version>`; on Linux, it is `/opt/altera<version>`.

You can obtain a license for the IP core only when you are completely satisfied with its functionality and performance, and want to take your design to production. After you purchase a license for the SDI II IP core, follow these steps:

1. Set up licensing.
2. Generate a programming file for the Altera device or devices on your board.
3. Program the Altera device or devices with the completed design.

Related Information

- [Introduction to Altera IP Cores](#)
Provides general information about all Altera FPGA IP cores, including parameterizing, generating, upgrading, and simulating IP cores.
- [Creating Version-Independent IP and Qsys Simulation Scripts](#)
Create simulation scripts that do not require manual updates for software or IP version upgrades.
- [Project Management Best Practices](#)
Guidelines for efficient management and portability of your project and IP files.
- [Simulating Altera Designs](#)

© 2016 Intel Corporation. All rights reserved. Intel, the Intel logo, Altera, Arria, Cyclone, Enpirion, MAX, NIOS, Quartus and Stratix words and logos are trademarks of Intel Corporation in the US and/or other countries. Other marks and brands may be claimed as the property of others. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO
9001:2008
Registered

ALTERA
now part of Intel

Design Walkthrough

This walkthrough explains how to create an SDI II IP core design using the Quartus Prime software and IP Catalog. After you generate a custom variation of the SDI II IP core, you can incorporate it into your overall project.

This walkthrough includes the following steps:

1. [Creating a New Quartus Prime Project](#) on page 3-2
2. [Launching IP Catalog](#) on page 3-2
3. [Parameterizing the IP Core](#) on page 3-3
4. [Generating a Design Example and Simulation Testbench](#) on page 3-3
5. [Simulating the SDI II IP Core Design](#) on page 8-12

Creating a New Quartus Prime Project

Before you begin

You need to create a new Quartus Prime project with the **New Project Wizard**, which specifies the working directory for the project, assigns the project name, and designates the name of the top-level design entity.

To create a new project, perform the following the steps.

1. From the Windows Start menu, select **Programs > Altera > Quartus Prime <version>** to run the Quartus Prime software. Alternatively, you can use the Quartus Prime Web Edition software.
2. On the **File** menu, click **New Project Wizard**.
3. In the **New Project Wizard: Directory, Name, Top-Level Entity** page, specify the working directory, project name, and top-level design entity name. Click **Next**.
4. In the **New Project Wizard: Add Files** page, select the existing design files (if any) you want to include in the project.⁽¹⁾ Click **Next**.
5. In the **New Project Wizard: Family & Device Settings** page, select the device family and specific device you want to target for compilation. Click **Next**.
6. In the **EDA Tool Settings** page, select the EDA tools you want to use with the Quartus Prime software to develop your project.
7. The last page in the **New Project Wizard** window shows the summary of your chosen settings. Click **Finish** to complete the Quartus Prime project creation.

Launching IP Catalog

To launch the IP Catalog in the Quartus Prime software, follow these steps:

1. On the **Tools** menu, click **IP Catalog**.
2. Expand the **Interface Protocols > Audio & Video** folder and double-click **SDI II** to launch the parameter editor.

⁽¹⁾ To include existing files, you must specify the directory path to where you installed the SDI II IP core. You must also add the user libraries if you installed the MegaCore IP Library in a different directory from where you installed the Quartus Prime software.

The parameter editor prompts you to specify your IP variation name, optional ports, architecture features, and output file generation options. The parameter editor generates a top-level **.qsys** or **.qip** file representing the IP core in your project.

3. Click **OK** to display the SDI II IP core parameter editor.

Parameterizing the IP Core

To parameterize your IP core, follow these steps:

1. Select the video standard.
2. Select **Bidirectional**, **Transmitter**, or **Receiver** interface direction.
3. Select **Combined Transceiver and Protocol**, **Separate Transceiver** or **Separate Protocol**, (for Arria V, Cyclone V, and Stratix V devices only).
4. Turn on the necessary transceiver options, (for Arria V, Cyclone V, and Stratix V devices only).
5. Turn on the necessary receiver options.

Some options may be grayed out, because they are not supported in the currently selected configuration.

6. Turn on the necessary transmitter options.

Some options may be grayed out, because they are not supported in the currently selected configuration.

7. Select the necessary options in the **Design Example** tab, (if you are generating the design example for Arria 10 devices).
8. Click **Finish**.

Related Information

[SDI II IP Core Parameters](#) on page 4-1

Generating a Design Example and Simulation Testbench

After you have parameterized the IP core, click **Generate Example Design** to create the following entities:

- Design example— serves as a common entity for simulation and hardware verification.
- Simulation testbench—consists of the design example entity and other non-synthesizable components. The example testbench and the automated script are located in:
 - Arria V, Cyclone V, and Stratix V: `<variation name>_example_design/sdi_ii/simulation/verilog`
or `<variation name>_example_design/sdi_ii/simulation/vhdl` directory.
 - Arria 10: `<your design example folder>/simulation` directory.

Note: Generating a design example can increase processing time.

You can now integrate your custom IP core variation into your design, simulate, and compile.

SDI II IP Core Component Files

Table 3-1: Generated Files

Table below describes the generated files and other files that might be in your project directory. The names and types of files vary depending on whether you create your design with VHDL or Verilog HDL.

Extension	Description
<code><variation name>.v</code> or <code>.sv</code>	An IP core variation file, which defines a Verilog HDL description of the custom IP core. Instantiate the entity defined by this file inside your design.
<code><variation name>.sdc</code>	Contains timing constraints for your SDI variation.
<code><variation name>.qip</code>	Contains Quartus Prime project information for your IP core variations. Add this file in your Quartus Prime project before you compile your design in the Quartus Prime software.

Compiling the SDI II IP Core Design

To compile your design, click **Processing** > **Start Compilation** in the Quartus Prime software. Use the generated `.qip` file to include relevant files into your project.

You can find the design examples of the SDI II IP core in:

- Arria V, Cyclone V, and Stratix V: `<variation name>_example_design/sdi_ii/example_design/sdi_ii_0001_ed` directory.
- Arria 10: `<your design example folder>/rtl` directory.

Note: To create a new project using the generated design example, follow the steps in the *Creating a New Quartus Prime Project* section and add the design example `.qip` file in **step 4**.

Related Information

- [Creating a New Quartus Prime Project](#) on page 3-2
- [Design Example for Arria V, Cyclone V, and Stratix V Devices](#) on page 8-1
Provides the design examples for Arria V, Cyclone V, and Stratix V devices.
- [SDI II IP Core Design Example User Guide](#)
Provides the design examples for Arria 10 devices.
- [Quartus Prime Help](#)
More information about compilation in Quartus Prime software.

Programming an FPGA Device

After successfully compiling your design, program the targeted Altera device with the Quartus Prime Programmer and verify the design in hardware.

For instructions on programming the FPGA device, refer to the *Device Programming* section in volume 3 of the Quartus Prime Handbook.

Related Information
Device Programming

2016.12.20

UG-01125



Subscribe



Send Feedback

Table 4-1: SDI II IP Core Parameters

Note: **Transceiver Options** are available only for Arria V, Cyclone V, and Stratix V devices.

Parameter	Value	Description
Configuration Options		
Video standard	SD-SDI, HD-SDI, 3G-SDI, HD-SDI dual link, Dual rate (up to HD-SDI), Triple rate (up to 3G-SDI), Multi rate (up to 12G-SDI)	<p>Sets the video standard.</p> <ul style="list-style-type: none"> SD-SDI—disables option for line insertion and extraction, and CRC generation and extraction HD-SDI—enables option for in line insertion and extraction and CRC generation and extraction Dual-, triple-, or multi-rate SDI—includes the processing blocks for the respective supported rates. Logics for bypass paths and to automatically switch between the input standards are included. <p>Note: SD-SDI, HD-SDI dual link, and Dual rate (up to HD-SDI) options are not available for Arria 10 devices. Multi rate (up to 12G-SDI) option is not available for Arria V, Cyclone V, and Stratix V devices.</p>
SD interface bit width	10, 20	Selects the SD interface bit width. Only applicable for dual rate and triple rate.
Direction	Bidirectional, Receiver, Transmitter	<p>Sets the port direction. The selection enables or disables the receiver and transmitter supporting logic appropriately.</p> <ul style="list-style-type: none"> Bidirectional—instantiates both the SDI transmitter and receiver. Receiver—instantiates the SDI receiver Transmitter—instantiates the SDI transmitter.

© 2016 Intel Corporation. All rights reserved. Intel, the Intel logo, Altera, Arria, Cyclone, Enpirion, MAX, NIOS, Quartus and Stratix words and logos are trademarks of Intel Corporation in the US and/or other countries. Other marks and brands may be claimed as the property of others. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO
9001:2008
Registered

ALTERA
now part of Intel

Parameter	Value	Description
Configuration Options		
Transceiver and/or Protocol	Combined, Transceiver, Protocol	<p>Selects the transceiver or protocol components, or both.</p> <ul style="list-style-type: none"> Transceiver—includes tx/rx_phy_mgmt/phy_adapter and hard transceiver. This option is useful if you want to use the same transceiver component to support both SDI and ASI IP cores. Protocol—allows each submodule to be removed or reused across different video standards. The transmitter and receiver data paths are independent from each other. <p>Note: This option is available only for Arria V, Cyclone V, and Stratix V devices.</p>
Transceiver Options		
Transceiver reference clock frequency	148.5/148.35 MHz, 74.25/74.175 MHz,	<p>Selects the transceiver reference clock frequency.</p> <p>The 74.25/74.175 MHz option is available only for HD-SDI and HD-SDI dual link video standards, and if you select CMU as the TX PLL.</p>
TX PLL type	CMU, ATX	<p>Selects the transmitter PLL for TX or bidirectional ports.</p> <p>ATX PLL is useful for bidirectional channels—you can use the ATX PLL as the transmitter PLL instead of the CMU PLL from another channel.</p>
Dynamic Tx clock switching	Off, Tx PLLs switching, Tx PLL reference clocks switching	<ul style="list-style-type: none"> Off: Disable dynamic switching Tx PLLs switching: Instantiates two PLLs, each with a reference input clock Tx PLL reference clocks switching: Instantiates a PLL with two reference input clocks. <p>Note: This option is not available if you select ATX PLL.</p> <p>Turn on this option to allow dynamic switching between 1 and 1/1.001 data rates.</p> <p>Note: This option is only available for TX or bidirectional ports, and all video standards except SD-SDI.</p>
Receiver Options		
Increase error tolerance level	On, Off	<ul style="list-style-type: none"> On: Error tolerance level = 15 Off: Error tolerance level = 4 <p>Turn on this option to increase the tolerance level for consecutive missed end of active videos (EAVs), start of active videos (SAVs), or erroneous frames.</p>

Receiver Options		
CRC error output	On, Off	<ul style="list-style-type: none"> On: CRC monitoring (Not applicable for SD-SDI mode) Off: No CRC monitoring (saves logic)
Extract Payload ID (SMPTE ST 352M)	On, Off	<ul style="list-style-type: none"> On: Extract payload ID Off: No payload ID extraction (saves logic) <p>It is compulsory to turn on this option for 3G-SDI, HD SDI dual link, triple-rate, and multi-rate modes. The extracted payload ID is required for consistent detection of the 1080p format.</p>
Convert HD-SDI dual link to 3G-SDI (level B)	On, Off	<ul style="list-style-type: none"> On: Converts to level B (2 × SMPTE 292M HD-SDI mapping, including SMPTE 372M dual link mapping) for HD-SDI dual link receiver output. Off: No conversion <p>Note: This option is only available for HD-SDI dual link receiver.</p>
Convert 3G-SDI (level B) to HD-SDI dual link	On, Off	<ul style="list-style-type: none"> On: Converts to HD-SDI dual link (direct image format mapping) for 3G-SDI receiver output. Off: No conversion <p>Note: This option is only available for 3G-SDI receiver.</p>
Transmitter Options		
Insert payload ID (SMPTE ST 352M)	On, Off	<ul style="list-style-type: none"> On: Insert payload ID Off: No payload ID insertion (saves logic)

Table 4-2: SDI II Design Example Parameters

These options are available only for Arria 10 devices.

Parameter	Value	Description
Available Design Example		
Select Design	None, with external VCXO, without external VCXO	<p>Select the design example to be generated.</p> <ul style="list-style-type: none"> None: No design example is available for the current parameter selection Parallel loopback with external VCXO: Parallel loopback design with an external VCXO. Available only if you select Triple rate (up to 3G-SDI) or Multi rate (up to 12G-SDI). The generated design example has preconfigured parameter settings—does not follow user settings. Parallel loopback without external VCXO: Parallel loopback design without an external VCXO. Available only if you select Triple rate (up to 3G-SDI). The generated design example has preconfigured parameter settings—does not follow user settings.
Design Example Options		
Tx PLL type	CMU, fPLL	<p>Sets the transceiver PLL type.</p> <ul style="list-style-type: none"> CMU PLL only supports data rates up to 3G-SDI. fPLL supports all data rates up to 12G-SDI.
Dynamic Tx clocks switching	Off, Tx PLLs switching, Tx PLL reference clocks switching	<ul style="list-style-type: none"> Off: Disable dynamic switching Tx PLLs switching: Instantiates two PLLs, each with a reference input clock Tx PLL reference clocks switching: Instantiates a PLL with two reference input clocks. <p>Turn on this option to allow dynamic switching between 1 and 1/1.001 data rates. This option is only available for TX or bidirectional ports, and all video standards except SD-SDI.</p> <p>Note: The offered design examples do not support the dynamic TX clock switching feature. Therefore, this option is disabled for 16.1 release.</p>
Design Example Files		
Simulation	On, Off	Turn on this option to generate the necessary files for the simulation testbench.
Synthesis	On, Off	Turn on this option to generate the necessary files for Quartus Prime compilation and hardware demonstration.

Generated HDL Format		
Generate File Format	Verilog, VHDL	<p>Select your preferred HDL format for the generated design example fileset.</p> <p>Note: This option only determines the format for the generated top level IP files. All other files (e.g. example testbenches and top level files for hardware demonstration) are in Verilog HDL format.</p>
Target Development Kit		
Select Board	No Development Kit, Arria 10 GX FPGA Development Kit, Custom Development Kit	<p>Select the board for the targeted design example.</p> <ul style="list-style-type: none"> No Development Kit: This option excludes all hardware aspects for the design example. The IP core sets all pin assignments to virtual pins. Arria 10 GX FPGA Development Kit: This option automatically selects the project's target device to match the device on this development kit. You may change the target device using the Change Target Device parameter if your board revision has a different device variant. The IP core sets all pin assignments according to the development kit. Custom Development Kit: This option allows the design example to be tested on a third party development kit with an Altera device. You may need to set the pin assignments on your own.
Target Device		
Change Target Device	On, Off	Turn on this option and select the preferred device variant for the development kit.

2016.12.20

UG-01125



Subscribe



Send Feedback

The SDI II IP core implements a transmitter, receiver, or full-duplex interface.

The SDI II IP core consists of the following components:

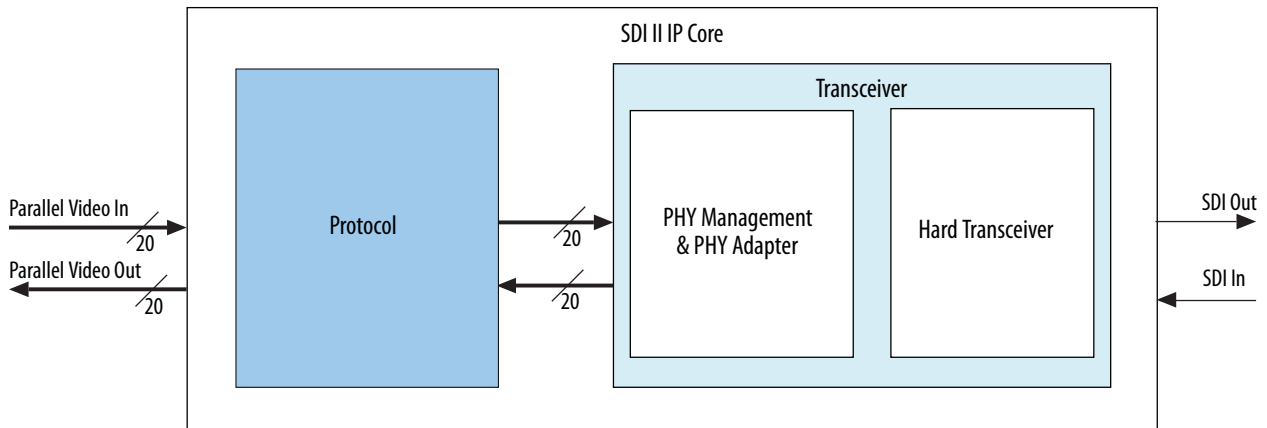
- Protocol block—transmitter or receiver
- Transceiver blocks—PHY management & adapter and hard transceiver

In the parameter editor, you can specify either protocol, transceiver, or combined blocks for your design. For example, if you have multiple protocol blocks in a design, you can multiplex them into one transceiver.

The modular hierarchy design allows you to remove or reuse each submodule within the components across different video standards. The transmitter and receiver data paths are independent from each other.

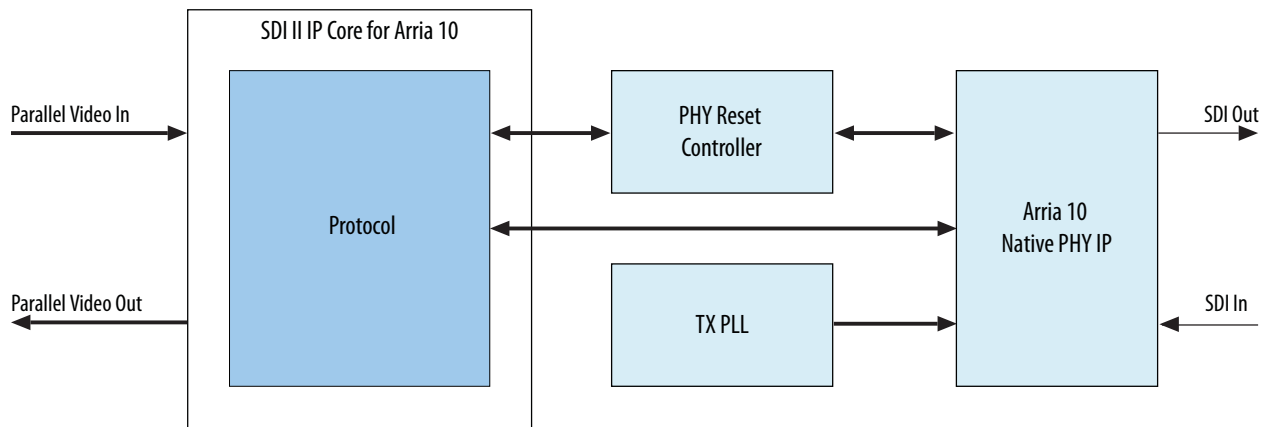
The following figure shows the block diagram for non-Arria 10 devices.

Figure 5-1: SDI II IP Core Block Diagram for Arria V, Cyclone V, and Stratix V Devices



For the Arria 10 devices, the SDI II IP core no longer provides the transceiver, and the TX PLL is no longer wrapped in the transceiver PHY. You must generate the transceiver and the TX PLL separately. The figure below shows the block diagram for Arria 10 devices.

Figure 5-2: SDI II IP Core Block Diagram for Arria 10 Devices



Protocol

The protocol block handles the SDI-specific parts of the core and generally operates on a parallel domain data.

Transmitter

The transmitter performs the following functions:

- HD-SDI LN insertion
- Sync bit insertion
- HD-SDI CRC generation and insertion
- Payload ID insertion
- Matching timing reference signal (TRS) word
- Clock enable signal generation
- Scrambling and non-return-zero inverted (NRZI) coding

The block diagrams below illustrate the SDI II IP core transmitter (simplex) data path for each supported video standard.

For more information about the function of each submodule, refer to the *Submodules* section.

Figure 5-3: SD-SDI Transmitter Data Path Block Diagram

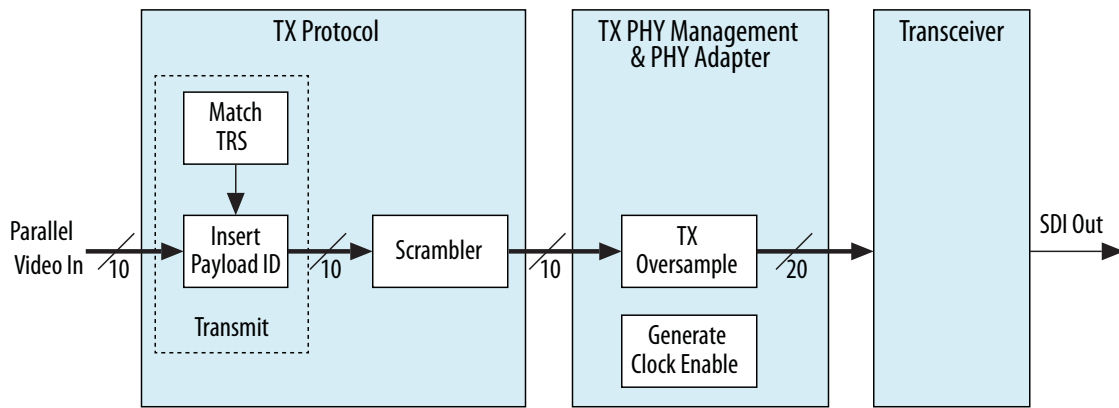


Figure 5-4: HD/3G-SDI Transmitter Data Path Block Diagram

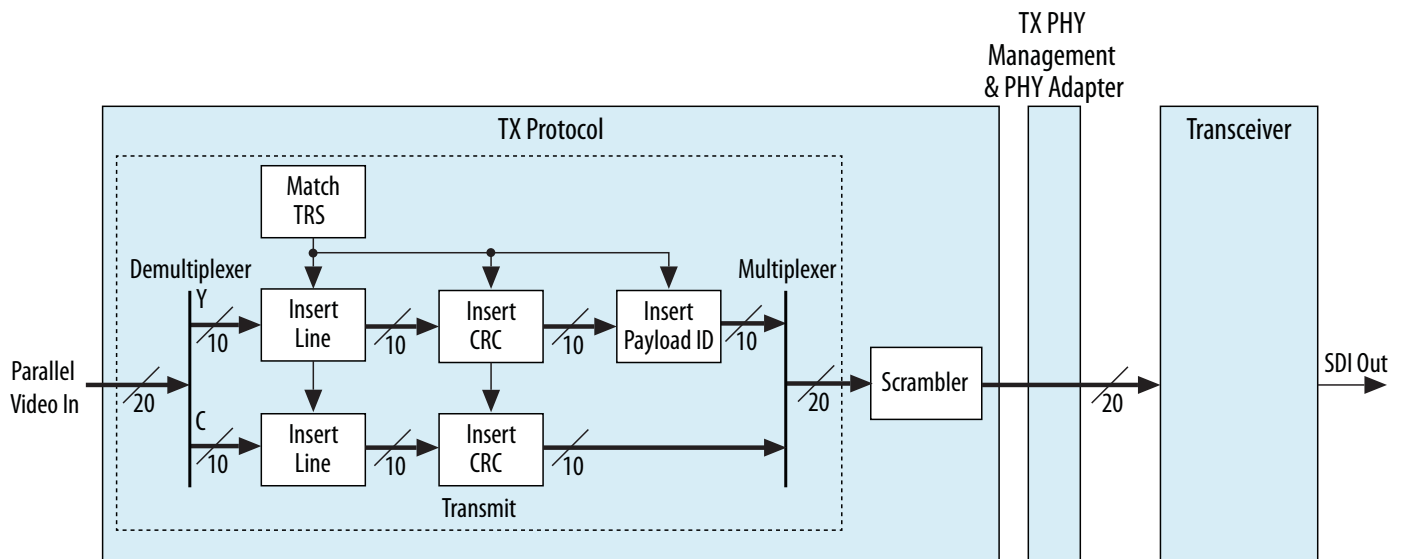


Figure 5-5: Dual Rate SDI Transmitter Data Path Block Diagram

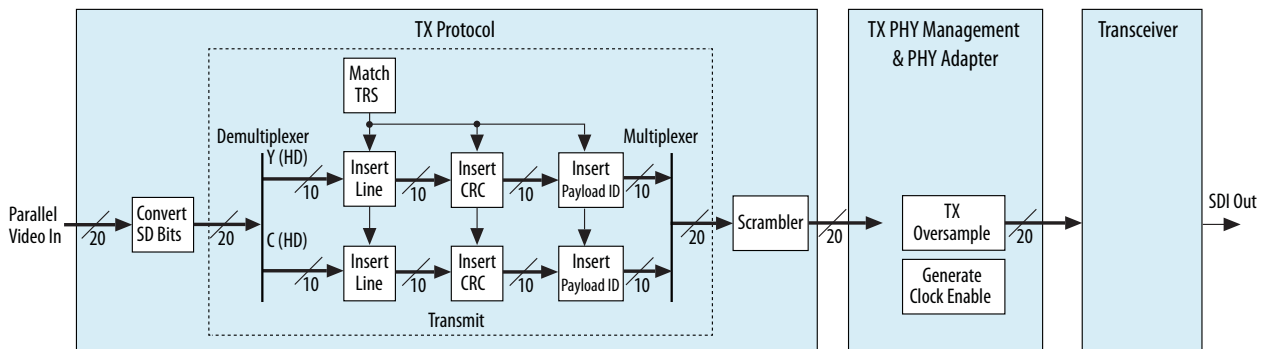


Figure 5-6: Dual Link HD-SDI Transmitter Data Path Block Diagram

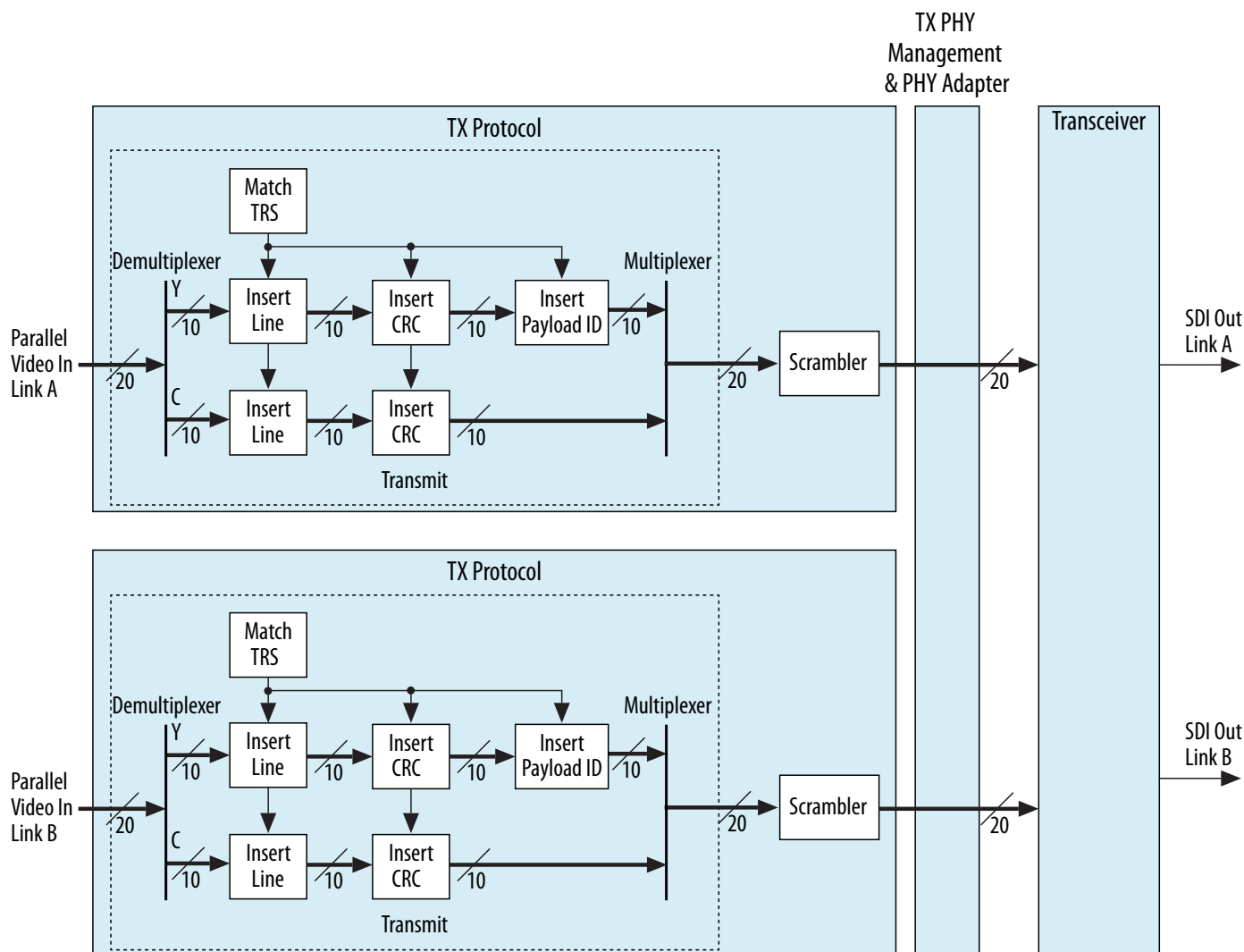


Figure 5-7: Triple Rate SDI Transmitter Data Path Block Diagram

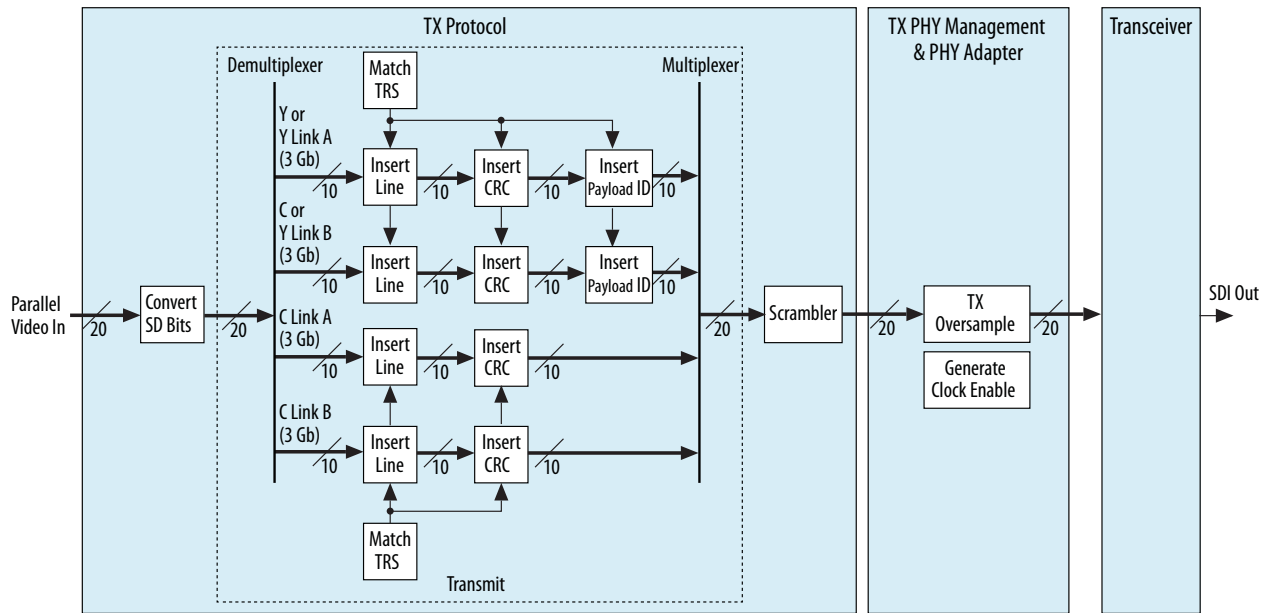
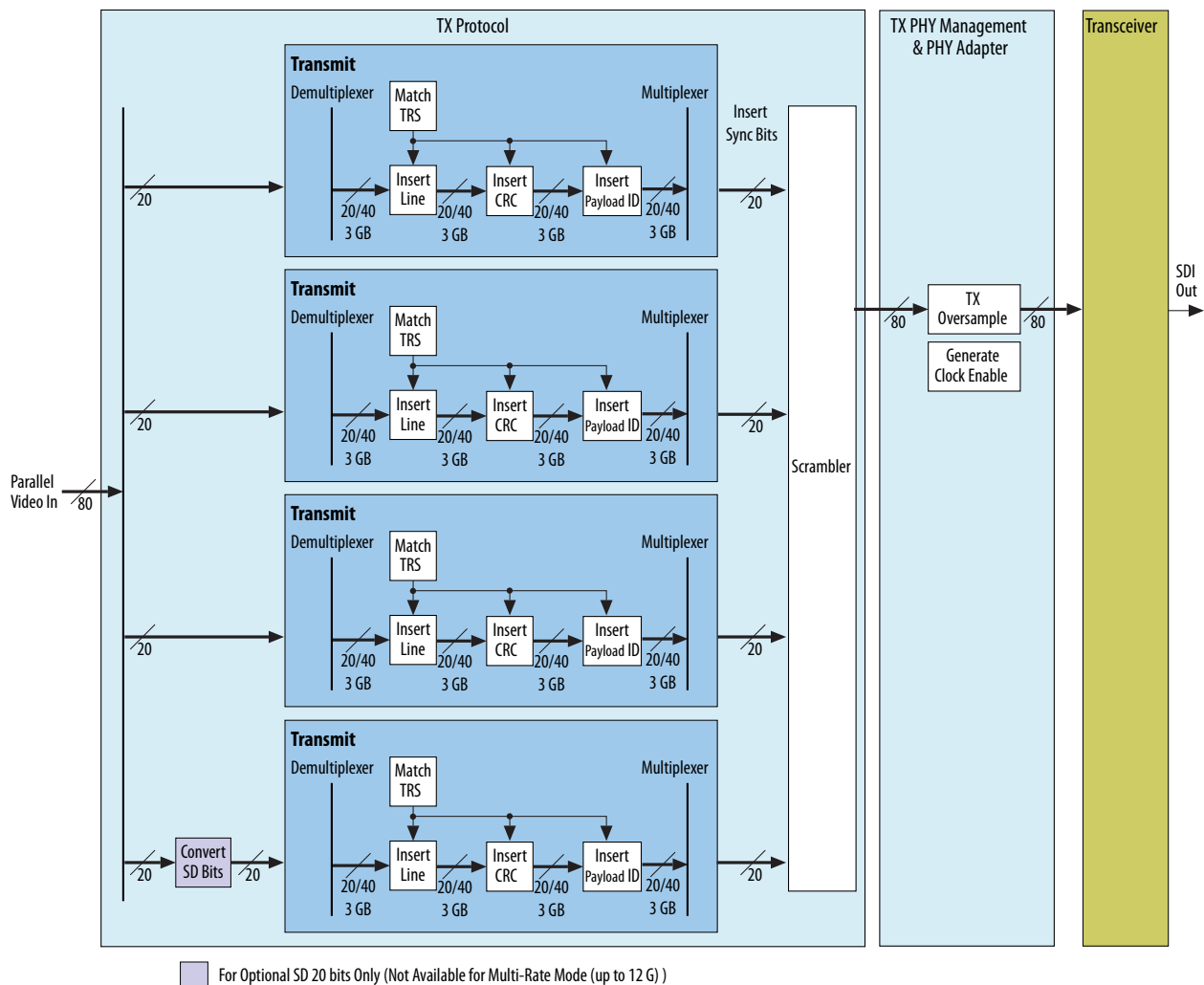


Figure 5-8: Multi Rate (up to 12G-SDI) Transmitter Data Path Block Diagram

Note: The transmit block shown in the diagram is the simplified version of the transmit block in the *Triple Rate SDI Transmitter Data Path Block Diagram*.



Related Information

[Submodules](#) on page 5-14

Receiver

The receiver performs the following functions:

- Video standard detection
- Video rate detection
- NRZI decoding and descrambling
- Word alignment
- Demultiplex data links

- Video timing flags extraction
- HD-SDI LN extraction
- HD-SDI CRC
- Payload ID extraction
- Synchronizing data streams
- Accessing transceiver
- Identifying and tracking of ancillary data
- Sync bit removal

The block diagrams below illustrate the SDI II IP core receiver (simplex) data path for each supported video standard.

Figure 5-9: SD-SDI Receiver Data Path Block Diagram

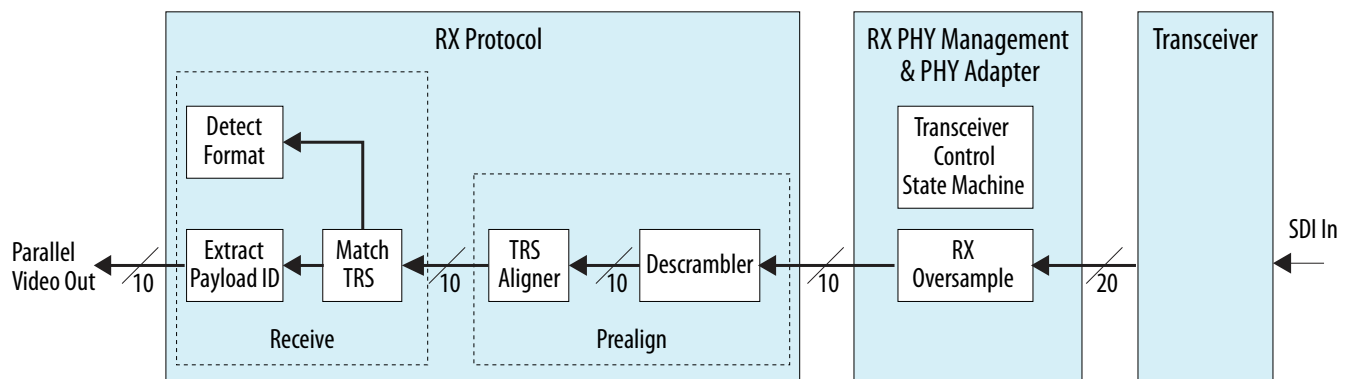


Figure 5-10: HD-SDI Receiver Data Path Block Diagram

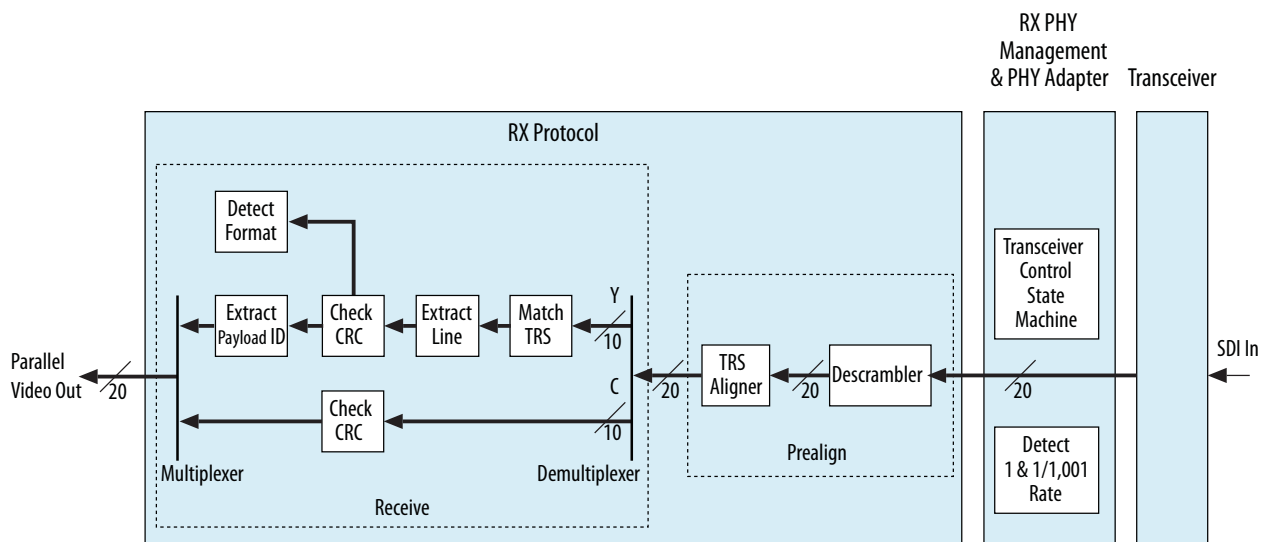


Figure 5-11: 3G-SDI Receiver Data Path Block Diagram

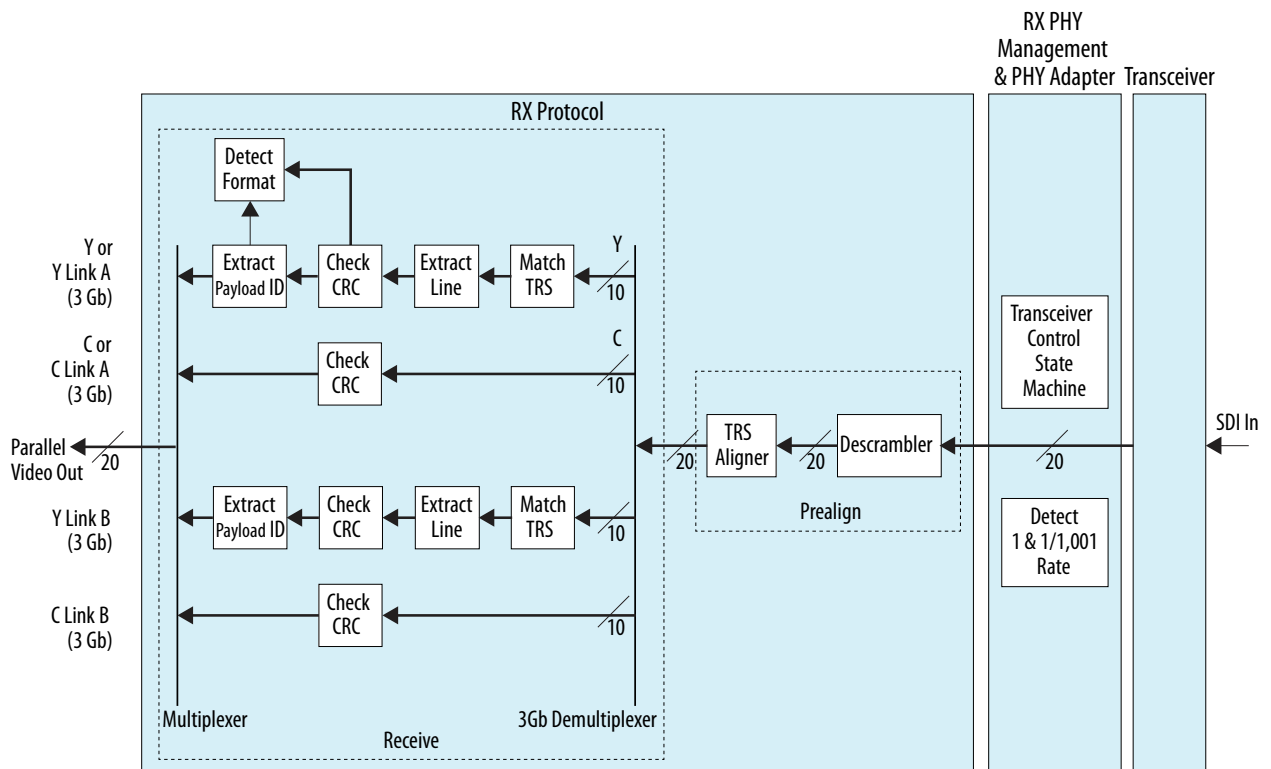


Figure 5-12: Dual Rate SDI Receiver Data Path Block Diagram

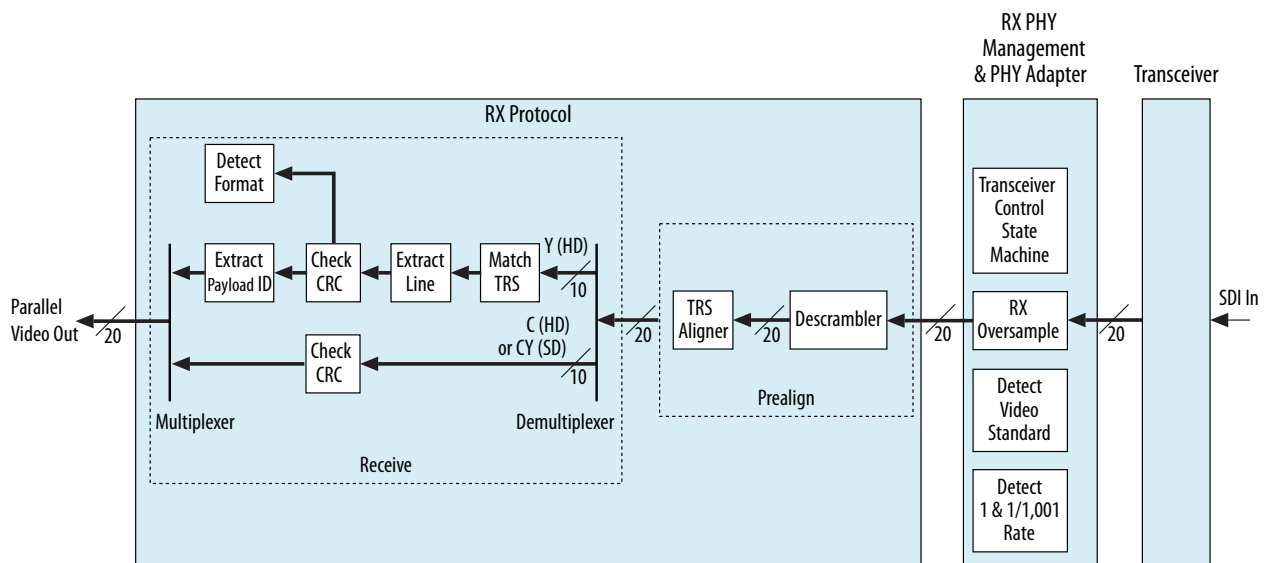


Figure 5-13: Dual Link HD-SDI Receiver Data Path Block Diagram

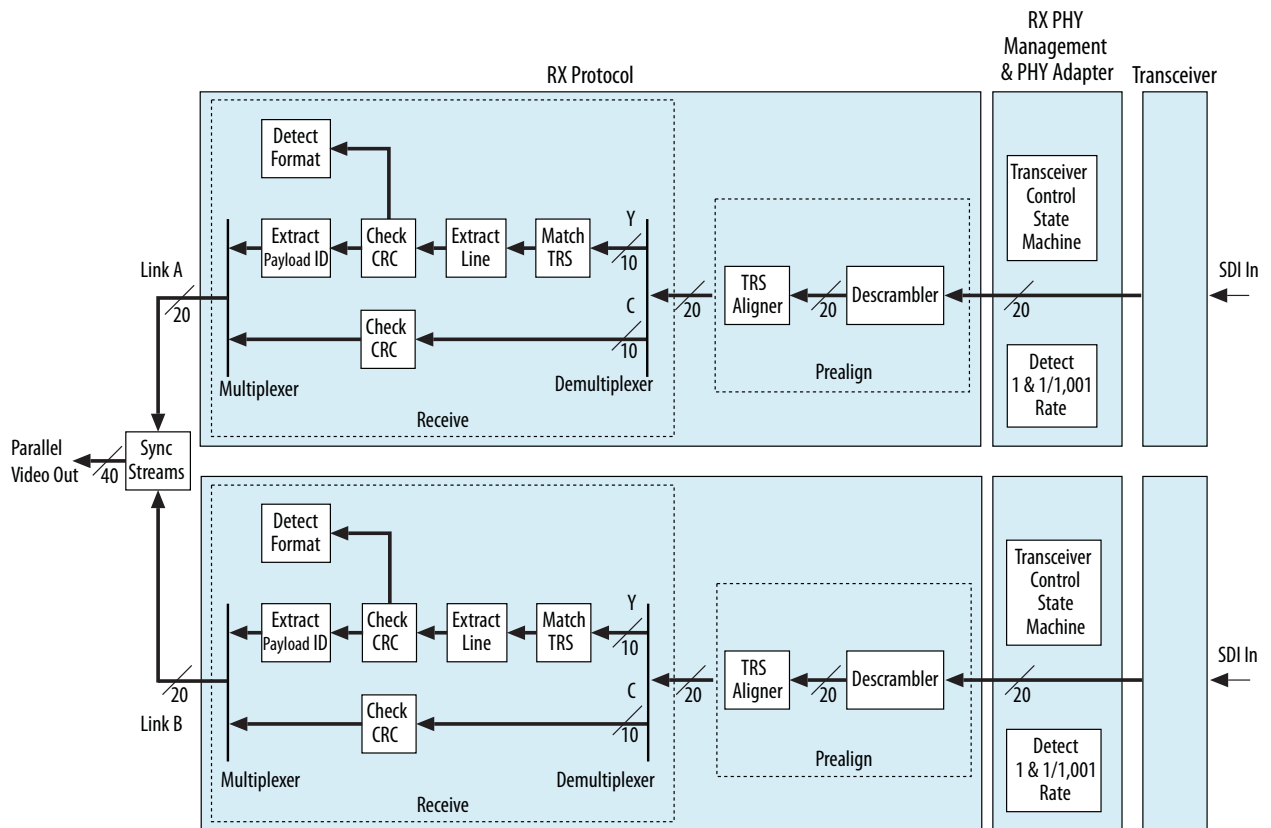


Figure 5-14: Triple Rate SDI Receiver Data Path Block Diagram

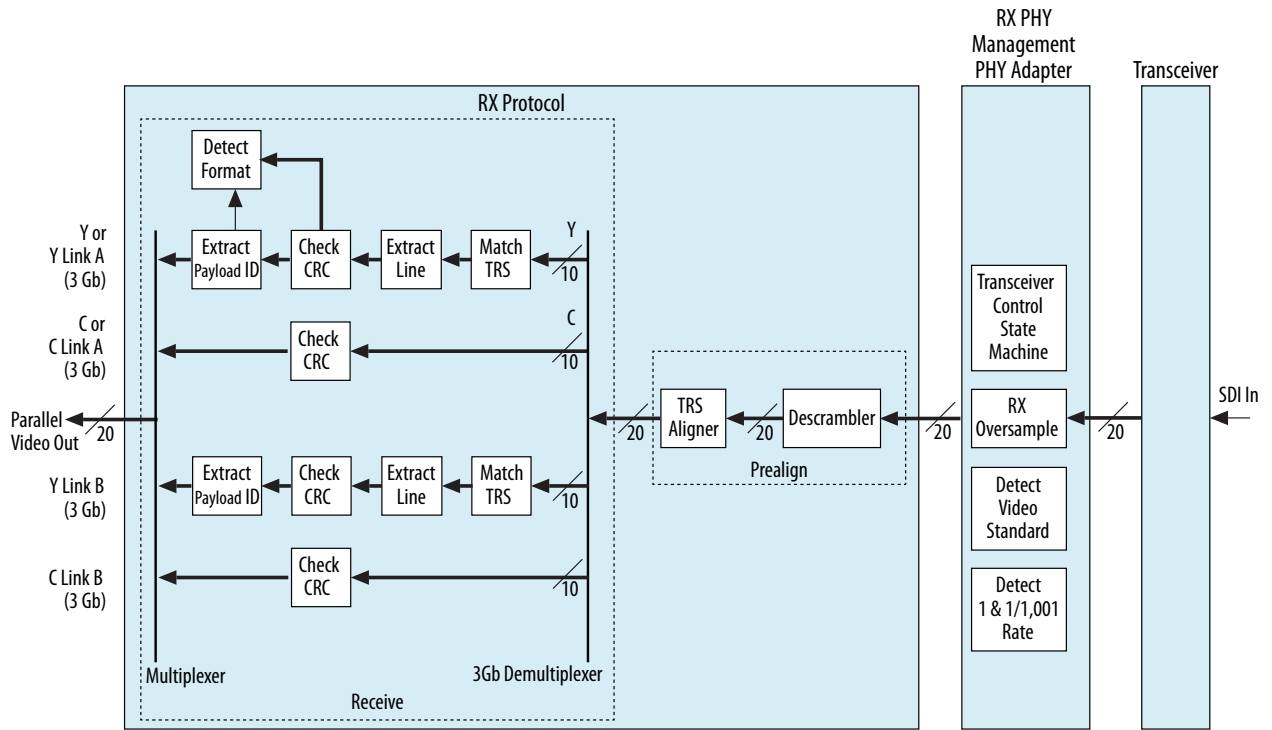
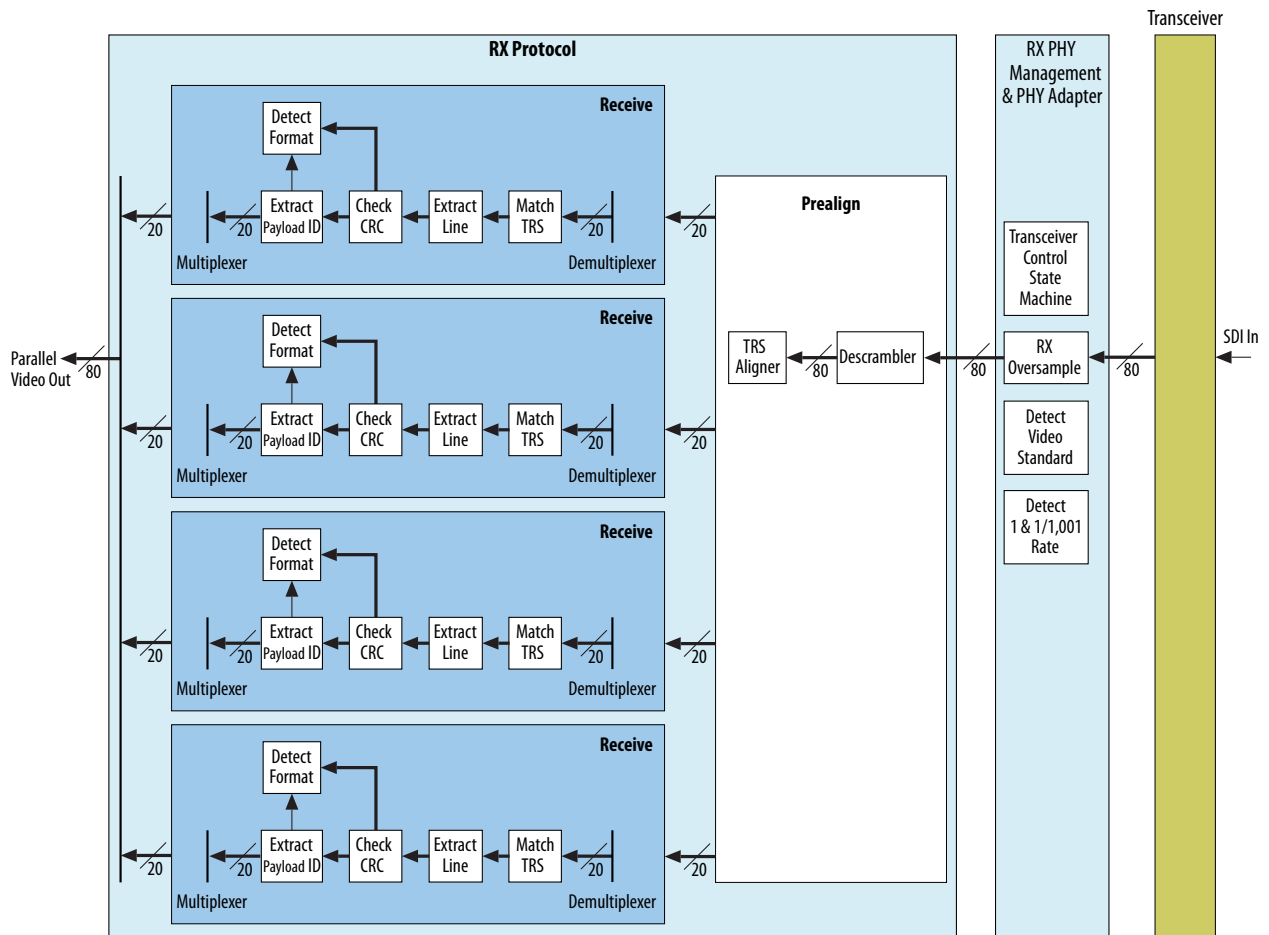


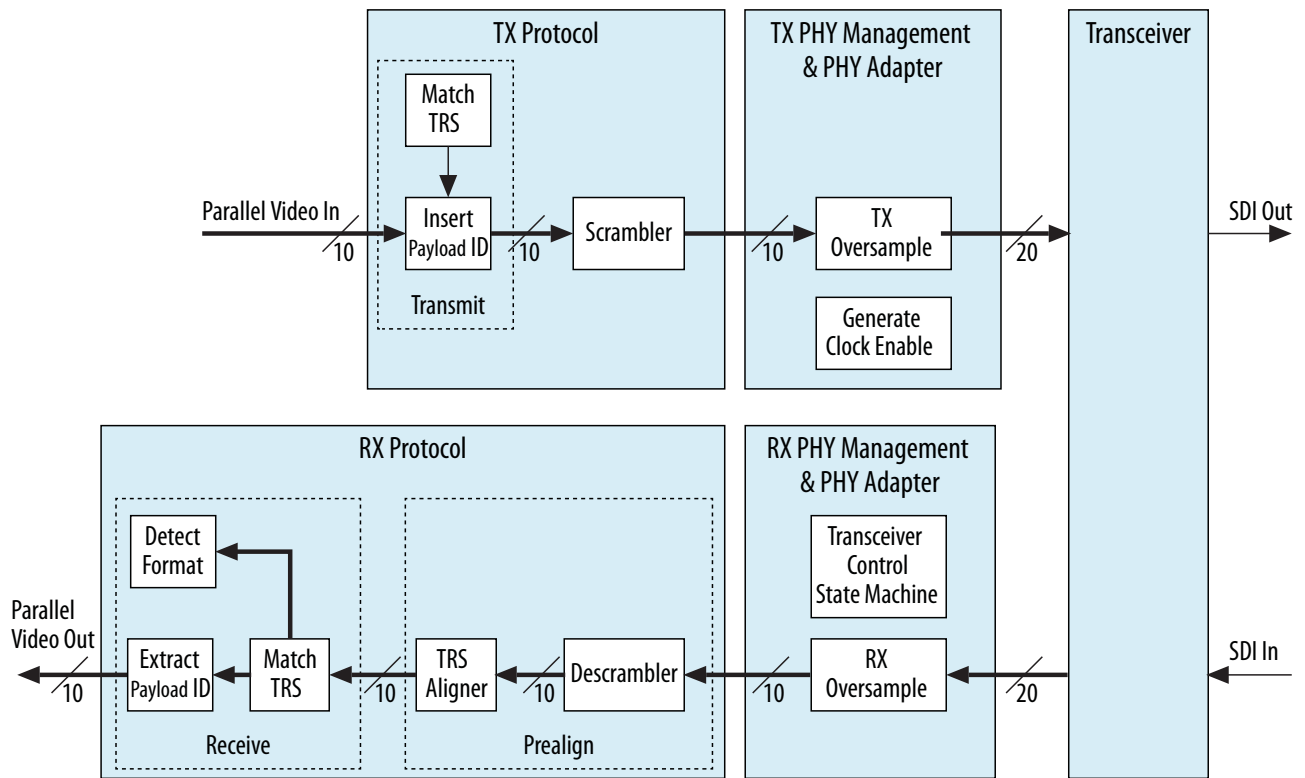
Figure 5-15: Multi Rate (up to 12G-SDI) Receiver Data Path Block Diagram

Note: The receive block shown in the diagram is the simplified version of the transmit block in the *Triple Rate SDI Receiver Data Path Block Diagram*.



For bidirectional or duplex mode, the protocol and PHY management & adapter blocks remain the same for each direction, except the hard transceiver, which is configured in duplex mode. The figure below illustrates the data path of a SD-SDI duplex mode.

Figure 5-16: SD-SDI Duplex Mode Block Diagram



Transceiver

The transceiver block consists of two components:

- PHY management and adapter
- hard transceiver

These two components handle the serial transport aspects of the SDI IP core.

Note: The transceiver block is only available for Arria V, Cyclone V, and Stratix V devices. For Arria 10 devices, you must generate the transceiver and the TX PLL separately.

The hard transceiver uses the Altera Native PHY IP Core for the following devices:

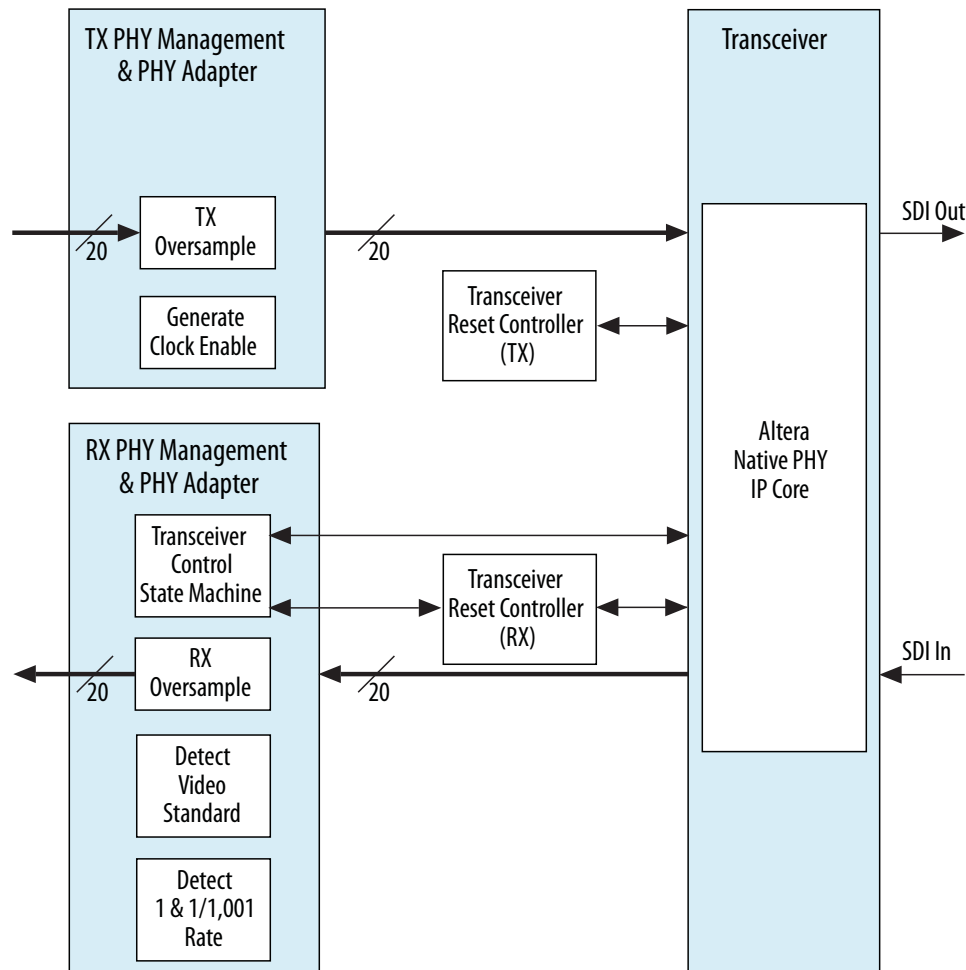
- Arria V GX (`altera_xcvr_native_av_hw.tcl`)
- Arria V GZ (`altera_xcvr_native_avgz_hw.tcl`)
- Stratix V (`altera_xcvr_native_sv_hw.tcl`)
- Cyclone V (`altera_xcvr_native_cv_hw.tcl`)

The SDI II IP core instantiates the PHY IP core using the Tcl file associated with each device.

The block diagram below illustrates the Native PHY IP core setup in the SDI II IP core (duplex) data path.

Figure 5-17: Altera Native PHY IP Core Setup in Duplex Mode

The Altera Native PHY IP Core does not include an embedded reset controller and an Avalon-MM interface. This PHY IP core exposes all signals directly as ports. To implement reset functionality for a new IP core, the transceiver reset controller is required to handle all the transceiver reset sequencing. The transceiver reset controller controls the embedded reset controller and also manages additional control options such as automatic or manual reset recovery mode.



Related Information

[Altera Transceiver PHY IP Core User Guide](#)

More information about the Altera Native PHY IP Core.

Submodules

You can reuse the submodules in the protocol and transceiver components across different video standards. The SDI II IP core consists of the following submodules:

- Insert Line
- Insert/Check CRC
- Insert Payload ID
- Match TRS
- Scrambler
- Tx Oversample
- Rx Oversample
- Detect Video Standard
- Detect 1 & 1/1.001 Rates
- Transceiver Controller
- Descrambler
- TRS Aligner
- 3Gb Demux
- Extract Line
- Extract Payload ID
- Detect Format
- Sync Streams
- Convert SD Bits
- Insert Sync Bit Values
- Remove Sync Bit Values

Insert Line

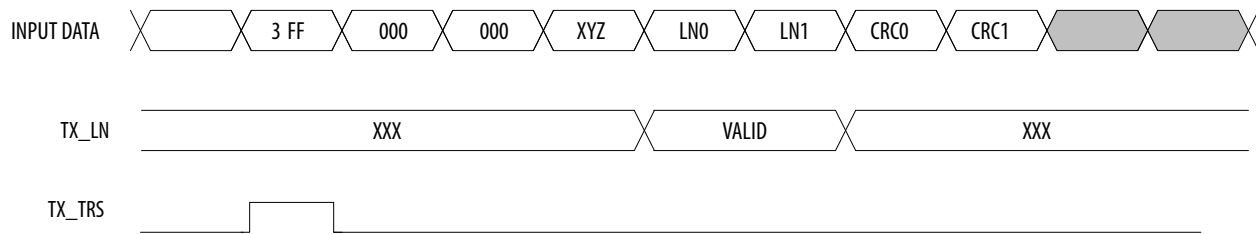
The insert line submodule provides HD-SDI and higher standards the option to include line numbers along with the video data.

This information is at the end of active video (EAV) extension words of the data stream, as defined in the SMPTE292M specification. The line number is 11 bits wide and spreads over two SDI words to use the SDI legal data space.

This submodule takes the 11-bit line number data value, correctly encodes them, and inserts them into the 10-bit stream. The line number value is user-defined. The top level port signal is `tx_ln[10:0]` and `tx_ln_b[10:0]` for link B in 3G-SDI (level B) and HD dual link modes. You also have the option to enable or disable this feature using the `tx_enable_ln` signal at the top level port. The SDI II IP core inserts the same line number value into both video channels. The Y and C channels require two of these submodules.

Figure 5-18: Line Number Insertion and Signal Requirements

This figure illustrates the line number insertion and signal requirements. For a correct line insertion, assert the `tx_trs` signal for the first word of both EAV and start of active video (SAV) TRS.



Insert/Check CRC

The HD-SDI can optionally include a line-based CRC code, which makes up two of the EAV extension words as defined in the SMPTE292M specification.

This submodule calculates the CRC based on the LFSR approach in the SMPTE specification. Note that you can configure this submodule to either insert or check the CRC.

For the transmitter, the core formats and inserts the CRC into two CRC EAV extension words—CRC0 and CRC1. For correct CRC generation and insertion, assert the `tx_trs` signal for the first word of both EAV and SAV TRS as shown in the [Insert Line](#) timing diagram. Perform CRC insertion only when the top level port, `tx_enable_crc`, is set to logic 1.

For the receiver, the core checks the CRC against the value of CRC0 and CRC1 that appear in the incoming stream. If there is a mismatch between the locally calculated value and the value in the stream, this submodule indicates an error.

Insert Payload ID

The SMPTE352M specification defines an ancillary packet type that provides specific information about the video payload carried by a digital interface. These payload ID packets carry information such as the interface type, sampling structure, component bit depth, and picture update rate.

Recent SMPTE interfaces such as dual link HD-SDI and 3G-SDI require the payload ID packets because it is very difficult to properly interpret the video data without the packet information from the payload ID packets.

The payload ID packet must be on specific video line locations at the beginning of the horizontal ancillary (HANC) space in one of these two conditions:

- Right after the EAV.
- Right after the CRC words that follow the EAV (for interfaces using CRC words).

Table 5-1: Payload ID Packet Location

Video Format	Field	Line Number
525i	1	13
	2	276
625i	1	9
	2	322
1080i	1	10
	2	572
525p	—	13
625p	—	9
720p	—	10
1080p	—	10

For dual link HD-SDI interface, the payload ID packets are placed only in the Y data stream of both links. This submodule in the transmitter data path modifies the Y data stream that passes through.

Note: This submodule introduces a latency of a few clock cycles. The C data stream is delayed by a few clock cycles to keep it synchronized with the Y data stream.

The following rules apply for inserting and overwriting payload ID packets:

- Rule 1: If there is no ancillary packet at the beginning of the HANC space on a line where the payload ID packet is supposed to occur, the submodule inserts the payload ID packet at the beginning of the HANC space.
- Rule 2: If there is an existing payload ID packet at the beginning of the HANC space on a line where the payload ID packet is supposed to occur, the submodule overwrites the packet with the new payload ID information if the `tx_vpid_overwrite` signal is high. If the `tx_vpid_overwrite` signal is low, the submodule will not overwrite.
- Rule 3: If there is a different type of ancillary packet(s) at the beginning of the HANC space on a line where the payload ID packet is supposed to occur, the submodule does not overwrite the existing ancillary packet(s). Instead, the submodule looks for empty space in the HANC space to insert the payload ID packet after the existing ancillary packet(s). If the submodule finds a payload ID packet later in the HANC space before finding an empty space, it overwrites the existing payload ID packet with the new data if the `tx_vpid_overwrite` signal is high. If the `tx_vpid_overwrite` signal is low, the submodule will not overwrite.

For correct payload ID insertion, assert the `tx_trs` signal for the first word of both EAV and SAV TRS as shown in the [Insert Line](#) timing diagram.

Match TRS

This submodule indicates that the current word is a particular TRS word in both the transmitter and receiver.

Scrambler

The SMPTE259M and SMPTE292M specifications define a common channel coding for both SD-SDI and HD-SDI. This channel coding consists of a scrambling function ($G_1(X) = X^9 + X^4 + 1$), followed by NRZI encoding ($G_2(X) = X + 1$).

The scrambling submodule implements the channel coding by iteratively applying the scrambling and NRZI encoding algorithm to each bit of the output data, processing the LSB first. The code handles all transmit data: SD (10 bits wide), HD/3G (20 bits wide), 6G (40 bits wide), and 12G (80 bits wide).

TX Sample

The TX sample submodule is a transmit oversampling block. It repeats each bit of the input word a given number of times and constructs the output words.

This submodule relies on the fact that the input data is only valid on $1/x$ of the clock cycles, where x is the oversampling factor. Both the input and output words are clocked from the same clock domain.

Table 5-2: Oversampling Requirement

The table below lists the number of times oversampling is required for the different video standards.

Real Video Rate vs. IP Mode	SD-SDI	HD-SDI	Dual Rate	Triple Rate	Multi Rate
SD-SDI	11	Not applicable	11	11	44
HD-SDI	Not applicable	—	2	2	8
3G-SDI	Not applicable	Not applicable	Not applicable	Not applicable	4
6G-SDI	Not applicable	Not applicable	Not applicable	Not applicable	2
12G-SDI	Not applicable	Not applicable	Not applicable	Not applicable	—

Clock Enable Generator

The clock enable generator is a simple logic that generates a clock enable signal.

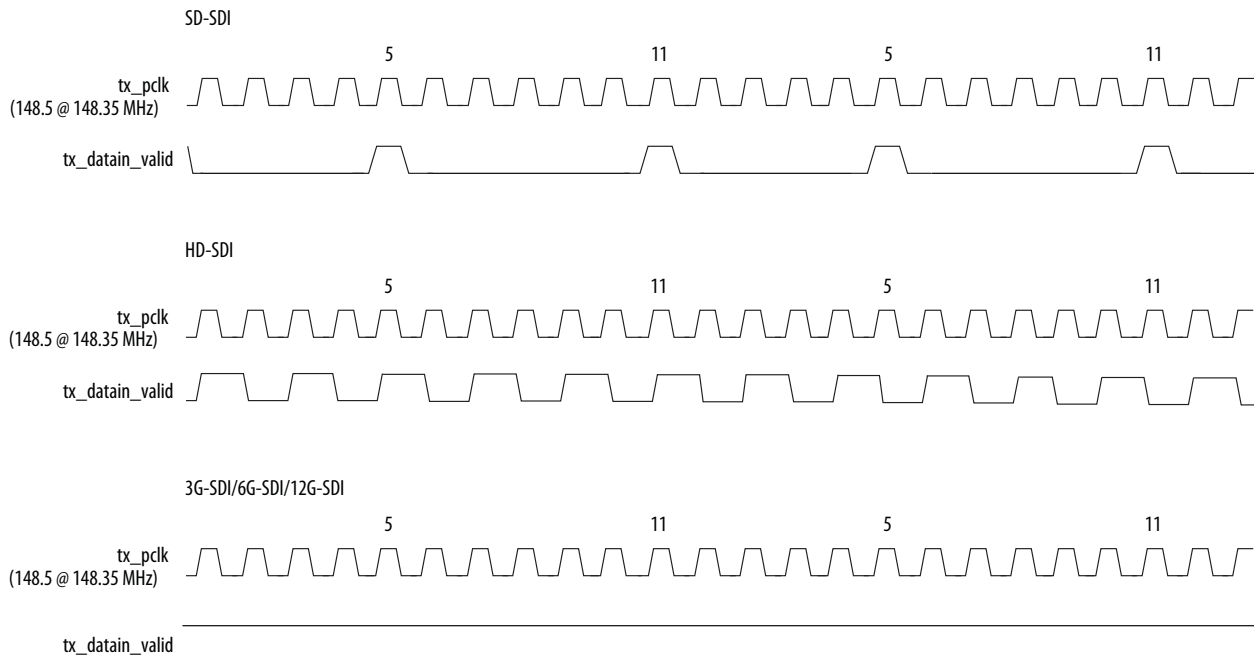
The clock enable signal serves as a data valid signal, `tx_datain_valid` for the incoming video data signal, `tx_datain`. The video data signal is based on the incoming video standard signal, `tx_std`. The transmit parallel clock, `tx_pclk`, can be a single frequency of either 148.5 MHz or 148.35 MHz.

The clock enable generator generates a clock signal in the following conditions:

- If the `tx_datain` signal is SD—generate a `tx_datain_valid` pulse every 5th and 11th clock cycle of the `tx_pclk` domain.
- If the `tx_datain` signal is HD—generate a `tx_datain_valid` pulse every other clock cycle of the `tx_pclk` domain.
- If the `tx_datain` signal is neither SD nor HD—the `tx_datain_valid` pulse remains high for 3G, 6G, or 12G.

Figure 5-19: Triple Rate Transmit Clocking Scheme

This figure illustrates the behavior of the `tx_datain_valid` pulse in each video standard.



RX Sample

This submodule extracts data from the oversampled incoming data stream. In oversampling schemes, each bit is repeated many times. For example, a stream of 0 1 0 1 may look like 000111000111 at the oversample clock or data rate.

Detect Video Standard

The detect video standard submodule performs coarse rate detection on the incoming video stream for dual-, triple-, or multi-rate SDI.

This scheme is required for the SDI II IP core to reprogram the transceivers to the correct settings for the video standard present at the input.

Related Information

[Transceiver Controller](#) on page 5-19

The transceiver controller controls the transceiver and performs dynamic reconfiguration (if necessary) to achieve the desired receiver functionality for the SDI.

Detect 1 and 1/1.001 Rates

This submodule indicates if the incoming video stream is running at PAL (1) or NTSC (1/1.001) rate. The output port signal, `rx_clkout_is_ntsc_paln` is set to 0 if the submodule detects the incoming stream as PAL (148.5 MHz or 74.25 MHz recovered clock) and set to 1 if the incoming stream is detected as NTSC (148.35 MHz or 74.175 MHz recovered clock).

For correct video rate detection, you must set the top level port signal, `rx_coreclk_is_ntsc_paln`, to the following bit:

- 0 if the `rx_coreclk` signal is 148.5 MHz or the `rx_coreclk_hd` signal is 74.25 MHz
- 1 if the `rx_coreclk` signal is 148.35 MHz or the `rx_coreclk_hd` signal is 74.175 MHz

Transceiver Controller

The transceiver controller controls the transceiver and performs dynamic reconfiguration (if necessary) to achieve the desired receiver functionality for the SDI.

When the interface receives SD-SDI, the receiver transceiver sets to lock-to-refclk (LTR) mode and when the interface receives HD-SDI or higher SDI data rate, the receiver transceiver sets to lock-to-data (LTD) mode.

In dual-rate, triple-rate, or multi-rate mode, the IP core first sets to the highest data-rate mode (transceiver running at 2.97 Gbps for dual/triple rate and 11.88 Gbps for multi rate) in LTR mode.

The detect video standard submodule starts running for a period of time. The output of this submodule determines if the transceiver requires dynamic reconfiguration to a new mode. The dual-rate and triple-rate modes use 11× oversampling to receive SD-SDI. This means that you require only two transceiver setups because the rates for 3G-SDI and 11× SD-SDI are the same. For multi-rate (up to 12G) modes, you require two more setups to accommodate 6G-SDI and 12G-SDI.

Related Information

[Detect Video Standard](#) on page 5-18

The detect video standard submodule performs coarse rate detection on the incoming video stream for dual-, triple-, or multi-rate SDI.

Descrambler

This submodule implements data descrambling as defined in the SMPTE259 and SMPTE292 specifications. This submodule is similar to the scrambler submodule, where it implements the reverse of the scrambling applied to the data. This submodule uses an LFSR and also implements NRZI.

TRS Aligner

The TRS aligner word aligns the descrambled receiver data until the bit order of the output data and the original video data are the same. The EAV and SAV sequences determine the correct word alignment.

Table 5-3: EAV and SAV Sequences

This table lists the sequence pattern for each video standard.	
Video Standard	EAV and SAV Sequences
SD-SDI	3FF 000 000
HD-SDI	3FF 3FF 000 000 000 000
3G-SDI Level A	3FF 3FF 000 000 000 000

Video Standard	EAV and SAV Sequences
3G-SDI Level B	3FF 3FF 3FF 3FF 000 000 000 000 000 000 000 000
6G-SDI with 4 Streams Interleaved	3FF 3FF 3FF 3FF 000 000 000 000 000 000 000 000
6G-SDI with 8 Streams Interleaved	3FF 3FF 3FF 3FF 3FF 3FF 3FF 3FF 000 000 000 000 000 000 000 000
12G-SDI with 8 Streams Interleaved	3FF 3FF 3FF 3FF 3FF 3FF 3FF 3FF 000 000 000 000 000 000 000 000
12G-SDI with 16 Streams Interleaved	3FF 3FF 3FF 3FF 3FF 3FF 3FF 3FF 3FF 3FF 3FF 3FF 3FF 3FF 3FF 3FF 000

The TRS aligner determines the correct word alignment for the data. The aligner looks for three consecutive TRSs with the same alignment and then stores that alignment. If the aligner subsequently detects two consecutive TRSs with a different alignment, then it stores this new alignment.

3Gb Demux

The 3Gb Demux submodule demultiplexes the Y link A, C link A, Y link B, and C link B from the received 20-bit data for further processing. This submodule is mainly for 3G-SDI (level B) operation and it is required in 3G-SDI and triple rate SDI modes.

Related Information

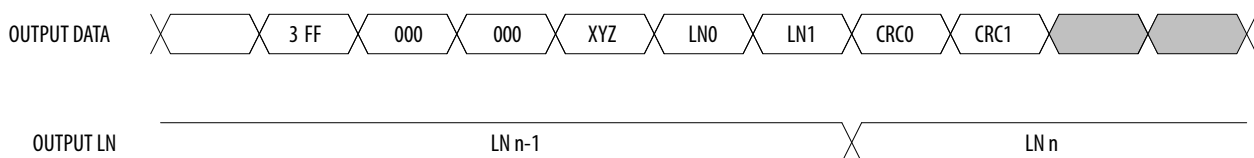
[3G-SDI \(Level B\) to HD-SDI Dual Link Conversion](#) on page 5-25

Extract Line

The HD-SDI and higher standards include the current video line number as part of the EAV extension words. The insert line submodule encodes the 11-bit line number in two of these extension words as defined in the SMPTE292M specification.

This submodule decodes the data words and registers them when the Match TRS submodule indicates that the current words are LN0 and LN1 extension words.

Figure 5-20: Line Number Extraction



Extract Payload ID

This submodule detects one 10-bit Y data stream from an interface and extracts the payload ID packet present in that data stream.

This submodule produces a valid signal, which indicates that a valid payload ID packet data is present on the submodule's payload output port. The submodule updates this payload each time it detects an error-free SMPTE352 packet. The submodule discards erroneous packets like checksum error and the payload port retains the information from the last good packet. The valid output signal goes high immediately upon receiving a good packet. If the submodule detects erroneous packets or the packets are no longer present, the valid output signal remains high for a number of frames or fields after the last good packet is received.

This submodule provides all four bytes of the payload ID data on its payload output port.

Detect Format

The detect format submodule monitors the line and frame timing of an incoming SDI stream. It generates various flags to indicate whether the receive stream is locked, and reports matching known video formats as `rx_format`.

A word counter monitors the EAV and SAV positions in the incoming video. The word counter increments on each valid word and stores the count value when an EAV or SAV is seen. If the count values are the same as a predefined value, the core determines the incoming video to be TRS locked. The predefined value is set to 6, therefore after six consecutive lines of the same EAV and SAV timing, the `rx_trs_locked` signal is active.

A line counter increments at the start of each video line. When the core finds the first active line of a field or frame, the line counter starts incrementing until the last active line of the same field or frame.

To determine the video format, a comparison logic compares the word and line count values in the video stream against the known values predefined for various video formats. The logic searches sequentially from one known value to another.

- If the logic finds a match, the core is determined to be frame locked and the `rx_frame_locked` signal is active. The core reports the matched known value as `rx_format`.
- If the logic does not find any match and the count is consistent over two video frames, the `rx_frame_locked` signal remains active but the `rx_format` stays asserted.

Sync Streams

This submodule is required in the HD-SDI dual link receiver as it synchronizes and deskews both data streams received by two separate transceivers of link A and link B. When the TRS word on both streams are aligned to each other, the core is considered locked and the `rx_dl_locked` signal asserts.

Convert SD Bits

This submodule is enabled when you set the **SD Interface Bit Width** parameter option to 20. This submodule converts the SD parallel data in 20 bits back to 10 bits as per the requirement for further processing.

This submodule contains a clock enable generator to generate two data valid pulses at every 11th clock cycle of the `tx_clk` domain. Each time the data valid signal is asserted, this block will alternately output the lower 10 bits and upper 10 bits of the SD 20-bit interface data to the downstream logic.

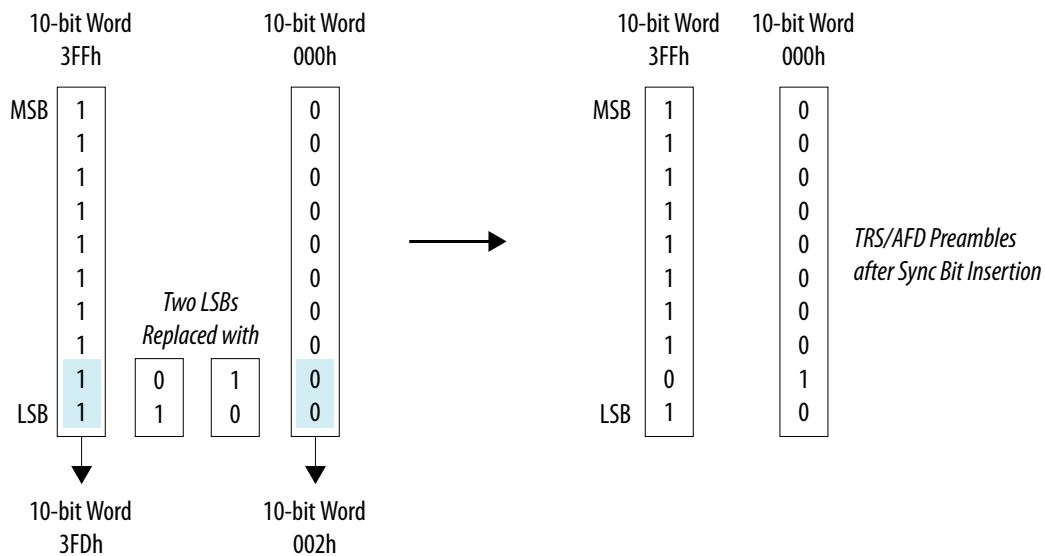
Insert Sync Bits

Inserting sync bits prevents long runs of 0s.

Repeating patterns of 3FFh or 000h for 6G-SDI and 12G-SDI video standards in the 10-bit parallel interface may result in a long run of zeroes feeding the scrambling polynomial. A long run of zeroes goes up to a length of 160 "1"s and 339 "0"s, which may cause the generation of the *pothole pathological* condition.

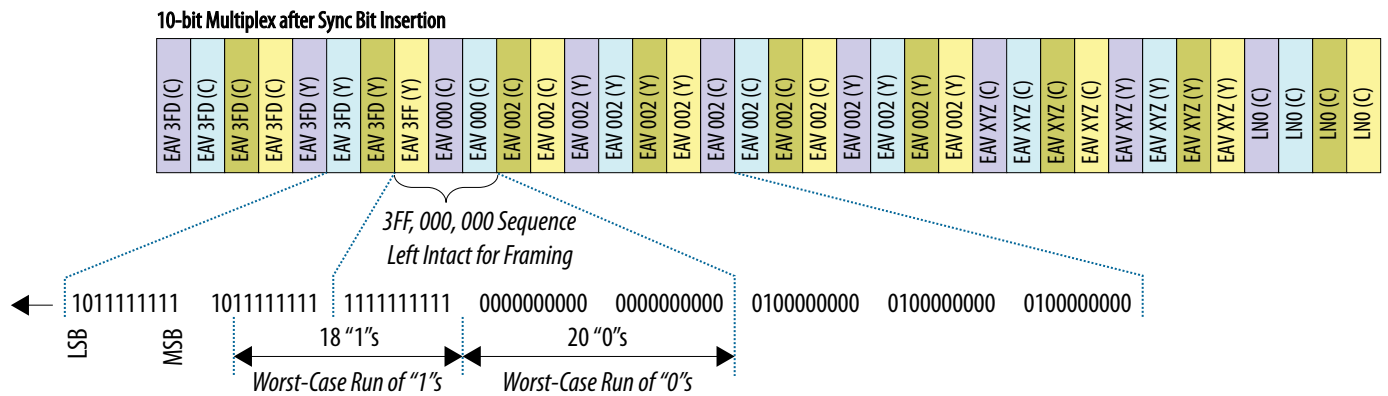
To prevent long runs, this feature modifies the 10-bit parallel interface data stream. It replaces the two LSBs of repeated 3FFh or 000 code words with sync-bit values of 10b for 000h words and 01b for 3FFh words.

Figure 5-21: Sync Bits



However, to ensure the words are synchronized and aligned in the receiver, this feature retains one complete sequence of preambles (3FFh 000h 000h) without modification.

Figure 5-22: Sync Bits Insertion Process



Remove Sync Bits

The sync bit inserted in 6G-SDI or 12G-SDI data from the source must be removed to allow other receiver submodules to function correctly.

This submodule detects the sync bit presented in the data stream and restores back the correct words, for example TRS words.

Optional Features

The SDI II IP core also provides some optional features.

HD-SDI Dual Link to 3G-SDI (Level B) Conversion

To interface between a HD-SDI dual link receiver and 3G-SDI single link transmitter equipment, perform a HD-SDI dual link to 3G-SDI (level B) conversion. Level B is defined as 2× SMPTE292 HD-SDI mapping, including SMPTE372 dual link mapping.

This conversion takes either two 1.485 Gbps dual link signals or two separate co-timed HD signals and combines them into a single 3G-SDI stream.

Figure 5-23: Example of HD-SDI Dual Link to 3G-SDI (Level B) Conversion

The figure shows the conversion of two HD-SDI data streams to 3G-SDI (level B) data streams.

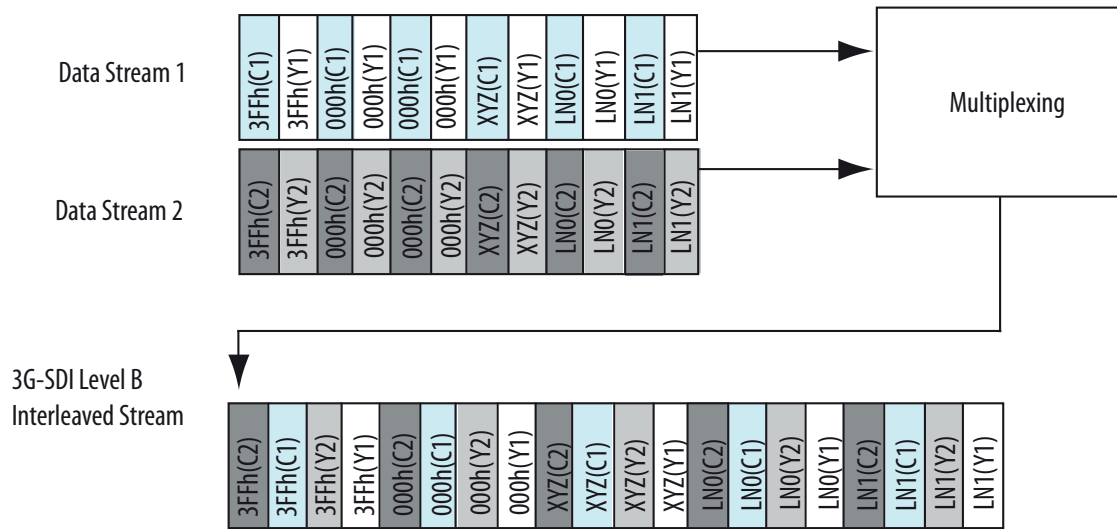
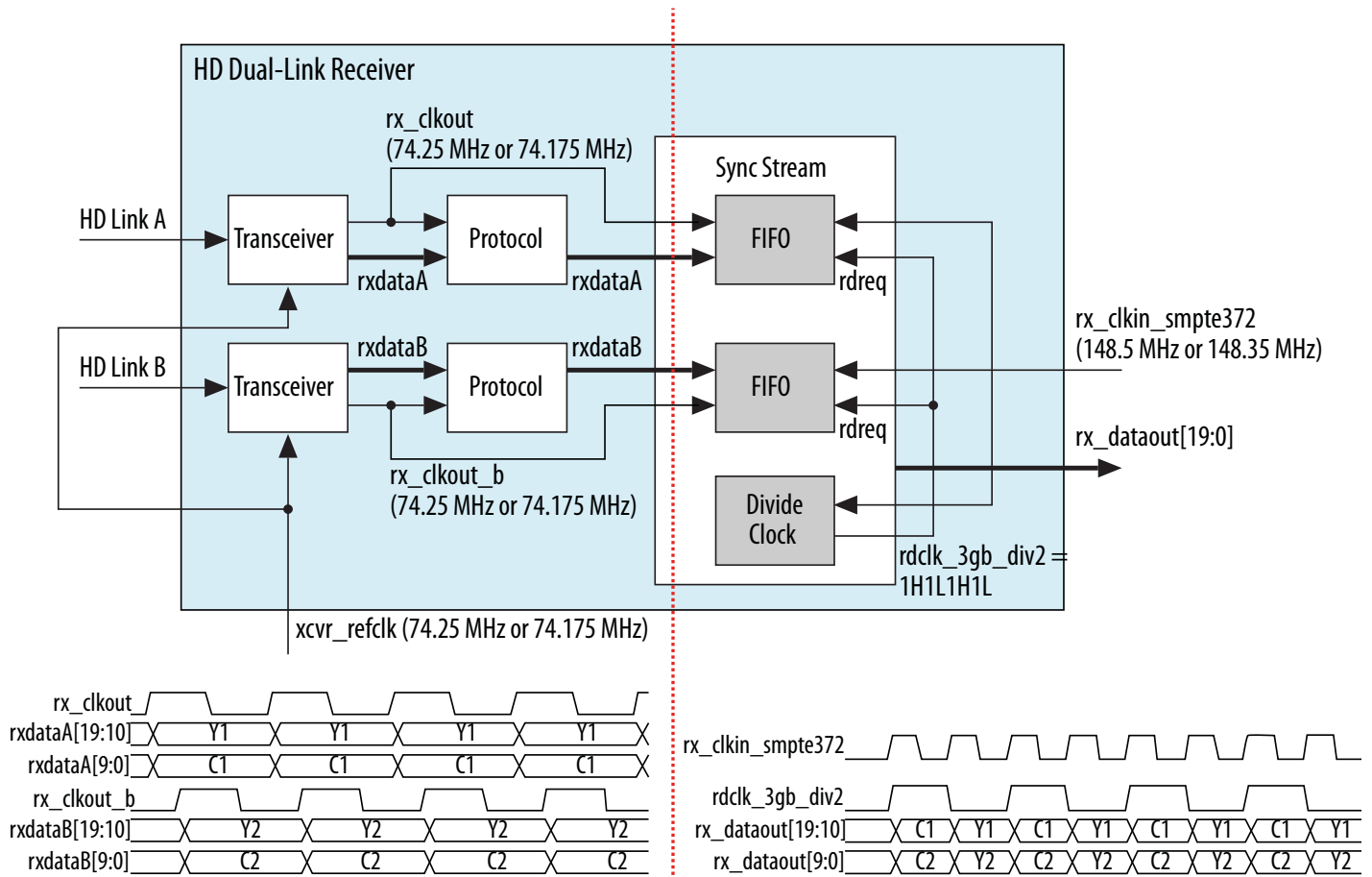


Figure 5-24: Implementation of HD-SDI Dual Link to 3G-SDI (Level B) Conversion

The figure shows a block diagram of HD-SDI dual link to 3G-SDI (level B) conversion.



3G-SDI (Level B) to HD-SDI Dual Link Conversion

To interface between 3-Gbps single link receiver and HD-SDI dual link transmitter equipment, perform a 3G-SDI (level B) to HD-SDI dual link conversion. This conversion takes a single 3G-SDI signal and separates the signal into two 1.485 Gbps signals, which can either be a dual link 1080p signal or two separate co-timed HD data streams.

Figure 5-25: Example of 3G-SDI (Level B) to HD-SDI Dual Link Conversion

The figure shows the conversion of 3G-SDI (level B) data to two HD-SDI data streams.

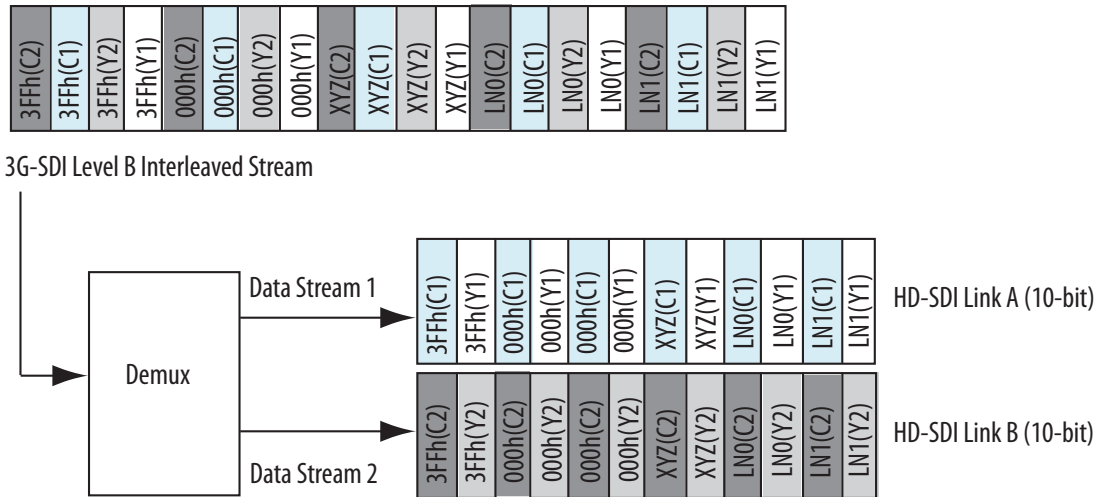
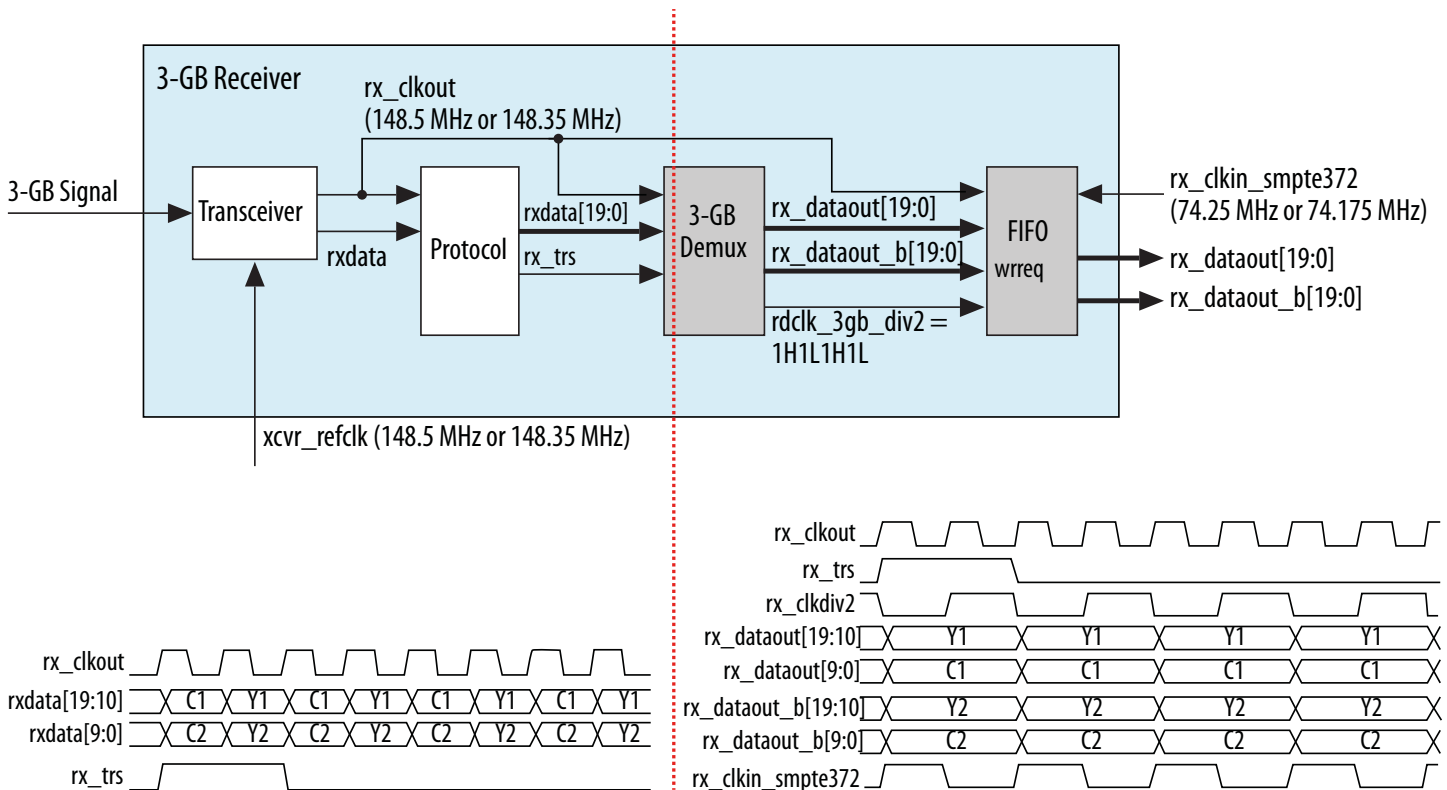


Figure 5-26: Implementation of 3G-SDI (Level B) to HD-SDI Dual Link Conversion

The figure shows a block diagram of 3G-SDI (level B) to HD-SDI dual link conversion.



SMPTE RP168 Switching Support

The SMPTE RP168 standard defines the requirements for synchronous switching between two video sources to take place with minimal interference to the receiver. The RP168 standard has restrictions for which lines the source switching can occur.

The SDI II IP core has flexibility and does not restrict you to switch at only a particular line defined in the RP168 standard. You can perform switching at any time between different video sources as long as the source has similar standard and format. After switching, all the status output signals, including the `rx_trs_locked`, `rx_frame_locked`, and `rx_align_locked` signals, remain unchanged. You should not see any interrupts at downstream.

SD 20-Bit Interface for Dual/Triple Rate

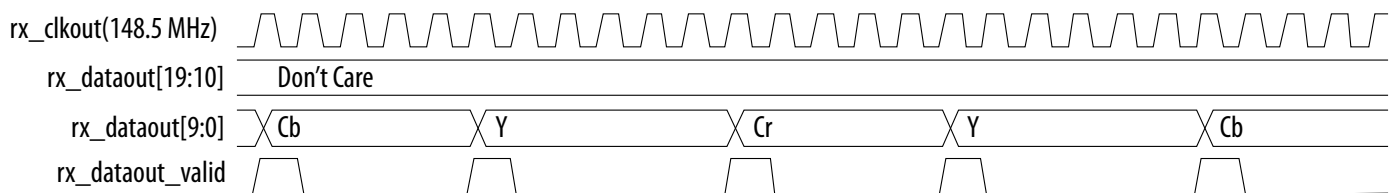
For a common SD interface, the serial data format is 10 bits wide, whereas for HD or 3G, the data format is 20 bits wide, divided into two parallel 10-bit datastreams (known as Y and C).

To make the interface bit width common for all standards in the dual-rate or triple-rate SDI mode:

- The receiver can extract the data and align them in 20-bit width
- The transmitter can accept SD data in 20-bit width and retransmit them successfully

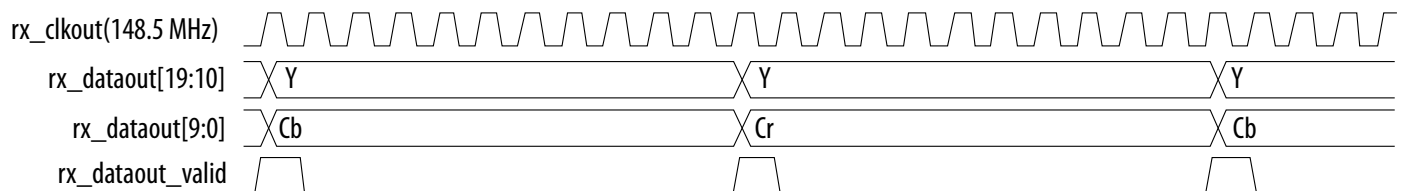
The timing diagrams below show a comparison of data arrangement between 10-bit and 20-bit interface.

Figure 5-27: SD 10-Bit Interface



- The upper 10 bits of `rx_dataout` are insignificant data.
- The lower 10 bits of `rx_dataout` are Luma (Y) and chroma (Cb, Cr) channels (interleaved).
- The 1H 4L 1H 5L cadence of `rx_dataout_valid` repeats indefinitely (ideal).

Figure 5-28: SD 20-Bit Interface



- The upper 10 bits of `rx_dataout` are Luma (Y) channel and the lower 10 bits are Chroma (Cb, Cr) channel.
- The 1H 10L cadence of `rx_dataout_valid` repeats indefinitely (ideal).

Dynamic TX Clock Switching for Arria V, Cyclone V, and Stratix V Devices

The dynamic TX clock switching feature allows you to dynamically switch between NTSC and PAL transceiver data rates for all video standards except SD-SDI.

The dynamic TX clock switching enables an SDI video equipment to operate on NTSC or PAL.

Note: This feature only supports Arria V, Cyclone V, and Stratix V devices.

You can choose to switch the TX clocks through one of these two methods:

- Instantiate an alternate TX PLL and supply two different clocks to the two PLLs. Switch between the primary PLL and the alternate PLL for transmission.
- Use the primary PLL with two reference input clocks. The PLL switches between these two clocks for transmission.

To implement this feature, you are required to provide two reference clocks (`xcvr_refclk` and `xcvr_refclk_alt`) to the SDI II IP core. The frequency of the reference clocks must be assigned to 148.5 MHz and 148.35 MHz in any assignment order.

The TX PLL select signal (`ch1_{tx/du}_tx_pll_sel`) is an input control signal that you provide to the core and the transceiver reconfiguration controller to select the desired clock input for the hard transceiver.

- Set `ch1_{tx/du}_tx_pll_sel` to 0 to select `xcvr_refclk`
- Set `ch1_{tx/du}_tx_pll_sel` to 1 to select `xcvr_refclk_alt`

To dynamically switch between the two reference clocks, you need to implement a simple handshaking mechanism. The handshake is initiated when the reconfiguration request signal (`ch1_{tx/du}_tx_start_reconfig`) is asserted high. This signal must remain asserted until the reconfiguration process completes. The reconfiguration process completes when the reconfiguration done signal (`ch1_{tx/du}_tx_reconfig_done`) is asserted high. The TX PLL select signal (`ch1_{tx/du}_tx_pll_sel`) needs to be stable throughout the reconfiguration process.

To complete the handshaking process, you must deassert the reconfiguration request signal (`ch1_{tx/du}_tx_start_reconfig`) upon assertion of the reconfiguration done signal (`ch1_{tx/du}_tx_reconfig_done`). The dynamic TX clock switching only takes effect after the `tx_rst` is asserted high and deasserted low accordingly.

Figure 5-29: Hardware Implementation of the Dynamic TX Clock Switching Feature

The figure shows the TX clock switching feature with two TX PLLs.

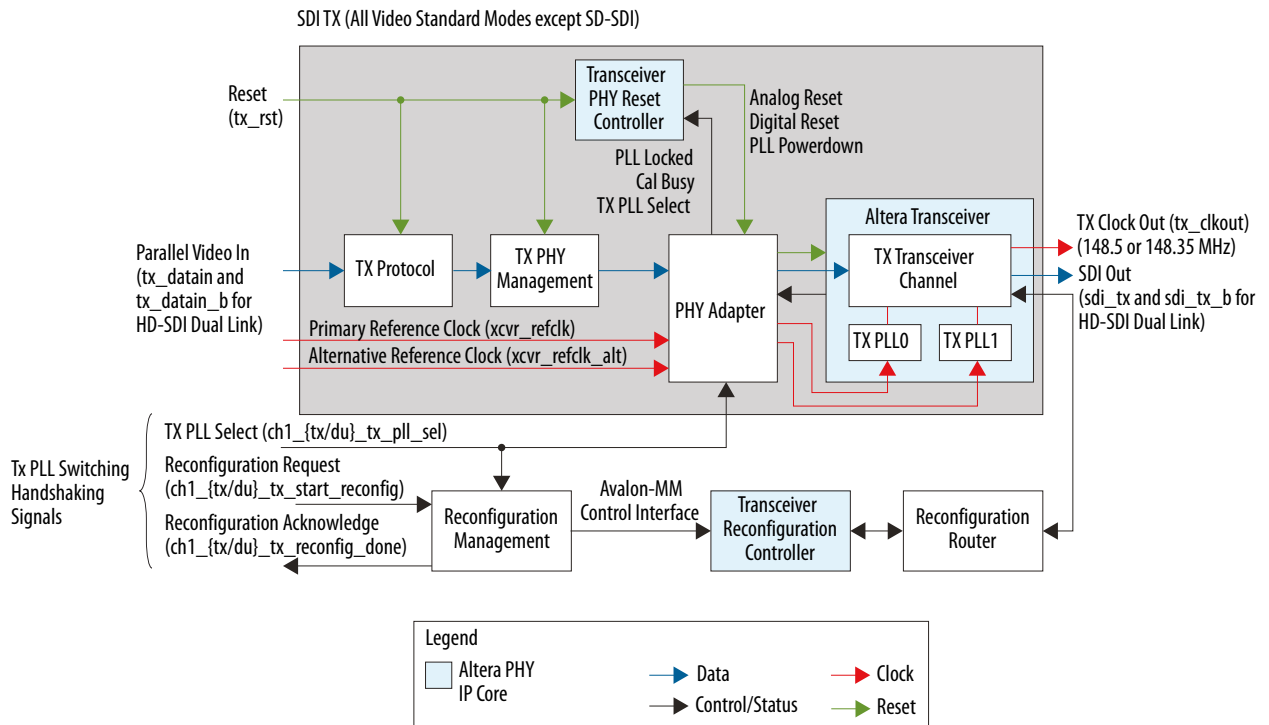
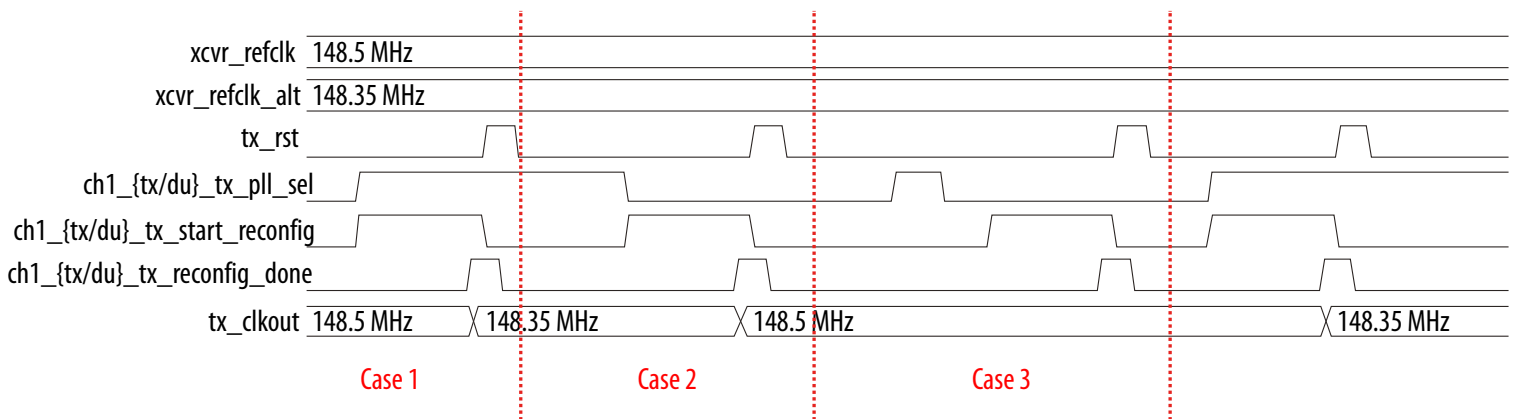


Figure 5-30: Dynamic TX Clock Switching Timing Diagram



The table below describes the behavior of the dynamic switching feature when you initiate a handshaking process (with reference to the timing diagram).

Table 5-4: Dynamic Switching Behavior During a Handshaking Process

Case	Description
1	The handshaking process attempts to switch to select <code>xcvr_refclk_alt</code> . <code>tx_clkout</code> successfully locks to <code>xcvr_refclk_alt</code> (148.35 MHz).
2	The handshaking process attempts to switch to select <code>xcvr_refclk</code> . <code>tx_clkout</code> successfully locks to <code>xcvr_refclk</code> (148.5 MHz).
3	The handshaking process attempts to switch to select <code>xcvr_refclk_alt</code> . The switching fails because <code>ch1_{tx/du}_tx_pll_sel</code> changes from 1 to 0 before the assertion of <code>ch1_{tx/du}_tx_start_reconfig</code> . Therefore, <code>tx_clkout</code> remains locked to <code>xcvr_refclk</code> (148.5MHz).



2016.12.20

UG-01125



Subscribe



Send Feedback

The following tables list the SDI II IP core signals by components.

- Protocol blocks—transmitter, receiver
- Transceiver blocks—PHY management, PHY adapter, hard transceiver

Note: These signals are applicable for Arria 10, Arria V, Cyclone V, and Stratix V devices unless specified otherwise.

SDI II Core Resets and Clocks

Table 6-1: Resets and Clock Signals

Signal	Width	Direction	Description
tx_rst	1	Input	<p>Reset signal for the transmitter. This signal is active high and level sensitive.</p> <p>This signal must be synchronous to tx_pclk clock domain (for Arria 10 devices) or tx_coreclk (for Arria V, Cyclone V, and Stratix V devices).</p>

© 2016 Intel Corporation. All rights reserved. Intel, the Intel logo, Altera, Arria, Cyclone, Enpirion, MAX, NIOS, Quartus and Stratix words and logos are trademarks of Intel Corporation in the US and/or other countries. Other marks and brands may be claimed as the property of others. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO
9001:2008
Registered

ALTERA
now part of Intel

Signal	Width	Direction	Description
pll_powerdown_in	1N	Input	<p>When asserted, this signal resets TX PLL. You must connect this signal to pll_powerdown_out.</p> <p>You can connect this signal from multiple SDI instances to pll_powerdown_out of one of the SDI instances to merge the PLL in these instances.</p> <p>For TX PLL merging, pll_powerdown_in and xcvr_refclk from multiple instances must share the same source.</p> <p>N = Number of PLLs in the core—1 (default) or 2 (when TX PLLs switching enabled)</p> <p>Note: Not applicable for these settings:</p> <ul style="list-style-type: none"> • In protocol only mode. • For Arria 10 devices. <p>If you enabled the Dynamic Tx clock switching parameter, your design requires XCVR_TX_PLL_RECONFIG_GROUP QSF assignment. Refer to the <i>Altera Transceiver PHY IP Core User Guide</i> for more information.</p>
pll_powerdown_out	1N	Output	<p>When asserted, this signal resets the selected TX PLL.</p> <p>N = Number of PLLs in the core—1 (default) or 2 (when TX PLLs switching enabled)</p> <p>Note: Not applicable for these settings:</p> <ul style="list-style-type: none"> • In protocol only mode. • For Arria 10 devices.
rx_rst	1	Input	<p>Reset signal for the receiver. This signal is active high and level sensitive. This reset signal must be synchronous to the rx_coreclk or rx_coreclk_hd clock domain.</p>
rx_rst_proto_in	1	Input	<p>Receiver protocol reset signal. This signal must be driven by the rx_rst_proto_out reset signal from the transceiver block.</p> <p>Note: Applicable for receiver protocol configuration only (Arria V, Cyclone V, and Stratix V devices).</p>
rx_rst_proto_in_b	1	Input	<p>Receiver protocol reset signal for link B. This signal must be driven by the rx_rst_proto_out_b reset signal from the transceiver block.</p> <p>Note: For HD-SDI dual link receiver protocol configuration only.</p>
rx_rst_proto_out	1	Output	<p>Reset the receiver protocol downstream logic. This generated signal is synchronous to rx_clkout clock domain and must be used to drive the rx_rst_proto_in signal of the receiver protocol block.</p>

Signal	Width	Direction	Description
rx_rst_proto_out_b	1	Output	Reset the receiver protocol downstream logic. Note: For HD-SDI dual link receiver transceiver configuration only.
trig_rst_ctrl	1	Output	Reset output signal to the transceiver reset controller to reset the transceiver. This signal is synchronous to the rx_coreclk or rx_coreclk_hd clock domain. Note: Applicable only for Arria 10 devices.
tx_pclk	1	Input	Transmitter core parallel clock signal. This clock signal must be driven by the by parallel output clock from TX transceiver. <ul style="list-style-type: none"> • SD-SDI = 148.5 MHz • HD-SDI = 74.25 MHz or 74.175 MHz, depending on video frame rate • 3G-SDI = 148.5 MHz or 148.35 MHz, depending on video frame rate • HD-SDI Dual Link = 74.25 MHz or 74.175 MHz, depending on video frame rate • Dual Rate = 148.5 MHz or 148.35 MHz, depending on video frame rate • Triple Rate = 148.5 MHz or 148.35 MHz, depending on video frame rate • Multi Rate (up to 12G-SDI) = 148.5 MHz or 148.35 MHz, depending on video frame rate
tx_coreclk	1	Input	148.5-MHz or 148.35-MHz transmitter core clock signal. This clock source must be always stable and can be shared with xcvr_refclk. Note: Not applicable for these settings: <ul style="list-style-type: none"> • In protocol only mode. • If the selected transceiver reference clock frequency is 74.25 MHz/74.175 MHz. • For Arria 10 devices.
tx_coreclk_hd	1	Input	74.25-MHz or 74.175-MHz transmitter core clock signal. This clock source must be always stable and can be shared with xcvr_refclk. Note: Applicable for HD-SDI and HD-SDI dual link modes only if the selected transceiver reference clock frequency is 74.25 MHz/74.175 MHz. Not applicable for Arria 10 devices.

Signal	Width	Direction	Description
rx_coreclk	1	Input	148.5-MHz or 148.35-MHz receiver core clock signal. This clock source must be always stable and can be shared with <code>xcvr_refclk</code> in Arria V, Cyclone V, and Stratix V devices. Note: Not applicable if the selected transceiver reference clock frequency is 74.25 MHz/74.175 MHz.
rx_coreclk_hd	1	Input	74.25-MHz or 74.175-MHz receiver core clock signal. This clock source must be always stable and can be shared with <code>xcvr_refclk</code> . Note: Applicable for HD-SDI and HD-SDI dual link modes only if the selected transceiver reference clock frequency is 74.25 MHz/74.175 MHz. Not applicable for Arria 10 devices.
rx_clkkin	1	Input	Receiver protocol clock input. This signal must be driven by the <code>rx_clkkout</code> clock signal from the transceiver block. <ul style="list-style-type: none"> SD-SDI = 148.5 MHz HD-SDI = 74.25 MHz or 74.175 MHz, depending on video frame rate 3G-SDI = 148.5 MHz or 148.35 MHz, depending on video frame rate Note: For receiver protocol configuration only. Not applicable for Arria 10 devices.
rx_clkkin_b	1	Input	Receiver protocol clock input for link B. This signal must be driven by the <code>rx_clkkout_b</code> clock signal from the transceiver block (74.25 MHz or 74.125 MHz, depending on video frame rate). Note: For HD-SDI dual link receiver protocol configuration only. Not applicable for Arria 10 devices.
rx_clkkin_smpte372	1	Input	Clock input for HD-SDI dual link to 3G-SDI (level B) and 3G-SDI (level B) to HD-SDI dual link operations. <ul style="list-style-type: none"> HD-SDI dual link to 3G-SDI (level B) = 148.5 MHz or 148.35 MHz 3G-SDI (level B) to HD-SDI dual link = 74.25 MHz or 74.175 MHz

Signal	Width	Direction	Description
<code>xcvr_rxclk</code>	1	Input	<p>Receiver parallel clock input. Driven by <code>rx_pma_div_clkout</code> (for multi-rate modes) or <code>rx_clkout</code> (for other modes) from the transceiver.</p> <ul style="list-style-type: none"> SD-SDI = 148.5 MHz HD-SDI = 74.25 MHz or 74.175 MHz, depending on video frame rate 3G-SDI = 148.5 MHz or 148.35 MHz, depending on video frame rate 6G-SDI = 148.5 MHz or 148.35 MHz, depending on video frame rate 12G-SDI = 148.5 MHz or 148.35 MHz, depending on video frame rate <p>Note: Applicable only for Arria 10 devices.</p>
<code>xcvr_refclk</code>	1	Input	<p>Reference clock signal for the transceiver. The clock source must be stable.</p> <ul style="list-style-type: none"> SD-SDI = 148.5 MHz HD-SDI = 74.25 MHz, 74.175 MHz, 148.5 MHz, or 148.35 MHz 3G-SDI = 148.5 MHz or 148.35 MHz HD-SDI Dual Link: 74.25 MHz, 74.175 MHz, 148.5 MHz, or 148.35 MHz Dual Rate: 148.5 MHz or 148.35 MHz Triple Rate: 148.5 MHz or 148.35 MHz <p>Note: Not applicable for Arria 10 devices.</p>
<code>xcvr_refclk_alt</code>	1	Input	<p>Alternative clock input for the hard transceiver. The frequency of this signal must be the alternate frequency value of the <code>xcvr_refclk</code> signal.</p> <ul style="list-style-type: none"> HD-SDI = 74.25 MHz, 74.175 MHz, 148.5 MHz, or 148.35 MHz 3G-SDI = 148.5 MHz or 148.35 MHz HD-SDI Dual Link: 74.25 MHz, 74.175 MHz, 148.5 MHz, or 148.35 MHz Dual Rate: 148.5 MHz or 148.35 MHz Triple Rate: 148.5 MHz or 148.35 MHz <p>Note: Applicable only when you turn on the Tx PLL Dynamic Switching option. Not applicable for Arria 10 devices.</p>
<code>tx_clkout</code>	1	Output	<p>TX transceiver parallel output clock.</p> <p>This frequency for this clock should be the same as the user-provided <code>xcvr_refclk</code>.</p> <p>Note: Not applicable for Arria 10 devices.</p>

Signal	Width	Direction	Description
rx_clkout	1	Output	RX transceiver parallel output clock. <ul style="list-style-type: none"> SD-SDI = 148.5 MHz HD-SDI = 74.25 MHz or 74.175 MHz, depending on video frame rate 3G-SDI = 148.5 MHz or 148.35 MHz, depending on video frame rate Note: Not applicable for Arria 10 devices.
rx_clkout_b	1	Output	RX transceiver parallel output clock for link B. The output clock frequency must be 74.25 or 74.175 MHz, depending on video frame rate. Note: For HD-SDI dual link only.

Figure 6-1: TX Clocking Diagram for Arria 10 Devices

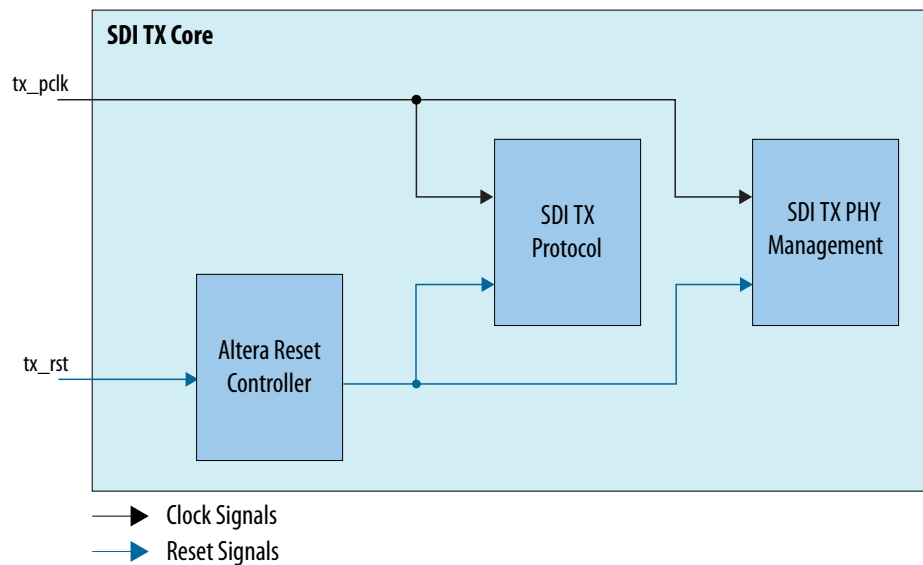
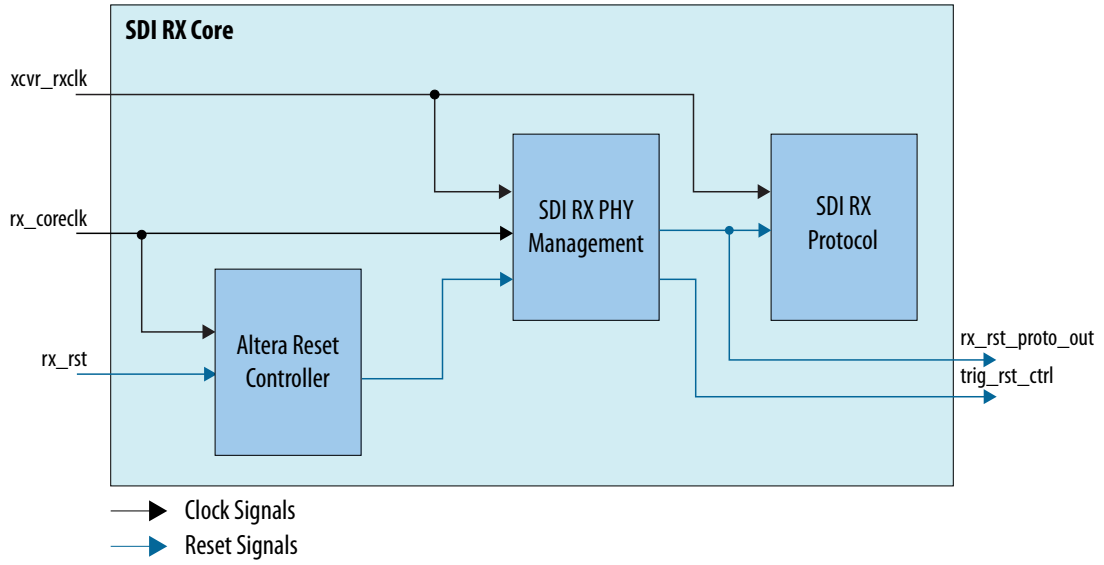


Figure 6-2: RX Clocking Diagram for Arria 10 Devices



Note: For a more comprehensive TX and RX Arria 10 clocking diagrams with transceivers, refer to the *SDI II IP Core Design Example User Guide*.

Figure 6-3: TX Clocking Diagram for Arria V, Cyclone V, and Stratix V Devices

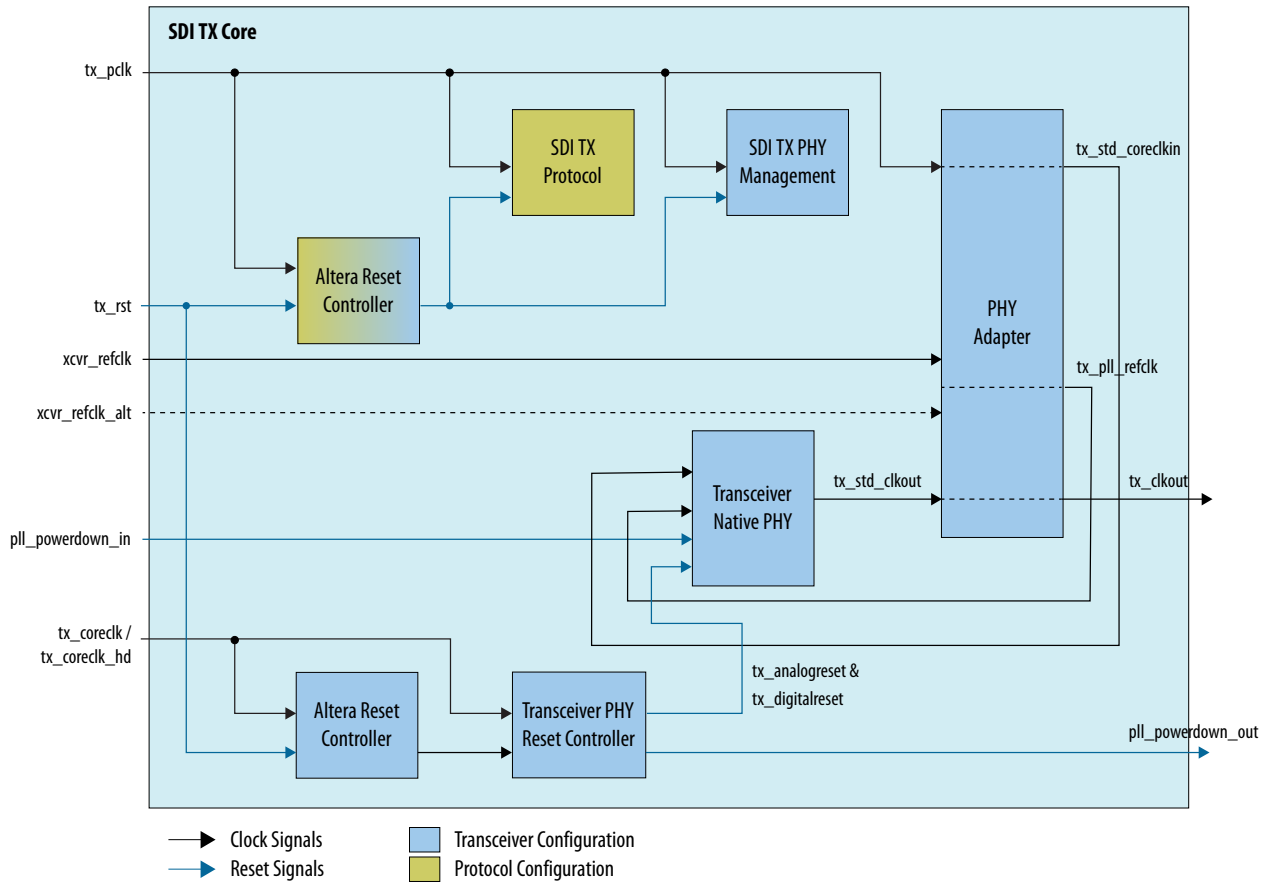
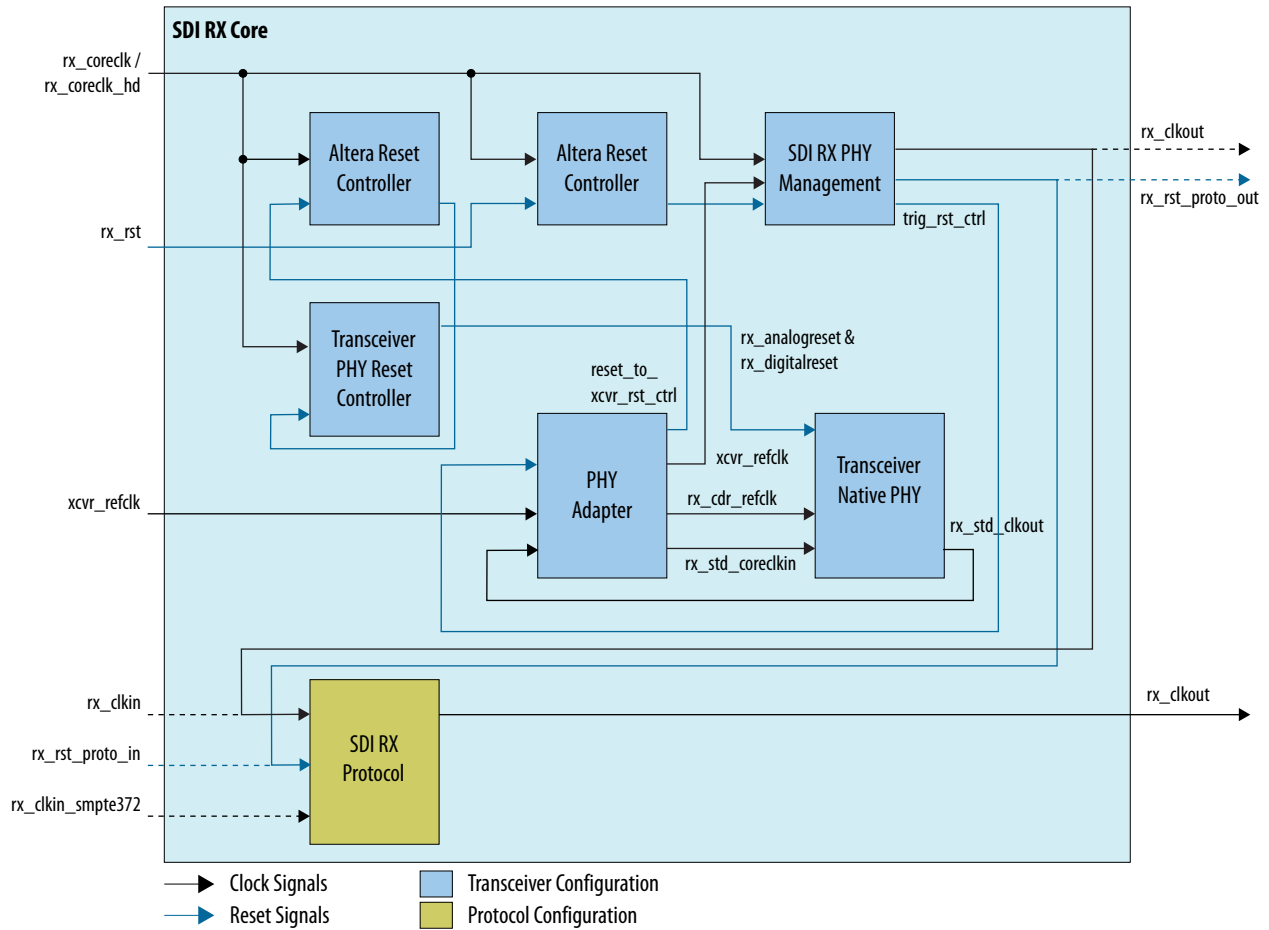


Figure 6-4: RX Clocking Diagram for Arria V, Cyclone V, and Stratix V Devices



Note: For the Arria V, Cyclone V, and Stratix V devices, the source or destination for dual link signals with "_b" suffix are the same as the original signals. For example, the destination for rx_clk_in_b is the same as rx_clk_in, which is directed to the SDI RX protocol block. Similarly, the source of the output signal rx_rst_proto_out_b is the same as rx_rst_proto_out, which comes from the SDI RX PHY management block.

Transmitter Protocol Signals

The transmitter protocol signals are synchronous to tx_pclk.

Table 6-2: Transmitter Protocol Signals

Note: S = Indicates the number of 20-bit interfaces; 4 for multi-rate (up to 12G) mode and 1 for other modes.

Signal	Width	Direction	Description
tx_enable_crc	1	Input	Enables CRC insertion for all modes except SD-SDI. Note: Not applicable for transceiver only configurations.
tx_enable_ln	1	Input	Enables LN insertion for all modes except SD-SDI. Note: Not applicable for transceiver only configurations.
tx_std	3	Input	Transmitter video standard. <ul style="list-style-type: none"> SD-SDI = 000 HD-SDI = 001 3G-SDI Level A = 011 3G-SDI Level B = 010 6G-SDI 4 Streams Interleaved = 101 6G-SDI 8 Streams Interleaved = 100 12G-SDI 8 Streams Interleaved = 111 12G-SDI 16 Streams Interleaved = 110 Note: Applicable for 3G-SDI, and dual-rate, triple-rate, and multi-rate modes.
tx_datain	20S	Input	User-supplied transmitter parallel data. <ul style="list-style-type: none"> SD-SDI = bits 19:10 unused; bits 9:0 C, Y multiplex HD-SDI = bits 19:10 Y; bits 9:0 C HD-SDI dual link = bits 19:10 Y link A, bits 9:0 C link A 3G-SDI Level A = bits 19:10 Y; bits 9:0 C 3G-SDI Level B = bits 19:10 C, Y multiplex (link A); bits 9:0 C, Y multiplex (link B) 6G-SDI: bits 79:40 unused; bits 39:30 data stream 1; bits 29:20 data stream 2; bit 19:10 data stream 3; bits 9:0 data stream 4. 12G-SDI: bits 79:70 data stream 1; bits 69:60 data stream 2; bit 59:50 data stream 3; bits 49:40 data stream 4; bits 39:30 data stream 5; bits 29:20 stream 6; bits 19:10 stream 7; bits 9:0 data stream 8 Refer to Image Mapping on page 6-13 for more information about the 6G-SDI and 12G-SDI image mapping. For transceiver only configurations, the transmitter does not scramble these data before sending to the hard transceiver.
tx_datain_b	20	Input	User-supplied transmitter parallel data for link B. HD-SDI dual link = bits 19:10 Y link B, bits 9:0 C link B For transceiver only configurations, the transmitter does not scramble these data before sending to the hard transceiver. Note: For HD-SDI dual link mode only.

Signal	Width	Direction	Description
tx_datain_valid	1	Input	<p>Transmitter parallel data valid. The timing (H: High, L: Low) must be synchronous to tx_pclk clock domain and has the following settings:</p> <ul style="list-style-type: none"> • SD-SDI = 1H 4L 1H 5L • HD-SDI = H • 3G-SDI = H • HD-SDI Dual Link = H • Dual rate = SD (1H 4L 1H 5L); HD (1H 1L) • Triple rate = SD (1H 4L 1H 5L); HD (1H 1L); 3G (H) • Multi rate (up to 12G) = SD (1H 4L 1H 5L); HD (1H 1L); 3G/6G/12G (H) <p>This signal can be driven by the tx_dataout_valid signal for SD-SDI, and dual-rate, triple-rate, and multi-rate modes.</p>
tx_datain_valid_b	1	Input	<p>Transmitter parallel data valid for link B. Applicable for HD-SDI dual link mode only.</p> <p>HD-SDI dual link = H</p> <p>This signal can be driven by the tx_dataout_valid_b signal.</p>
tx_trs	1	Input	<p>Transmitter TRS input. For use in LN, CRC, or payload ID insertion. Assert this signal on first word of both EAV and SAV TRSs.</p> <p>Note: Not applicable for transceiver configurations.</p>
tx_trs_b	1	Input	<p>Transmitter TRS input for link B.</p> <p>Note: For HD-SDI dual link combined or protocol only configurations.</p>
tx_ln	11S	Input	<p>Transmitter line number.</p> <p>Not applicable when you disable the Insert Video Payload ID (SMPTE ST 352M) option in SD-SDI.</p>
tx_ln_b	11S	Input	<p>Transmitter line number for link B. For use in 3G-SDI, HD-SDI dual link, triple-rate, and multi-rate (up to 12G) line number insertion.</p>
tx_dataout	20S	Output	<p>Transmitter parallel data out.</p> <ul style="list-style-type: none"> • Arria V, Cyclone V, and Stratix V devices: Available for transmitter protocol configuration only. • Arria 10 devices: Available whenever TX core is included.
tx_dataout_b	20	Output	<p>Transmitter parallel data out for link B.</p> <p>Note: Applicable for HD-SDI dual link transmitter protocol configuration only.</p>

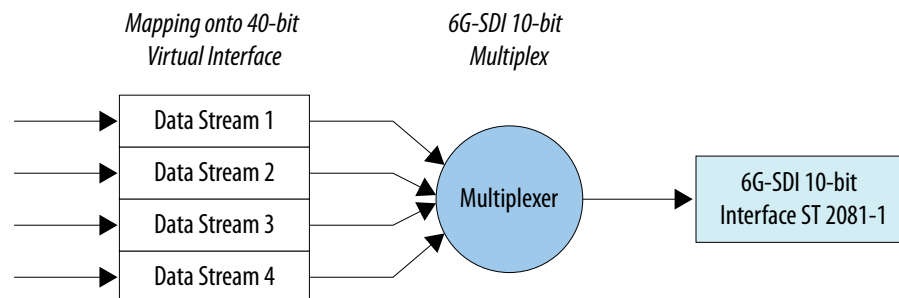
Signal	Width	Direction	Description
tx_dataout_valid	1	Output	Data valid generated by the core. This signal can be used to drive tx_datain_valid. The timing (H: High, L: Low) must be synchronous to tx_pclk clock domain and have the following settings: <ul style="list-style-type: none"> SD-SDI = 1H 4L 1H 5L HD-SDI = H 3G-SDI = H HD-SDI Dual Link = H Dual rate = SD (1H 4L 1H 5L); HD (1H 1L) Triple rate = SD (1H 4L 1H 5L); HD (1H 1L); 3G (H) Multi rate (up to 12G) = SD (1H 4L 1H 5L); HD (1H 1L); 3G/6G/12G (H)
tx_dataout_valid_b	1	Output	Data valid generated by the core for link B. The timing (H: High, L: Low) is identical to the tx_dataout_valid signal and is synchronous to tx_pclk clock domain. Note: Applicable for HD-SDI dual link mode only.
tx_std_out	3	Output	Indicates the transmitted video standard. This signal connects to tx_std in the transceiver only configuration. Note: Applicable for 3G-SDI, dual-rate, and triple-rate transmitter protocol only configuration. Not applicable for Arria 10 devices.
tx_vpid_overwrite	1	Input	When a payload ID is embedded in the video stream, the core enables this signal to overwrite the existing payload ID. No effect when disabled. Applicable only when you enable the Insert Payload ID (SMPTE ST 352M) option.
tx_vpid_byte1	8S	Input	The core inserts payload ID byte 1. Applicable only when you enable the Insert Payload ID (SMPTE ST 352M) option.
tx_vpid_byte2	8S	Input	The core inserts payload ID byte 2. Applicable only when you enable the Insert Payload ID (SMPTE ST 352M) option.
tx_vpid_byte3	8S	Input	The core inserts payload ID byte 3. Applicable only when you enable the Insert Payload ID (SMPTE ST 352M) option.
tx_vpid_byte4	8S	Input	The core inserts payload ID byte 4. Applicable only when you enable the Insert Payload ID (SMPTE ST 352M) option.

Signal	Width	Direction	Description
tx_vpid_byte1_b	8S	Input	The core inserts payload ID byte 1 for link B. For 3G-SDI, HD-SDI dual link, triple-rate, and multi-rate (up to 12G) modes only. Applicable only when you enable the Insert Payload ID (SMPTE ST 352M) option.
tx_vpid_byte2_b	8S	Input	The core inserts payload ID byte 2 for link B. For 3G-SDI, HD-SDI dual link triple-rate, and multi-rate (up to 12G) modes only. Applicable only when you enable the Insert Payload ID (SMPTE ST 352M) option.
tx_vpid_byte3_b	8S	Input	The core inserts payload ID byte 3 for link B. For 3G-SDI, HD-SDI dual link, triple-rate, and multi-rate (up to 12G) modes only. Applicable only when you enable the Insert Payload ID (SMPTE ST 352M) option.
tx_vpid_byte4_b	8S	Input	The core inserts payload ID byte 4 for link B. For 3G-SDI, HD-SDI dual link, triple-rate, and multi-rate (up to 12G) modes only. Applicable only when you enable the Insert Payload ID (SMPTE ST 352M) option.
tx_line_f0	11S	Input	Line number of field 0 (F0) of inserted payload ID. Applicable only when you enable the Insert Payload ID (SMPTE ST 352M) option.
tx_line_f1	11S	Input	Line number of field 1 (F1) of inserted payload ID. Applicable only when you enable the Insert Payload ID (SMPTE ST 352M) option.

Image Mapping

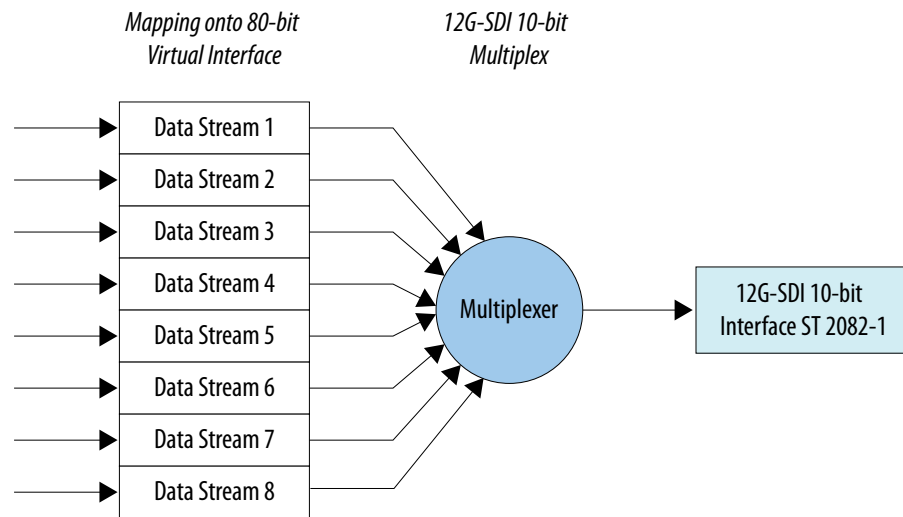
Image mapping differs for 6G-SDI and 12-SDI interfaces.

Figure 6-5: Transmitting 6G-SDI Image



For instance, if you are transmitting image per ST 2081-10 Mode 1 mapping, each data stream should be C, Y multiplex of each sub image.

Figure 6-6: Transmitting 12G-SDI Image



For instance, if you are transmitting image per ST 2082-10 mode 1 mapping, each odd data stream should be Y samples of each sub image, while the even data stream should be C samples of each sub image.

Receiver Protocol Signals

Table 6-3: Receiver Protocol Signals—Synchronous to `rx_coreclk`

Signal	Width	Direction	Description
<code>rx_coreclk_is_ntsc_paln</code>	1	Input	Indicates to the receiver core if <code>rx_coreclk</code> or <code>rx_coreclk_hd</code> is at NSTC (1/1.001) or PAL (1) rate. This signal is required for the receiver core to detect the incoming video rate as NTSC or PAL. <ul style="list-style-type: none"> 0 = PAL rate (when <code>rx_coreclk</code> = 148.5 MHz or <code>rx_coreclk_hd</code> = 74.25 MHz) 1 = NTSC rate (when <code>rx_coreclk</code> = 148.35 MHz or <code>rx_coreclk_hd</code> = 74.175 MHz) Note: Not applicable for SD-SDI and protocol only configurations.
<code>rx_std_in</code>	3	Input	Indicates to the receiver core protocol block the video standard received by the transceiver block. Note: Applicable for 3G-SDI, dual-rate, and triple-rate receiver protocol only configurations. Not applicable for Arria 10 devices.

Signal	Width	Direction	Description
rx_clkout_is_ntsc_paln	1	Output	<p>Indicates that the receiver core is receiving video rate at NSTC (1/1.001) or PAL (1).</p> <ul style="list-style-type: none"> 0 = PAL rate (when rx_coreclk = 148.5 MHz or rx_coreclk_hd = 74.25 MHz) 1 = NTSC rate (when rx_coreclk = 148.35 MHz or rx_coreclk_hd = 74.175 MHz) <p>Note: Not applicable for SD-SDI and protocol only modes.</p>
rx_std (for transceiver only configurations)	3	Output	<p>Receiver video standard.</p> <ul style="list-style-type: none"> 3'b000: SD-SDI 3'b001: HD-SDI 3'b011: 3G-SDI <p>Note: Applicable for 3G-SDI, dual-rate, and triple-rate configurations only. Not applicable for Arria 10 devices.</p>

Table 6-4: Receiver Protocol Signals—Synchronous to rx_clkout or Parallel Recovered Clock from Transceiver

Note: S = Indicates the number of 20-bit interfaces; 4 for multi-rate (up to 12G) mode and 1 for other modes.

Signal	Width	Direction	Description
rx_datain	20S	Input	<p>Receiver parallel data from the transceiver.</p> <p>For Arria 10 devices, this signal is directly connected to the rx_parallel_data signal from the transceiver.</p> <p>Note: If you are not enabling the simplified data interface, refer to the Arria 10 Transceiver parameter editor or the <i>Arria 10 Transceiver PHY IP Core User Guide</i> for proper data bit mapping.</p> <p>For older supported devices, this signal is directly connected to the rx_dataout signal from the SDI receiver in transceiver mode.</p> <p>Note: Available only in protocol mode.</p>
rx_datain_b	20	Input	<p>Receiver parallel data from the transceiver for link B.</p> <p>This signal is directly connected to the rx_dataout_b signal from the SDI receiver in transceiver mode.</p> <p>Note: Applicable for HD-SDI dual link protocol only configuration.</p>

Signal	Width	Direction	Description
rx_datain_valid	1	Input	<p>Data valid from the oversampling logic. Assertion of this signal indicates the current data on rx_datain is valid. The timing (H: High, L: Low) for each video standard has the following settings:</p> <ul style="list-style-type: none"> • SD-SDI = 1H 4L 1H 5L • HD-SDI = H • 3G-SDI = H • HD-SDI Dual Link = H • Dual rate = SD (1H 4L 1H 5L); HD (H) • Triple rate = SD (1H 4L 1H 5L); HD (H); 3G (H) • Multi rate (up to 12G) = SD (1H 4L 1H 5L); HD (H); 3G/6G/12G (H) <p>This signal is directly connected to the rx_dataout_valid signal from the SDI receiver in transceiver mode.</p> <p>Note: Applicable for protocol only configuration. Not applicable for Arria 10 devices.</p>
rx_datain_valid_b	1	Input	<p>Data valid from the oversampling logic. Assertion of this signal indicates the current data on rx_datain_b is valid.</p> <p>This signal is directly connected to the rx_dataout_valid_b signal from the SDI receiver in transceiver mode.</p> <p>Note: Applicable for HD-SDI dual link receiver protocol only configuration. Not applicable for Arria 10 devices.</p>
rx_trs_loose_lock_in	1	Input	<p>Indicates that the receiver protocol block detects a single and valid TRS locking signal. This signal must be driven by rx_trs_loose_lock_out of the receiver protocol block.</p> <p>Note: Applicable for receiver transceiver configuration only. Not applicable for Arria 10 devices.</p>
rx_trs_loose_lock_in_b	1	Input	<p>Indicates that the receiver protocol block for link B detects a single and valid TRS locking signal. This signal must be driven by rx_trs_loose_lock_out_b of the receiver protocol block.</p> <p>Note: Applicable for HD-SDI dual link receiver transceiver configuration only. Not applicable for Arria 10 devices.</p>
rx_trs_in	1	Input	<p>The signal driven by rx_trs to indicate to the PHY management block that the receiver protocol block detected a valid TRS.</p> <p>Note: Applicable for receiver transceiver configuration only. Not applicable for Arria 10 devices.</p>

Signal	Width	Direction	Description
rx_dataout	20S	Output	Receiver parallel data out. Only lower 10 bits are valid for SD when SD Interface Bit Width = 10 . In multi-rate mode: <ul style="list-style-type: none"> • HD/3G-SDI: Only lower 20 bits are valid • 6G-SDI: Only lower 40 bits are valid For bit ordering, refer to tx_datain signal description.
rx_dataout_b	20	Output	Parallel data out signal for the receiver (link B). Applicable only for HD-SDI dual link configuration. Note: Applicable for HD-SDI dual link configuration only.
rx_dataout_valid	1	Output	Data valid from the oversampling logic. The receiver asserts this signal to indicate current data on rx_dataout is valid. The timing (H: High, L: Low) for each video standard has the following settings: <ul style="list-style-type: none"> • SD-SDI = 1H 4L 1H 5L • HD-SDI = H • 3G-SDI = H • HD-SDI Dual Link = H • Dual rate = SD (1H 4L 1H 5L); HD (H) • Triple rate = SD (1H 4L 1H 5L); HD (H); 3G (H) • Multi rate (up to 12G) = SD (1H 4L 1H 5L); HD (H); 3G/6G/12G (H) The 1H4L 1H5L cadence for SD-SDI repeats indefinitely in an ideal case where the video source clock matches the CDR reference clock source. In a typical scenario, you may observe the cadence being shifted periodically (for instance, 1H4L 1H5L 1H5L 1H4L).
rx_dataout_valid_b	1	Output	Data valid from the oversampling logic. The receiver asserts this signal to indicate current data on rx_dataout_b is valid. The timing (H: High, L: Low) for each video standard is identical to the rx_dataout_valid signal. Note: Applicable for HD-SDI dual link configuration only.
rx_f	1S	Output	Field bit timing signal. This signal indicates which video field is currently active. For interlaced frame, 0 means first field (F0) while 1 means second field (F1). For progressive frame, the value is always 0.
rx_v	1S	Output	Vertical blanking interval timing signal. The receiver asserts this signal when the vertical blanking interval is active.

Signal	Width	Direction	Description
rx_h	1S	Output	Horizontal blanking interval timing signal. The receiver asserts this signal when the horizontal blanking interval is active.
rx_ap	1S	Output	Active picture interval timing signal. The receiver asserts this signal when the active picture interval is active.
rx_std	3	Output	Receiver video standard. <ul style="list-style-type: none"> 3'b000: SD-SDI 3'b001: HD-SDI 3'b011: 3G-SDI Level A 3'b010 3G-SDI Level B 3'b101: 6G-SDI 4 Streams Interleaved 3'b100: 6G-SDI 8 Streams Interleaved 3'b111: 12G-SDI 8 Streams Interleaved 3'b110: 12G-SDI 16 Streams Interleaved Note: Applicable for 3G-SDI, dual-rate, triple-rate, and multi-rate configurations.
rx_format	4S	Output	Indicates the format for the received video transport. Refer to rx_format on page 6-21 for more information about the video format values.
rx_eav	1S	Output	Receiver output that indicates current TRS is EAV. This signal is asserted at the fourth word of TRS, which is the XYZ word.
rx_trs	1S	Output	Receiver output that indicates current word is TRS. This signal is asserted at the first word of 3FF 000 000 TRS.
rx_ln	11S	Output	Receiver line number output. Note: Applicable for all modes except SD-SDI.
rx_ln_b	11S	Output	Receiver line number output for link B. Note: Applicable for 3G-SDI, HD-SDI dual link, triple-rate, and multi-rate (up to 12G) modes only.
rx_align_locked	1	Output	Alignment locked, indicating that a TRS has been spotted and word alignment is performed.
rx_align_locked_b	1	Output	Alignment locked for link B, indicating that a TRS has been spotted and word alignment is performed. Note: Applicable for HD-SDI dual link configuration only.
rx_trs_locked	1S	Output	TRS locked, indicating that six consecutive TRSs with same timing has been spotted.

Signal	Width	Direction	Description
rx_trs_locked_b	1	Output	TRS locked for link B, indicating that six consecutive TRSs with same timing has been spotted. Note: Applicable for HD-SDI dual link configuration only.
rx_frame_locked	1	Output	Frame locked, indicating that multiple frames with same timing has been spotted.
rx_frame_locked_b	1	Output	Frame locked for link B, indicating that multiple frames with same timing has been spotted. Note: Applicable for HD-SDI dual link configuration only.
rx_dl_locked	1	Output	Dual link locked, indicating that both ports are aligned. Note: Applicable for HD-SDI dual link configuration only.
rx_trs_loose_lock_out	1	Output	Indicates that the receiver protocol block detects a single and valid TRS locking signal. This signal must be used to drive rx_trs_loose_lock_in of the receiver transceiver block. Note: Applicable for protocol only configuration. Not applicable for Arria 10 devices.
rx_trs_loose_lock_out_b	1	Output	Indicates that the receiver protocol block for link B detects a single and valid TRS locking signal. This signal must be used to drive rx_trs_loose_lock_in_b of the receiver transceiver block. Note: Applicable for HD-SDI dual link protocol only configuration. Not applicable for Arria 10 devices.
rx_crc_error_c	1S	Output	CRC error on chroma channel. Applicable only when you enable CRC checking. Note: Applicable for all modes except SD-SDI.
rx_crc_error_y	1S	Output	CRC error on luma channel. Note: Applicable only when you enable CRC checking. Applicable for all modes except SD-SDI.
rx_crc_error_c_b	1S	Output	CRC error on chroma channel for link B. Note: Applicable only when you enable CRC checking. Applicable for 3G-SDI, HD-SDI dual link, triple-rate, and multi-rate modes only.
rx_crc_error_y_b	1S	Output	CRC error on luma channel for link B. Applicable only when you enable CRC checking. Note: Applicable for 3G-SDI, HD-SDI dual link, triple-rate, and multi-rate modes only.

Signal	Width	Direction	Description
rx_vpid_byte1	8S	Output	The core extracts payload ID byte 1. Applicable only when you enable the Extract Payload ID (SMPTE ST 352M) option.
rx_vpid_byte2	8S	Output	The core extracts payload ID byte 2. Applicable only when you enable the Extract Payload ID (SMPTE ST 352M) option.
rx_vpid_byte3	8S	Output	The core extracts payload ID byte 3. Applicable only when you enable the Extract Payload ID (SMPTE ST 352M) option.
rx_vpid_byte4	8S	Output	The core extracts payload ID byte 4. Applicable only when you enable the Extract Payload ID (SMPTE ST 352M) option.
rx_vpid_valid	1S	Output	Indicates that the extracted payload ID is valid. Applicable only when you enable the Extract Payload ID (SMPTE ST 352M) option.
rx_vpid_checksum_error	1S	Output	Indicates that the extracted payload ID has a checksum error. Applicable only when you enable the Extract Payload ID (SMPTE ST 352M) option.
rx_vpid_byte1_b	8S	Output	The core extracts payload ID byte 1 for link B. Applicable only when you enable the Extract Payload ID (SMPTE ST 352M) option. Note: Applicable for 3G-SDI, HD-SDI dual link, triple-rate, and multi-rate (up to 12G) modes only.
rx_vpid_byte2_b	8S	Output	The core extracts payload ID byte 2 for link B. Applicable only when you enable the Extract Payload ID (SMPTE ST 352M) option. Note: Applicable for 3G-SDI, HD-SDI dual link, triple-rate, and multi-rate (up to 12G) modes only.
rx_vpid_byte3_b	8S	Output	The core extracts payload ID byte 3 for link B. Applicable only when you enable the Extract Payload ID (SMPTE ST 352M) option. Note: Applicable for 3G-SDI, HD-SDI dual link, triple-rate, and multi-rate (up to 12G) modes only.

Signal	Width	Direction	Description
rx_vpid_byte4_b	8S	Output	The core extracts payload ID byte 4 for link B. Applicable only when you enable the Extract Payload ID (SMPTE ST 352M) option. Note: Applicable for 3G-SDI, HD-SDI dual link, triple-rate, and multi-rate (up to 12G) modes only.
rx_vpid_valid_b	1S	Output	Indicates that the extracted payload ID for link B is valid. Applicable only when you enable the Extract Payload ID (SMPTE ST 352M) option. Note: Applicable for 3G-SDI, HD-SDI dual link, triple-rate, and multi-rate (up to 12G) modes only.
rx_vpid_checksum_error_b	1S	Output	Indicates that the extracted payload ID for link B has a checksum error. Applicable only when you enable the Extract Payload ID (SMPTE ST 352M) option. Note: Applicable for 3G-SDI, HD-SDI dual link, triple-rate, and multi-rate (up to 12G) modes only.
rx_line_f0	11S	Output	Line number of field 0 (F0) of the payload ID location. Requires two complete frames to update this signal. Applicable only when you enable the Extract Video Payload ID (SMPTE 352M) option.
rx_line_f1	11S	Output	Line number of field 1 (F1) of the payload ID location. Requires two complete frames to update this signal. Applicable only when you enable the Extract Video Payload ID (SMPTE 352M) option.

rx_format

The format represents only the video transport format; not the picture format. For example, when the core transports 1080p50 video on HD-SDI dual link, the video transport format is 1080i50.

Table 6-5: Video Format Values

Encoding Value	SMPTE Standard	Active Lines Per Frame	Transport Format	Frame Rate
0000	SMPTE259M	486	I	29.97
0001	SMPTE259M	576	I	25
0100	SMPTE274M	1080	I	30/29.97/60/59.94 ⁽²⁾

⁽²⁾ Frame rates 60 and 59.94 are meant for 3G Level B/HD Dual Link when receiving 1080p60/59.94 format.

Encoding Value	SMPTE Standard	Active Lines Per Frame	Transport Format	Frame Rate
0101	SMPTE274M	1080	I	25/50 ⁽³⁾
0110	SMPTE274M	1080	P	24/23.98
0111	SMPTE296M	720	P	60/59.94
1000	SMPTE296M	720	P	50
1001	SMPTE296M	720	P	30/29.97
1010	SMPTE296M	720	P	25
1011	SMPTE296M	720	P	24/23.98
1100	SMPTE274M	1080	P	30/29.97/60/59.94
1101	SMPTE274M	1080	P	25/50
1110	SMPTE274M	1080	I	24
1111	Undetectable format, revert to default value			
Others	Reserved			

To differentiate video format with 1 and 1/1.001 rate, refer to the `rx_clkout_is_ntsc_paln` output signal. For example, if `rx_format = 0100`, `rx_clkout_is_ntsc_paln = 1`, then the format for the received video is 1080i59.94. Otherwise, it is 1080i60.

To differentiate between video format across HD-SDI and 3G-SDI interfaces, also refer to the `rx_std` output signal. For example, if `rx_format = 1100` and `rx_clkout_is_ntsc_paln = 0`, `rx_std = 01`, then the received video format is 1080p30. If the `rx_std = 11` or `10`, then the received video format is 1080p60.

Note: Altera recommends that you refer to the Payload ID to get the most accurate video format details.

For 6G-SDI or 12G-SDI interfaces, each of the 20-bit interface reports its own detected format. For example, `rx_format` of all four 20-bit interfaces report 1100 (1080p60) when receiving 2160p60 in 12G-SDI, whereas only the lower two interfaces report valid `rx_format` in 6G-SDI.

Transceiver Signals

Table 6-6: Transceiver Serial Data Pins (for Arria V, Cyclone V, and Stratix V Devices)

Signal	Direction	Description
<code>sdi_tx</code>	Output	Transmitter serial out.

⁽³⁾ Frame rate 50 is meant for 3G Level B/HD Dual Link when receiving 1080p50 format.

Signal	Direction	Description
sdi_tx_b	Output	Transmitter serial out for link B. Note: Applicable for HD-SDI dual link configuration only.
sdi_rx	Input	Receiver serial in.
sdi_rx_b	Input	Receiver serial in for link B. Note: Applicable for HD-SDI dual link configuration only.

Table 6-7: Transceiver Signals

Signal	Width	Clock Domain	Direction	Description
xcvr_refclk_sel	1	tx_coreclk	Input	Transceiver reference clock select signal that selects which clock to be used. <ul style="list-style-type: none"> 0 = xcvr_refclk 1 = xcvr_refclk_alt Applicable only when you enable the TX PLL Dynamic Switching option. Note: Not applicable for Arria 10 devices.
tx_pll_locked	1	-	Output	PLL locked signal (TX PLL0) for the hard transceiver. Note: Not applicable for Arria 10 devices.
tx_pll_locked_alt	1	-	Output	PLL locked signal (TX PLL1) for the hard transceiver. Applicable only when you enable the TX PLL Dynamic Switching option. Note: Not applicable for Arria 10 devices.
reconfig_to_xcvr	70N	-	Input	Dynamic reconfiguration input for the hard transceiver, where N is the reconfiguration interface. <ul style="list-style-type: none"> N = 1 for receiver N = 2 for transmitter and bidirectional Note: Not applicable for Arria 10 devices.

Signal	Width	Clock Domain	Direction	Description
reconfig_to_xcvr_b	70N	-	Input	Dynamic reconfiguration input for the hard transceiver, where N is the reconfiguration interface. <ul style="list-style-type: none"> N = 1 for receiver N = 2 for transmitter and bidirectional Note: For HD-SDI dual link configuration only. Not applicable for Arria 10 devices.
reconfig_from_xcvr	46N	-	Output	Dynamic reconfiguration output for the hard transceiver, where N is the reconfiguration interface. <ul style="list-style-type: none"> N = 1 for receiver N = 2 for transmitter and bidirectional Note: Not applicable for Arria 10 devices.
reconfig_from_xcvr_b	46N	-	Output	Dynamic reconfiguration output for the hard transceiver, where N is the reconfiguration interface. <ul style="list-style-type: none"> N = 1 for receiver N = 2 for transmitter and bidirectional Note: For HD-SDI dual link configuration only. Not applicable for Arria 10 devices.
rx_sdi_start_reconfig	1	rx_coreclk	Output	Request to start dynamic reconfiguration. Note: Applicable for dual rate, triple-rate, and multi-rate modes only.
rx_sdi_reconfig_done	1	-	Input	Indicates that dynamic reconfiguration has completed. Note: Applicable for dual rate, triple-rate, and multi-rate modes only.

Signal	Width	Clock Domain	Direction	Description
rx_ready	1	-	Input	Status signal from the transceiver reset controller to indicate when Rx PHY sequence is complete. Note: Applicable only for Arria 10 devices.
gxb_ltr	1	rx_coreclk	Output	Control signal to the transceiver rx_set_locktoref input signal. Assertion of this signal programs the Rx CDR to lock manually to reference mode. Note: Applicable only for Arria 10 devices.
gxb_ltd	1	rx_coreclk	Output	Control signal to the transceiver rx_set_locktodata input signal. Note: Applicable only for Arria 10 devices.

2016.12.20

UG-01125



Subscribe



Send Feedback

There are several considerations that require your attention to ensure the success of your designs.

Transceiver Handling Guidelines

The Arria V, Cyclone V, and Stratix V devices handle transceivers differently than the Arria 10 devices.

Handling Transceiver in Arria V, Cyclone V, and Stratix V Devices

In the Arria V, Cyclone V, and Stratix V design example, you can expand the transceiver to multiple channels.

The generated design example consists of two SDI channels, where the SDI duplex instance always occupy Channel 0 (Ch0), while the SDI instance at Channel 1 (Ch1) depends on your selection from the parameter editor. To expand and accommodate more channels, you must perform some modifications to the source files.

For example, when Ch0 is duplex, Ch1 is RX and TX, if you want to instantiate an additional SDI duplex instance at Channel 2 (Ch2), you need to make some modifications to the following components.

© 2016 Intel Corporation. All rights reserved. Intel, the Intel logo, Altera, Arria, Cyclone, Enpirion, MAX, NIOS, Quartus and Stratix words and logos are trademarks of Intel Corporation in the US and/or other countries. Other marks and brands may be claimed as the property of others. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO
9001:2008
Registered

ALTERA
now part of Intel

Modifying the Transceiver Reconfiguration Controller

Perform the following changes to modify the transceiver reconfiguration controller:

- Edit the **Number_of_reconfig_interfaces** parameter. This parameter specifies the total number of reconfiguration interfaces that connect to this block.
- Each channel or TX PLL needs one reconfiguration interface. Therefore, an SDI duplex or SDI TX mode requires two interfaces while an SDI RX mode requires only one interface. If you enable the dynamic TX clock switching feature, the SDI duplex or SDI TX mode requires three interfaces. The additional interface is for the additional TX PLL. For simplicity, assume this option is disabled.
- Determine the total number of reconfiguration interfaces required in your design and assign the parameter value accordingly. In this design example, the total number of reconfiguration interfaces is 7 (Ch0=2, Ch1=3 and Ch2=2).
- Link the `reconfig_to_xcvr` and `reconfig_from_xcvr` signals from the additional SDI duplex instance at Ch2. You must link the signals in the order of the logical channel number (`rx_log_ch_num` and `tx_log_ch_num`) in the reconfiguration logic source file (`sdi_ii_reconfig_logic.v`).
- In the design example that instantiates the transceiver reconfiguration controller, add the wire connection between the additional SDI duplex instance at Ch2 and the transceiver reconfiguration controller as shown below:

```
wire [ 139:0] reconfig_to_xcvr_du_ch2;
wire [  91:0] reconfig_from_xcvr_du_ch2;
wire [ 139:0] reconfig_to_xcvr_tx_ch1;
wire [  69:0] reconfig_to_xcvr_rx_ch1;
wire [  91:0] reconfig_from_xcvr_tx_ch1;
wire [  45:0] reconfig_from_xcvr_rx_ch1;
wire [ 139:0] reconfig_to_xcvr_du_ch0;
wire [  91:0] reconfig_from_xcvr_du_ch0;

alt_xcvr_reconfig #(
    .number_of_reconfig_interfaces (7),
    ...
) u_reconfig (
    .reconfig_to_xcvr    ({reconfig_to_xcvr_du_ch2,
                          reconfig_to_xcvr_tx_ch1,
                          reconfig_to_xcvr_rx_ch1,
                          reconfig_to_xcvr_du_ch0}),
    .reconfig_from_xcvr ({reconfig_from_xcvr_du_ch2,
                          reconfig_from_xcvr_tx_ch1,
                          reconfig_from_xcvr_rx_ch1,
                          reconfig_from_xcvr_du_ch0}),
);
```

Modifying the Reconfiguration Management

Perform the following changes to modify the reconfiguration management:

- Edit the **Number_of_channels** parameter in `sdi_ii_ed_reconfig_mgmt.v`. This parameter value should be the total number of the SDI RX channels declared in the design. In this example, the `NUM_CHS` is 3.
- Link the interface signals—`sdi_rx_start_reconfig`, `sdi_rx_reconfig_done`, and `sdi_rx_std`—between multiple SDI instances and reconfiguration management block. Link the interface signals—`sdi_tx_start_reconfig`, `sdi_tx_reconfig_done`, and `sdi_tx_pll_sel`—between user and reconfiguration management block. You must link the signals in the order of the logical channel number (`rx_log_ch_num` and `tx_log_ch_num`) in the reconfiguration logic source file (`sdi_ii_reconfig_logic.v`). For example:

```
wire tx_start_reconfig_ch2,tx_start_reconfig_ch1,tx_start_reconfig_ch0;
wire tx_pll_sel_ch2,tx_pll_sel_ch1,tx_pll_sel_ch0;
```

```
wire tx_reconfig_done_ch2,tx_reconfig_done_ch1,tx_reconfig_done_ch0;
wire rx_start_reconfig_ch2,rx_start_reconfig_ch1,rx_start_reconfig_ch0;
wire [1:0] rx_std_ch2, rx_std_ch1,rx_std_ch0;
wire rx_reconfig_done_ch2,rx_reconfig_done_ch1,rx_reconfig_done_ch0;

sdi_ii_ed_reconfig_mgmt #(
    . NUM_CHS (3),
) u_reconfig_mgmt (
    .sdi_tx_start_reconfig (tx_start_reconfig_ch2,
tx_start_reconfig_ch1,tx_start_reconfig_ch0),
    .sdi_tx_pll_sel (tx_pll_sel_ch2,tx_pll_sel_ch1,tx_pll_sel_ch0),
    .sdi_tx_reconfig_done (tx_reconfig_done_ch2,
tx_reconfig_done_ch1,tx_reconfig_done_ch0),
    .sdi_rx_start_reconfig (rx_start_reconfig_ch2,
rx_start_reconfig_ch1,rx_start_reconfig_ch0),
    .sdi_rx_std (rx_std_ch2,rx_std_ch1,rx_std_ch0),
    .sdi_rx_reconfig_done (rx_reconfig_done_ch2,
rx_reconfig_done_ch1,rx_reconfig_done_ch0)
)

```

- In the reconfiguration logic source file, the default setting for the wire rx_log_ch_num is 0 and 2 for channel 0 and channel 1, respectively. The default setting for the wire tx_log_ch_num is 0 and 2 (duplex) or 3 (TX) for channel 0 and channel 1, respectively. These numbers are referring to the **Number_of_channels** parameter value that was set in the transceiver reconfiguration controller. The logical channel number for each SDI channel is as listed in the table below.

Table 7-1: Logical Channel Number for Each SDI Channel

SDI Channel	Direction	Number of Reconfiguration Interfaces	Logical Channel Number
0	Duplex	2	<ul style="list-style-type: none"> • 0: RX/TX channel • 1: Tx PLL
1	RX and TX	3 (1 for RX and 2 for TX)	<ul style="list-style-type: none"> • 2: RX channel • 3: TX channel • 4: TX PLL
2	Duplex	2	<ul style="list-style-type: none"> • 5: RX/TX channel • 6: TX PLL

- Edit the reconfiguration logic source file to assign the logical channel number for the additional SDI duplex instance, which occupies the SDI Ch2. The logical channel number specified in the source file is the reconfiguration interface that is intended for dynamic reconfiguration. For example, if TX channel is intended for dynamic reconfiguration, tx_log_ch_num[2] should be 5.

```
wire [7:0] rx_log_ch_num [0:NUM_CHS-1];
    assign rx_log_ch_num[0] = 8'd0; // Duplex Rx channel share same
        logical channel number with Tx
    assign rx_log_ch_num[1] = 8'd2; // Rx channel
    assign rx_log_ch_num[2] = 8'd5; // Duplex Rx channel

wire [7:0] tx_log_ch_num [0:NUM_CHS-1];
    assign tx_log_ch_num[0] = 8'd0; // Duplex Tx channel share same
        logical channel number with Rx

```

```
assign tx_log_ch_num[1] = 8'd3; // Tx channel
assign tx_log_ch_num[2] = 8'd5; // Duplex Tx channel
```

Related Information

[Altera Transceiver PHY IP Core User Guide](#)

More information about the transceiver reconfiguration controller logical channel numbering.

Modifying the Reconfiguration Router

For ease of implementation, you can bypass this block by connecting the interface signals—`reconfig_to_xcvr`, `reconfig_from_xcvr`, `sdi_rx_start_reconfig`, `sdi_rx_reconfig_done`, `sdi_rx_std`, `sdi_tx_start_reconfig`, `sdi_tx_reconfig_done`, and `sdi_tx_pll_sel`—directly between the SDI instance and the transceiver reconfiguration controller or the reconfiguration management.

Handling Transceiver in Arria 10 Devices

For Arria 10 design examples, duplicate another transceiver reconfiguration management generated from the design for additional channels.

The Arria 10 Transceiver Native PHY IP core provides the following SDI presets that you can apply to your design.

Table 7-2: SDI Presets in the Arria 10 Native PHY IP Core

Presets	Description
SDI 3G NTSC	<ul style="list-style-type: none"> • Preset for 3G-SDI single rate (TX and RX) and triple rate TX • Set for SDI data rate factor of 1/1.001 • Configured in Duplex mode <p>You may change the direction based on your design needs.</p>
SDI 3G PAL	<ul style="list-style-type: none"> • Preset for 3G-SDI single rate (TX and RX) and triple rate TX • Set for SDI data rate factor of 1/1 • Configured in Duplex mode <p>You may change the direction based on your design needs.</p>
SDI HD NTSC	<ul style="list-style-type: none"> • Preset for HD-SDI single rate and HD-SDI dual link (TX and RX) • Set for SDI data rate factor of 1/1.001 • Configured in Duplex mode <p>You may change the direction based on your design needs.</p>
SDI HD PAL	<ul style="list-style-type: none"> • Preset for HD-SDI single rate and HD-SDI dual link (TX and RX) • Set for SDI data rate factor of 1/1 • Configured in Duplex mode <p>You may change the direction based on your design needs.</p>

Presets	Description
SDI Multi rate (up to 12G) Rx	<ul style="list-style-type: none"> • Preset for multi rate up to 12G-SDI (RX) • Contains multiple profiles for HD-SDI, 3G-SDI, 6G-SDI, and 12G-SDI for dynamic reconfiguration
SDI Multi rate (up to 12G) Tx	<ul style="list-style-type: none"> • Preset for multi rate up to 12G-SDI (TX) • Configured in data rate of 11,880 Mbps • Change the data rate to 11,868 Mbps to transmit with data rate factor of 1/1.001
SDI Triple rate Rx	<ul style="list-style-type: none"> • Preset for triple rate up to 3G-SDI (RX) • Contains multiple profiles for HD-SDI and 3G-SDI for dynamic reconfiguration

Merging Simplex Mode Transceiver in the Same Channel

To merge simplex mode transceiver in the same channel, add the following commands in the Quartus Settings File (.qsf) in your project directory:

- `set_instance_assignment -name XCVR_RECONFIG_GROUP 1 -to <tx_serial_pin>`
- `set_instance_assignment -name XCVR_RECONFIG_GROUP 1 -to <rx_serial_pin>`

For more details about merging transceivers, refer to section Dynamic Reconfiguration Interface Merging Across Multiple IP Blocks in Arria 10 Transceiver PHY User Guide.

Using Generated Arria 10 Reconfiguration Management for Triple and Multi Rates

You may encounter the following errors when you use the generated reconfiguration management block from the Quartus Prime software:

- Error (10161): Verilog HDL error at rcfg_sdi_cdr.sv: object "altera_xcvr_native_a10_reconfig_parameters_CFG0" is not declared. Verify the object name is correct. If the name is correct, declare the object.
- Error (10161): Verilog HDL error at rcfg_sdi_cdr.sv: object "altera_xcvr_native_a10_reconfig_parameters_CFG1" is not declared. Verify the object name is correct. If the name is correct, declare the object.

The reconfiguration management block requires the CFG files that are generated from the transceiver to determine which registers to be reconfigured for data rate changes. However, the Quartus Prime software cannot recognize these files outside of the transceiver library files.

To resolve this issue, add the library switch to the `rcfg_sdi_cdr.sv` file in your project's .qsf.

```
set_global_assignment -name SYSTEMVERILOG_FILE <file hierarchy before the file>/
rcfg_sdi_cdr.sv -library <phy_name quartus version>
```

1. Find the exact *library* name that you should assign in the transceiver .qip file.
2. Open the transceiver .qip file and search for the string: `parameter_CFG0`.

You should see: `set_global_assignment -library <phy_name quartus version> -name SYSTEMVERILOG_FILE ...CFG0.sv`.

Ensuring Independent RX and TX Operations in the Same Channel

The `rx_cal_busy` and `tx_cal_busy` signals from the transceiver are from the same internal node and change state concurrently during calibration. Because these signals are from the same internal node, the RX and TX transceivers in the same channel are affected by each other when one transceiver is in calibration. Problems may occur when the RX and TX transceivers in the same channel are required to work independently, because the TX will be held in reset when the RX is being recalibrated or vice versa.

A possible workaround for this problem is to use the transceiver arbiter from the generated design example. For more details about the arbiter's signal interface, refer to the *SDI II IP Core Design Example User Guide*.

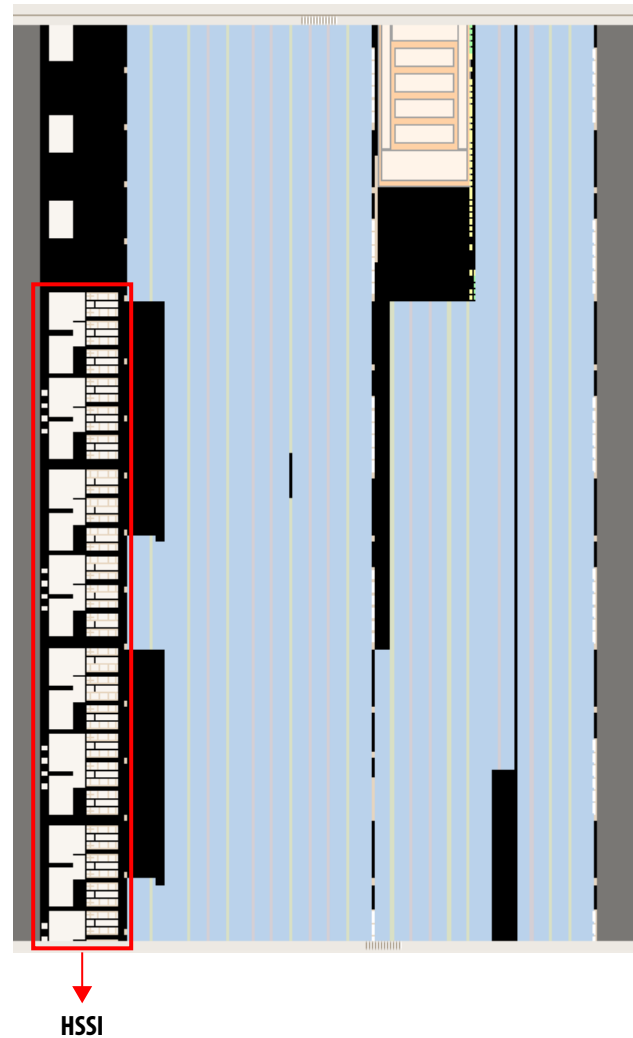
Related Information

[Capability Registers](#)

Provides more information about capability registers.

Potential Routability Issue During Fitter Stage

The SDI II IP core need to be paired with HSSI channels. For certain Arria 10 device parts, all the HSSI channels reside at one side of the chip. Multiple instantiations of the SDI II IP core in a design (especially for multi-rate mode) may cause that side of the chip to be congested with the ALMs and core logic.

Figure 7-1: Chip Planner View of HSSI Channels Placement on an Arria 10 Device

The Arria 10 architecture is designed to place most HSSI clocks on the peripheral clocks (PCLKs). The logic of the IP core may not fit efficiently into the available regions covered by the PCLKs, and moving the logic farther away is not ideal because the logic needs to interact with the HSSI channels. These circumstances may cause routability challenge and Fitter failure.

To overcome this issue, check the placement of the HSSI channels on the chip and consider the availability of the resources on that side before starting your design.

Timing Violation

You may avoid some timing violation for Arria V, Cyclone V, and Stratix V designs by editing `.qsf`.

After you create a new project, the Quartus Prime software generates a .qsf. Add the following assignments to the .qsf file to avoid timing violation from the synchronizers.

```
set_instance_assignment -name GLOBAL_SIGNAL OFF -to *|altera_reset_synchronizer:alt_rst_sync_uq1|altera_reset_synchronizer_int_chain_out
```



Design Example for Arria V, Cyclone V, and Stratix V Devices

8

2016.12.20

UG-01125



Subscribe



Send Feedback

Each design example provided with the SDI II IP core is synthesizable.

Note: For Arria 10 design examples, refer to the *SDI II IP Core Design Example User Guide*.

Figure below illustrates the generated design example entity and simulation testbench for Arria V, Cyclone V, and Stratix V devices. This design example consists of a video pattern generator, transceiver reconfiguration controller, reconfiguration management, loopback path, and various SDI blocks occupying two transceiver channels.

© 2016 Intel Corporation. All rights reserved. Intel, the Intel logo, Altera, Arria, Cyclone, Enpirion, MAX, NIOS, Quartus and Stratix words and logos are trademarks of Intel Corporation in the US and/or other countries. Other marks and brands may be claimed as the property of others. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO
9001:2008
Registered

ALTERA
now part of Intel

Figure 8-1: Design Example Entity and Simulation Testbench

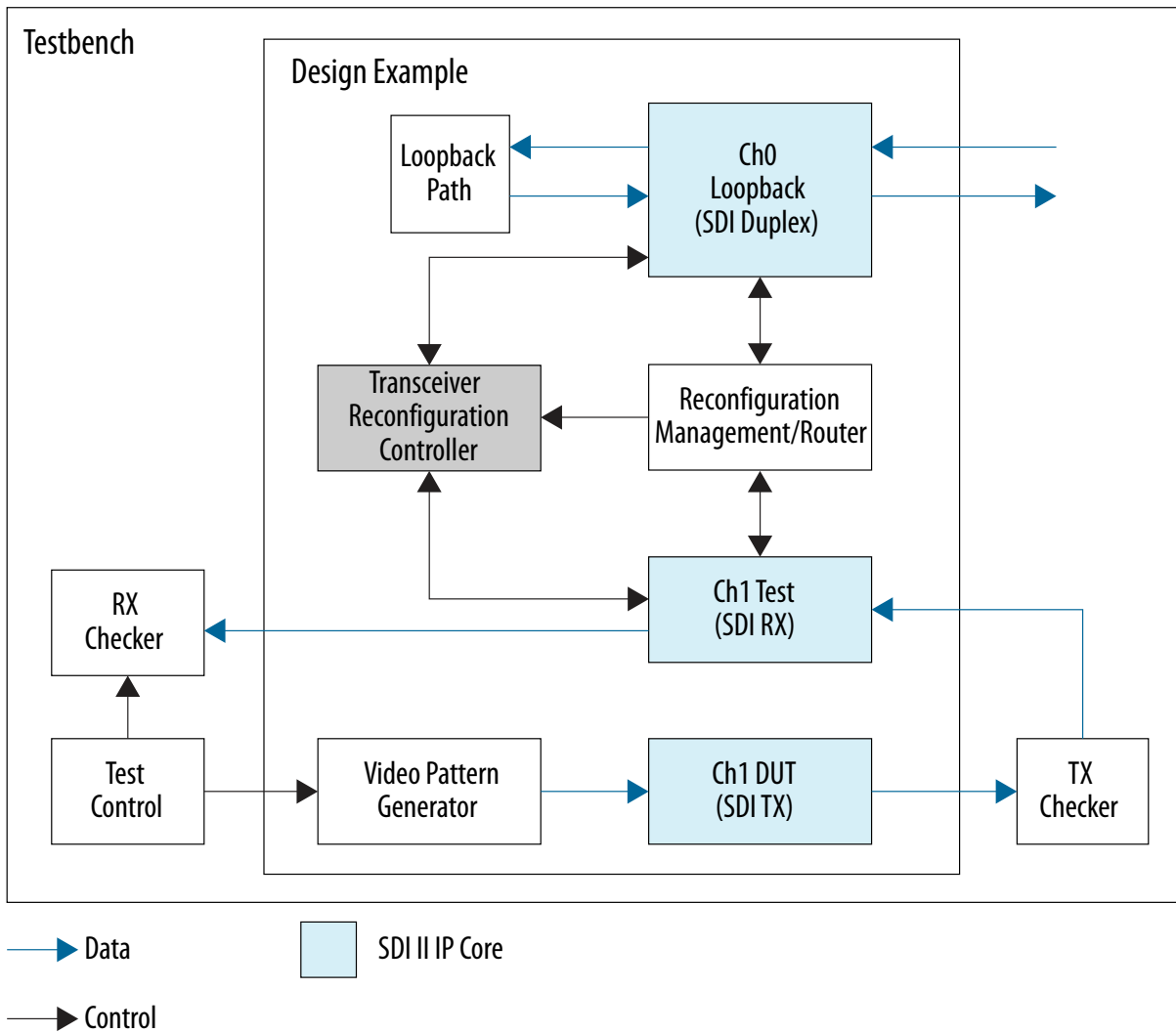


Figure 8-2: Design Example Entity and Simulation Testbench for HD-SDI Dual Link to 3G-SDI (Level B) Conversion

The figure below illustrates the generated design example entity and simulation testbench when you generate HD-SDI dual link receiver with **Convert HD-SDI dual link to 3G-SDI (level B)** option enabled.

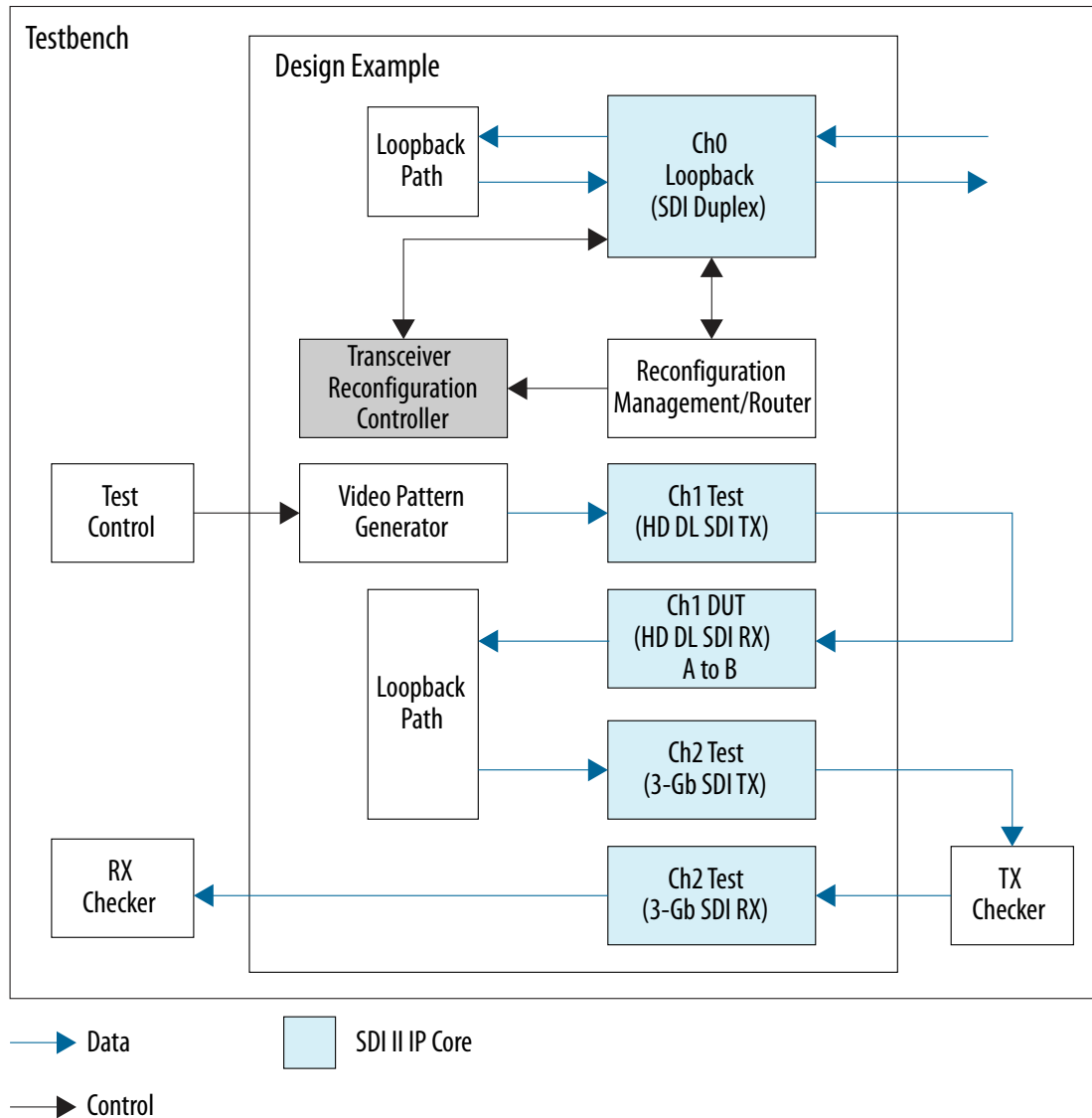
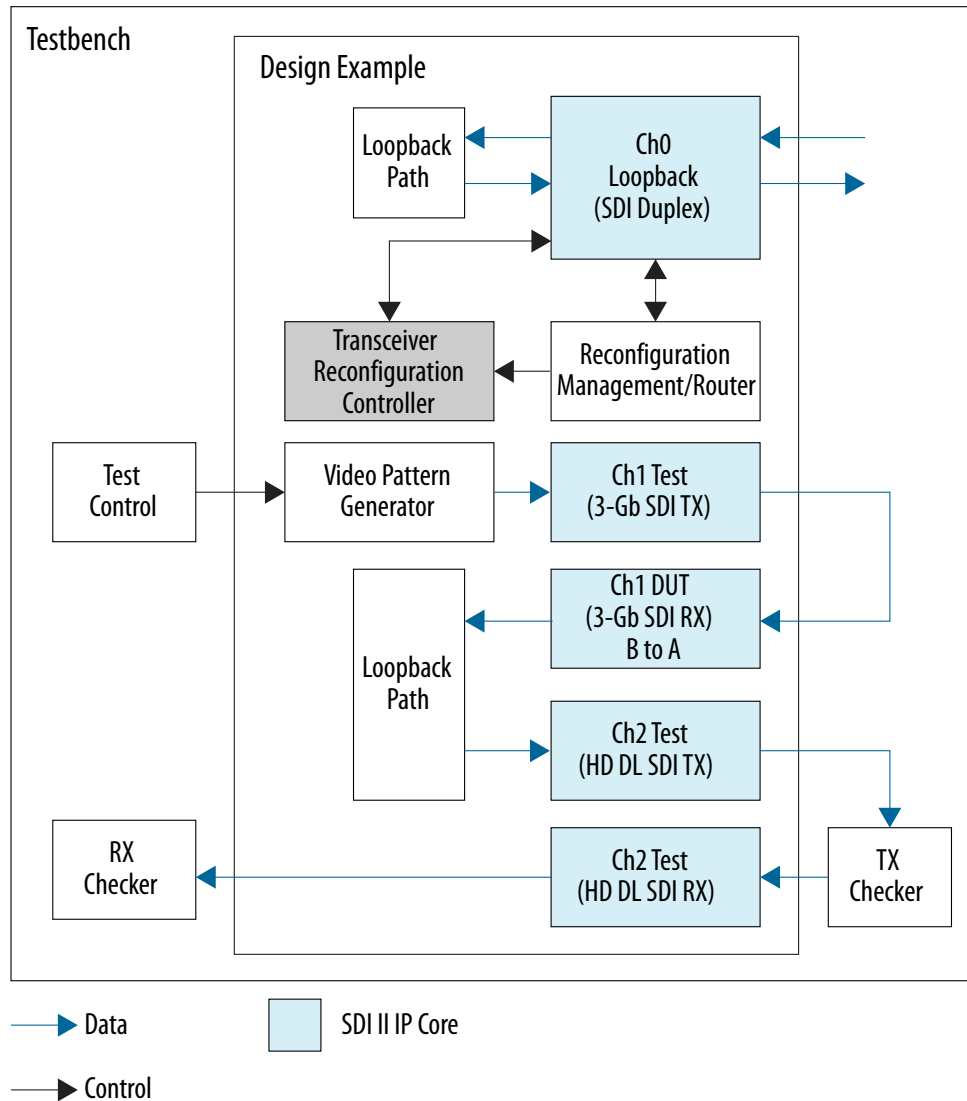


Figure 8-3: Design Example Entity and Simulation Testbench for 3G-SDI (Level B) to HD-SDI Dual Link Conversion

The figure below illustrates the generated design example entity and simulation testbench when you generate 3G-SDI or triple rate SDI receiver with **Convert 3G-SDI (level B) to HD-SDI dual link** option enabled.



Design Example Components

The Arria V, Cyclone V, and Stratix V design examples for the SDI II IP core consist of the following components:

- Video pattern generator
- Transceiver reconfiguration controller
- Reconfiguration management
- Reconfiguration router

Video Pattern Generator

The video pattern generator generates a colorbar or pathological pattern. The colorbar is preferable for image generation while the pathological pattern can stress the PLL and cable equalizer of the attached video equipment. You can configure the video pattern generator to generate various video formats.

Table 8-1: Configuring the Video Pattern Generator to Generate Different Video Formats

Table below lists the examples of how to configure the video pattern generator signals to generate a video format that you desire.

Example	Video Format	Interface	Signal		
			pattgen_tx_std	pattgen_tx_format	pattgen_dl_mapping
Example 1: Generate 1080i video format	1080i60	HD-SDI	2'b01	4'b0100	1'b0
	1080i60x2	HD-SDI dual link	2'b01	4'b0100	1'b0
		3Gb	2'b10	4'b0100	1'b0
Example 2: Generate 1080p video format	1080p30	HD-SDI	2'b01	4'b1100	1'b0
	1080p30x2	HD-SDI dual link	2'b01	4'b1100	1'b0
		HD-SDI dual link	2'b01	4'b1100	1'b1
	1080p60	3Ga	2'b11	4'b1100	1'b0
		3Gb	2'b10	4'b1100	1'b1

Related Information

[Video Pattern Generator Signals](#) on page 8-8

Transceiver Reconfiguration Controller (Arria V, Cyclone V, and Stratix V Devices)

The transceiver reconfiguration controller reconfigures the transceivers.

For Arria V, Cyclone V, and Stratix V design examples, the transceiver reconfiguration controller allows you to change the device transceiver settings at any time. Any portion of the transceiver can be selectively reconfigured. Each portion of the reconfiguration requires a read-modify-write operation (read first, then

write), in such a way by modifying only the appropriate bits in a register and not changing other bits. Prior to this operation, you must define the logical channel number and the streamer module mode.

You can perform a transceiver dynamic reconfiguration in these two modes:

- streamer module mode 1 (manual mode)—execute a series of Avalon[®] Memory-Mapped (Avalon-MM) write operation to change the transceiver settings. In this mode, you can execute a write operation directly from the reconfiguration management/router interface to the device transceiver registers.
- streamer module mode 0—use the **.mif** files to change the transceiver settings.

For read operation, after defining the logical channel number and the streamer module mode, the following sequence of events occur:

1. Define the transceiver register offset in the offset register.
2. Read the data register. Toggle the read process by setting bit 1 of the control and status register (CSR) to logic 1.
3. Once the busy bit in the CSR is cleared to logic 0, it indicates that the read operation is complete and the required data should be available for reading.

For write operation, after setting the logical channel number and the streamer module mode, the following sequence of events occur:

1. Define the transceiver register offset (in which the data will be written to) in the offset register.
2. Write the data to the data register. Toggle the write process by setting bit 0 of the CSR to logic 1.
3. When the busy bit in the CSR is cleared to logic 0, it indicates that the transceiver register offset modification is successful.

For more information about the transceiver reconfiguration controller streamer module, refer to the Transceiver Reconfiguration Controller IP Core Overview chapter of the *Altera Transceiver PHY IP Core User Guide*.

Related Information

- [Transceiver Reconfiguration Controller Signals](#) on page 8-10
- [Modifying the Transceiver Reconfiguration Controller](#) on page 7-2
- [Altera Transceiver PHY IP Core User Guide](#)
More information about the transceiver reconfiguration controller streamer module.

Reconfiguration Management

The reconfiguration management block (`sdi_ii_ed_reconfig_mgmt.v` and `sdi_ii_reconfig_logic.v`) contains the reconfiguration user logic (a finite state machine) to determine the bits that needs to be modified, and selects the correct data to be written to the appropriate transceiver register through streamer module mode 1. It also provides handshaking between the SDI receiver and the transceiver reconfiguration controller. In this design, each reconfiguration block must interface with only one transceiver reconfiguration controller.

During the reconfiguration process, the logic first reads the data from the transceiver register that needs to be reconfigured and stores the data temporarily in a local register. Then, the logic overwrites only the appropriate bits of the data with predefined values and write the modified data to the transceiver register. Since only one transceiver register can be accessed at a time, the whole process repeats when reconfiguring other registers.

For multiple SDI channels reconfiguration, the logical channel number needs to be set appropriately for each channel and reconfiguration interface. For example, in the design example and simulation testbench figure, there are one SDI duplex, one SDI RX, and one SDI TX block. The number of reconfiguration interface for SDI duplex is 2 (one for channel and one for TX PLL), for SDI RX is 1 (for channel), for SDI TX is 2 (one for channel and one for TX PLL). The total number of reconfiguration interface required in the transceiver reconfiguration controller is 5.

The table below lists the channel and transceiver reconfiguration controller interface numbers.

The logical channel number for the receiver in SDI duplex is 0 and the logical channel number for SDI RX is 2. The generated example design entity demonstrates this interface connection.

Table 8-2: Channel Numbers Setting for Multiple SDI Channels Reconfiguration

SDI Block	SDI Channel Number	Transceiver Reconfiguration Controller Interface Number
SDI Duplex	0	0 and 1
SDI RX	1	2
SDI TX	1	3 and 4

For more information about the logical channel number, refer to the Transceiver Reconfiguration Controller IP Core Overview chapter of the *Altera Transceiver PHY IP Core User Guide*.

Related Information

- [Reconfiguration Management Parameters](#) on page 8-10
Tables below list the parameters for reconfiguration management.
- [Modifying the Reconfiguration Management](#) on page 7-2
- [Altera Transceiver PHY IP Core User Guide](#)
For more information about the logical channel number.

Reconfiguration Router

The reconfiguration router (`sdi_ii_ed_reconfig_router.v`) connects multiple SDI instances to the reconfiguration management and transceiver reconfiguration controller blocks. The reconfiguration router receives all the interface signals between the transceiver reconfiguration controller and reconfiguration management, as well as SDI instances, and transmits the signals to their respective destinations.

The reconfiguration router converts reconfiguration related interface signals of multiple SDI instances and user interface to a single-wide data bus for the reconfiguration management and transceiver reconfiguration controller blocks. You can bypass this component if you want to implement designs that expands to more channels.

Related Information

- [Reconfiguration Router Signals](#) on page 8-11
Table below lists the signals for the reconfiguration router.
- [Modifying the Reconfiguration Router](#) on page 7-4

Avalon-MM Translators

The Avalon-MM Master Translator and Avalon-MM Slave Translator are Avalon-MM interface blocks that access the Transceiver Reconfiguration Controller registers. The translators are not SDI-specific and are automatically instantiated when the core interfaces with an Avalon-MM master or slave component.

If you want to bypass the Avalon MM translator in your design, connect `reconfig_mgmt_address[8:2]` from the reconfiguration management block to `reconfig_mgmt_address` from the Transceiver Reconfiguration Controller.

You can connect the other signals from the reconfiguration management block directly to the Transceiver Reconfiguration Controller.

- `reconfig_mgmt_waitrequest`
- `reconfig_mgmt_read`
- `reconfig_mgmt_readdata`
- `reconfig_mgmt_write`
- `reconfig_mgmt_writedata`

For more information about the Avalon-MM Translator functions, refer to the Qsys Interconnect chapter in volume 1 of the *Quartus II Handbook*.

Related Information

- [Qsys Interconnect](#)
More information about the Avalon-MM Translator functions.
- [Avalon Interface Specifications](#)

Design Reference

This section includes detailed description about the SDI II IP core design examples.

Video Pattern Generator Signals

Table 8-3: Video Pattern Generator Top Level Signals

Table below lists the input signals for the video pattern generator. The listed signals are exported at the top level of the design example. Other signals—that are not exported—connect within the design example entity.

Signal	Width	Direction	Description
<code>pattgen_tx_std</code>	2	Input	Transmit video standard. <ul style="list-style-type: none"> • 00: SD-SDI • 01: HD-SDI or HD-SDI dual link • 10: 3G-SDI level B • 11: 3G-SDI level A

Signal	Width	Direction	Description
pattgen_tx_format	4	Input	Transmit video format. <ul style="list-style-type: none"> • 0000: SMPTE259M 525i • 0001: SMPTE259M 625i • 0100: SMPTE274M 1080i60 • 0101: SMPTE274M 1080i50 • 0110: SMPTE274M 1080p24 • 0111: SMPTE296M 720p60 • 1000: SMPTE296M 720p50 • 1001: SMPTE296M 720p30 • 1010: SMPTE296M 720p25 • 1011: SMPTE296M 720p24 • 1100: SMPTE274M 1080p30 • 1101: SMPTE274M 1080p25 • 1110: SMPTE274M 1080sF24 • Others: Reserved for future use
pattgen_dl_mapping	1	Input	Dual link mapping. Set to 1 for HD-SDI dual link and 3Gb transmit video standard only.
pattgen_ntsc_paln	1	Input	Transmit rate. <ul style="list-style-type: none"> • 0: PAL (1) rate. For example, 1080p30 • 1: NTSC (1/1.001) rate. For example, 1080p29.97. This input ignores all SD video formats (525i, 625i) and certain HD video formats that do not support NTSC rate (1080i50, 720p50, 720p25, 1080p25).
pattgen_bar_100_75n	1	Input	Generate color bars. <ul style="list-style-type: none"> • 0: 75% color bars • 1: 100% color bars
pattgen_patho	1	Input	Set to 1 to generate pathological pattern.
pattgen_blank	1	Input	Set to 1 to generate black signal.
pattgen_no_color	1	Input	Set to 1 to generate bars with no color.
pattgen_sgmt_frame	1	Input	Set to 1 to generate segmented frame picture for tx_format: <ul style="list-style-type: none"> • 0100: SMPTE274M 1080sF30 • 0101: SMPTE274M 1080sF25

Related Information

Video Pattern Generator on page 8-5

Transceiver Reconfiguration Controller Signals

Table 8-4: Transceiver Reconfiguration Controller Signals for Arria V, Cyclone V, and Stratix V Devices

Table below lists the input signals for the transceiver reconfiguration controller. The listed signals are exported at the top level of the design example. Other signals—that are not exported—connects within the design example entity.

Signal	Width	Direction	Description
reconfig_clk	1	Input	Clock signal for the transceiver reconfiguration controller and reconfiguration management/router. Refer to the transceiver reconfiguration controller section in the Altera Transceiver PHY IP User Guide for information about the frequency range.
reconfig_rst	1	Input	Reset signal for the transceiver reconfiguration controller and reconfiguration management/router. This signal is active high and level sensitive.

Related Information

- [Transceiver Reconfiguration Controller \(Arria V, Cyclone V, and Stratix V Devices\)](#) on page 8-5
The transceiver reconfiguration controller reconfigures the transceivers.
- [Modifying the Transceiver Reconfiguration Controller](#) on page 7-2
- [Altera Transceiver PHY IP Core User Guide](#)
More information about the transceiver reconfiguration controller frequency range.

Reconfiguration Management Parameters

Tables below list the parameters for reconfiguration management.

Table 8-5: Reconfiguration Management Parameters for Arria V, Cyclone V, and Stratix V Devices

Parameter	Value	Description
NUM_CHS	1 (minimum)	Number of channels required to do reconfiguration.
FAMILY	<ul style="list-style-type: none"> • Arria V • Arria V GZ • Cyclone V • Stratix V 	Supported device family.

Parameter	Value	Description
DIRECTION	<ul style="list-style-type: none"> tx rx du 	<p>Direction of the core selected in the parameter editor. This parameter affects the logical channel number assigned in the generated example design.</p> <p>If you are making any changes to the design, please ignore this parameter and assign the logical channel number correctly.</p> <p>Refer to <i>Expanding to Multiple Channels</i> section to know how to assign the logical channel number.</p>
VIDEO_STANDARD	<ul style="list-style-type: none"> tr dl 	<p>Current video standard.</p> <p>Specify <i>dl</i> for HD dual-link or <i>tr</i> for other standards.</p>
XCVR_TX_PLL_SEL	<ul style="list-style-type: none"> 1 2 	<p>The selected method to perform TX PLL reconfiguration for dynamic clock switching. Specify 1 to switch TX PLLs or 2 to switch TX PLL reference clocks.</p> <p>The specified value must match the parameter value you select when you instantiate the core.</p> <p>Refer to <i>Dynamic TX Clock Switching</i> section to know more about clock switching.</p>

Related Information

- [Reconfiguration Management](#) on page 8-6
- [Modifying the Reconfiguration Management](#) on page 7-2

Reconfiguration Router Signals

Table below lists the signals for the reconfiguration router.

Table 8-6: Reconfiguration Router Top Level Signals

The listed signals are exported at the top level of the design example. Other signals—that are not exported—connect within the design example entity.

Note: These signals are available only when you use the Dynamic TX clock switching feature.

Refer to [Dynamic TX Clock Switching for Arria V, Cyclone V, and Stratix V Devices](#) for usage requirements.

Signal	Width	Direction	Description
ch1_<direction>_tx_start_reconfig	1	Input	Dynamic reconfiguration request signal for TX PLL dynamic switching at transmitter or duplex instance at channel 1.

Signal	Width	Direction	Description
ch1_<direction>_tx_pll_sel	1	Input	TX PLL select signal for TX PLL dynamic switching at transmitter or duplex instance at channel 1. This signal is also connected to <code>xcvr_refclk_sel</code> signal of the SDI instance.
ch1_<direction>_tx_reconfig_done	1	Output	Dynamic reconfiguration acknowledge signal for TX PLL dynamic switching at transmitter or duplex instance at channel 1.

Related Information

- [Reconfiguration Router](#) on page 8-7
- [Modifying the Reconfiguration Router](#) on page 7-4

Simulating the SDI II IP Core Design

After design generation, the files located in the simulation testbench directory are available for you to simulate your design.

The SDI II IP core supports the following EDA simulators listed in the table below.

Table 8-7: Supported EDA Simulators

Simulator	Supported Platform	Supported Language
ModelSim-SE	Windows/Linux	VHDL and Verilog HDL
ModelSim-Altera	Windows/Linux	Verilog
VCS/VCS MX	Windows/Linux	Verilog
Aldec Riviera-PRO	Linux	Verilog

To simulate the design using the ModelSim-SE or ModelSim-Altera simulator, follow these steps:

1. Start the simulator.
2. On the File menu, click **Change Directory > Select** *<simulation folder>/<preferred HDL>/mentor*.
3. Run the provided `run_sim.tcl` script. This file compiles the design and runs the simulation automatically. It provides a pass/fail indication on completion.

To simulate the design using the VCS/VCS MX simulator (in Linux), follow these steps:

1. Start the VCS/VCS MX simulator.
2. On the File menu, click **Change Directory** > **Select** <simulation folder>/<preferred HDL>/synapsys.
3. Run the provided `run_vcs.sh` (in VCS) or `run_vcsmx.sh` (in VCSMX) script. This file compiles the design and runs the simulation automatically. It provides a pass/fail indication on completion.

To simulate the design using the Aldec Riviera-PRO simulator, follow these steps:

1. Start the Aldec Riviera-PRO simulator.
2. On the File menu, click **Change Directory** > **Select** <simulation folder>/<preferred HDL>/aldec.
3. Run the provided `run_riviera.tcl` script. This file compiles the design and runs the simulation automatically. It provides a pass/fail indication on completion.

Simulation Run Time

Table 8-8: Estimated Simulation Run Time

The table lists the default estimated run-time settings for each SDI II video standard.

Video Standard	Estimated Run Time (ms)
SD-SDI	3.26
HD-SDI	9.51
3G-SDI	7.62
HD-SDI Dual Link	6.15
Dual Rate (up to HD-SDI)	13.44
Triple Rate (up to 3G-SDI)	55.83
Multi Rate (up to 12G-SDI)	416.94

SDI II IP Core User Guide Archives



2016.12.20

UG-01125



Subscribe



Send Feedback

If an IP core version is not listed, the user guide for the previous IP core version applies.

IP Core Version	User Guide
16.0	SDI II IP Core User Guide
15.1	SDI II IP Core User Guide
15.0	SDI II IP Core User Guide
14.1	SDI II IP Core User Guide

© 2016 Intel Corporation. All rights reserved. Intel, the Intel logo, Altera, Arria, Cyclone, Enpirion, MAX, NIOS, Quartus and Stratix words and logos are trademarks of Intel Corporation in the US and/or other countries. Other marks and brands may be claimed as the property of others. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO
9001:2008
Registered

Revision History for SDI II IP Core User Guide



2016.12.20

UG-01125



Subscribe



Send Feedback

Date	Version	Changes
December 2016	2016.12.20	<ul style="list-style-type: none"> Added detailed description for <code>tx_datain</code> and <code>rx_dataout</code> signals about 6G-SDI and 12G-SDI interfaces. Added information about image mapping for 6G-SDI and 12G-SDI interfaces. Added information for <code>rx_dataout_valid</code> signal that the 1H4L 1H5L cadence for SD-SDI repeats indefinitely in an ideal case but in a typical scenario the cadence shift periodically (for instance, 1H4L 1H5L 1H5L 1H4L). Updated <code>rx_format</code> information to include that for 6G-SDI or 12G-SDI interfaces, each of the 20-bit interface reports its own detected format. Added information for <code>pll_powerdown_in</code> signal that sharing Tx PLLs for designs that also implement dynamic reconfiguration require <code>XCVR_TX_PLL_RECONFIG_GROUP</code> QSF assignment.
October 2016	2016.10.31	<ul style="list-style-type: none"> Restructured the chapters. Added information for the new Design Example parameters. Removed all Arria 10 design example related information. For more information about Arria 10 design examples, refer to the <i>SDI II IP Core Design Example User Guide</i>. Added clocking diagrams for Arria 10 devices and the V series devices—Arria V, Cyclone V, and Stratix V. Added guideline to overcome potential routability issue during Fitter stage.

© 2016 Intel Corporation. All rights reserved. Intel, the Intel logo, Altera, Arria, Cyclone, Enpirion, MAX, NIOS, Quartus and Stratix words and logos are trademarks of Intel Corporation in the US and/or other countries. Other marks and brands may be claimed as the property of others. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO
9001:2008
Registered



Date	Version	Changes
May 2016	2016.05.02	<ul style="list-style-type: none"> • Added new option, fPLL, for the Arria 10 TX PLL parameter and removed the ATX PLL option. • Added estimated run-time settings for the different SDI II video standards. • Added guideline for transceiver handling. The transceiver handling guidelines differ for Arria 10 devices and the V series devices—Arria V, Cyclone V, and Stratix V. • Added new transceiver signals: <ul style="list-style-type: none"> • rx_analogreset_ack • tx_analogreset_ack • rx_cal_busy • pll_powerdown • xcvr_rxclk • xcvr_rxclk_b • rst_tx_phy • Added a new receiver signals: rx_datain and rx_datain_valid. • Removed these signals: rx_pll_locked and rx_pll_locked_b. These signals are redundant and no longer required after the switch to Native PHY. • Updated the design example directory. • Added links to archived versions of the <i>SDI II IP Core User Guide</i>.
November 2015	2015.11.02	<ul style="list-style-type: none"> • Added information that the rx_format signal for each stream reports its own detected format for 6G-SDI and 12G-SDI interfaces. • Added information about 3 new interface signals for Arria V, Cyclone V, Stratix V devices: rx_trs_in, pll_powerdown_in, and pll_powerdown_out • Added reconfiguration management parameters for Arria 10 devices: VIDEO_STANDARD, ED_TXPLL_SWITCH, and XCVR_RCFG_IF_TYPE. • Added descriptions for the SDI presets available in the Arria 10 Transceiver Native PHY IP core.
May 2015	2015.05.04	<ul style="list-style-type: none"> • Changed the resource utilization table to include data for each SDI standard and updated the data for version 15.0. • Added new multi-rate data path block diagrams for transmitter and receiver. • Added new information about inserting sync bits. • Renamed the term <i>video payload ID (VPID)</i> to <i>payload ID</i> as per SMPTE specification. • Renamed <i>Level A</i> to <i>HD-SDI dual link</i> and <i>Level B</i> to <i>3G-SDI (level B)</i>.

Date	Version	Changes
		<ul style="list-style-type: none"> • Updated the following new parameter options: <ul style="list-style-type: none"> • Added new video standard Multi rate (up to 12G) for Arria 10 devices. • Added TX PLL reference clock switching option for Dynamic Tx clock switching parameter. • Added a note for the interface signals to indicate that multi-rate (up to 12G) mode requires 4 streams and the rest require one stream. • Added a new parameter for Reconfiguration Management: <code>XCVR_TX_PLL_SEL</code>. • Added information for multi standard support including 6G-SDI and 12G-SDI. • Added the multi standard (including 6G-SDI and 12G-SDI) information for the following signals: <ul style="list-style-type: none"> • <code>tx_enable_ln</code> • <code>tx_std</code> • <code>tx_datain</code> • <code>tx_datain_valid</code> • <code>tx_ln_b</code> • <code>tx_dataout</code> • <code>tx_dataout_valid</code> • <code>tx_vpid_byte(1-4)_b</code> • <code>rx_std</code> • <code>rx_dataout_valid</code> • <code>rx_format</code> • <code>rx_ln_b</code> • <code>rx_vpid_byte(1-4)_b</code> • <code>rx_vpid_checksum_error_b</code> • Added information that the following signals are not applicable for Arria 10 devices: <ul style="list-style-type: none"> • <code>rx_coreclk_hd</code> • <code>rx_clkin</code> • <code>rx_clkin_b</code> • <code>rx_rst_proto_in</code> • <code>rx_rst_proto_in_b</code>

Date	Version	Changes
January 2015	2015.01.23	<ul style="list-style-type: none"> • Updated the resource utilization table for version 14.1. • Changed the names of the following parameters for receiver options: <ul style="list-style-type: none"> • Convert Level A to Level B (SMPTE 372M) changed to Convert HD-SDI dual link to 3G-SDI (level B). • Convert Level B to Level A (SMPTE 372M) changed to Convert 3G-SDI (level B) to HD-SDI dual link. • Edited information about <code>rx_format</code> signal, which now reports video transport format instead of picture format. The signal reports 3G Level A RGB or YCbCr 4:4:4 format.
August 2014	2014.08.18	<ul style="list-style-type: none"> • Added support for Arria 10 devices. • Revised the resource utilization table with information about ALM needed and primary and secondary logic registers. • Added information related to Arria 10 devices. <ul style="list-style-type: none"> • Added new parameters for Example Design Options. • Added new transceiver information—for the Arria 10 devices, the SDI II IP core no longer provides the transceiver, and the TX PLL is no longer wrapped in the transceiver PHY. You must generate the transceiver and the TX PLL separately. • Added new transceiver signals: <code>rx_ready</code>, <code>gxb_ltr</code>, <code>gxb_ltd</code>, <code>rx_ready_b</code>, <code>gxb_ltr_b</code>, <code>gxb_ltd_b</code>, and <code>trig_rst_ctrl</code>. • Added information for the newly added Arria 10 design example. <ul style="list-style-type: none"> • Added design example entity and simulation testbench diagram. • Added connecting input signals: <code>rx_manual</code> and <code>rx_is_lockedtoata</code>. • Added information about transceiver reconfiguration controller—for Arria 10 designs, the reconfiguration interface is integrated into the Arria 10 Native PHY instance and TX PLL. • Added transceiver reconfiguration controller signals. • Added information about IP catalog and removed information about MegaWizard Plug-In Manager.
July 2013	2013.06.28	<ul style="list-style-type: none"> • Added a section for each new feature: <ul style="list-style-type: none"> • Tx PLL Dynamic Switching • SMPTE RP168 Switching • SD Optional 20-bit Interface for Dual/Triple Rate • Added information about a new submodule, Convert SD Bits. • Added information about a new parameter, SD Interface Bit Width. • Added more information about the design example components—Reconfiguration Management, Reconfiguration Router, Avalon-MM Translators.

Date	Version	Changes
		<ul style="list-style-type: none">• Added more information about the design example operation:<ul style="list-style-type: none">• Transceiver Dynamic Reconfiguration• Expanding to Multiple Channels• Updated the protocol and transceiver signals table.• Updated the resource utilization table.
November 2012	2012.11.15	Initial release.