



# Intel® FPGA SDI II Design Example User Guide for Intel® Cyclone 10 GX Devices

Updated for Intel® Quartus® Prime Design Suite: **17.1.1**



**Subscribe**

**Send Feedback**

**UG-20124 | 2017.12.25**

Latest document on the web: [PDF](#) | [HTML](#)



## Contents

---

<b>1 Intel® FPGA SDI II Design Example Quick Start Guide for Intel® Cyclone® 10 GX Devices.....</b>	<b>3</b>
1.1 Directory Structure.....	3
1.2 Hardware and Software Requirements.....	6
1.3 Generating the Design.....	7
1.4 Simulating the Design.....	7
1.5 Compiling and Testing the Design .....	8
1.5.1 Connection and Settings Guidelines.....	10
1.5.2 Design Considerations for Serial Loopback Design.....	12
1.6 Intel FPGA SDI II Design Example Parameters .....	12
<b>2 Intel FPGA SDI II Design Example Detailed Description.....</b>	<b>15</b>
2.1 Parallel Loopback Design Examples.....	16
2.2 Serial Loopback Design Examples.....	19
2.3 Design Components.....	23
2.4 Clocking Scheme Signals.....	26
2.5 Interface Signals.....	28
2.6 Video Pattern Generator Parameters.....	39
2.7 Hardware Setup.....	39
2.8 Simulation Testbench.....	40
<b>A Revision History for Intel FPGA SDI II Design Example User Guide for Intel Cyclone 10 GX Devices.....</b>	<b>44</b>

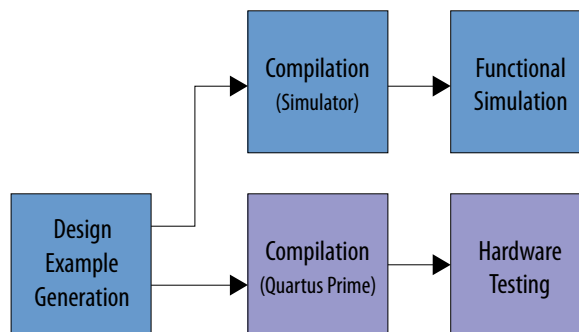


## 1 Intel® FPGA SDI II Design Example Quick Start Guide for Intel® Cyclone® 10 GX Devices

The Intel FPGA Serial Digital Interface (SDI) II design examples for Intel® Cyclone® 10 GX devices feature a simulation testbench and a hardware design that supports compilation and hardware testing.

When you generate a design example, the parameter editor automatically creates the files necessary to simulate, compile, and test the design in hardware.

**Figure 1. Development Steps**



### Related Links

[Intel FPGA SDI II IP Core User Guide](#)

### 1.1 Directory Structure

The directories contain the generated files for the design examples.



Figure 2. Directory Structure for the Design Examples

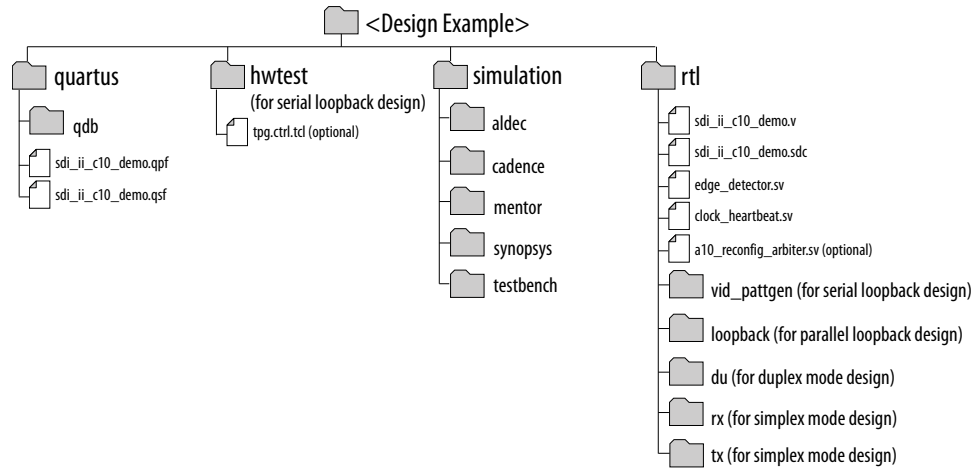


Table 1. Other Generated Files in RTL Folder

Folders	Files
vid_pattgen	/sdi_ii_colorbar_gen.v
	/sdi_ii_ed_vid_pattgen.v
	/sdi_ii_makeframe.v
	/sdi_ii_patho_gen.v
	/jtag.sdc
	/pattgen_ctrl.qsys
	<qsys generated folder>
loopback	/loopback_top.v
	/fifo/sdi_ii_ed_loopback.sdc
	/fifo/sdi_ii_ed_loopback.v
	/pfd/clock_crossing.v (optional)
	/pfd/pfd.sdc (optional)
	/pfd/pfd.v (optional)
	/reclock/sdi_reclock.v (optional)
	/reclock/pid_controller.v (optional)
du	/du_top.v
	/sdi_ii_rx_rcfg_c10.sv (optional)
	/rcfg_sdi_cdr.sv (optional)
	/rcfg_pll_sw.sv (optional)
	/rcfg_refclk_sw.sv (optional)
	/sdi_ii_tx_rcfg_c10.sv (optional)

*continued...*



Folders	Files
	/sdi_du_sys.qsys
	/sdi_rx_phy.ip
	/tx_pll.ip
	/tx_pll_alt.ip (optional)
	<qsys generated folder>
rx	/rx_top.v
	/sdi_ii_rx_rcfg_c10.sv (optional)
	/rcfg_sdi_cdr.sv (optional)
	/sdi_rx_sys.qsys
	<qsys generated folder>
tx	/tx_top.v
	/rcfg_pll_sw.sv (optional)
	/rcfg_refclk_sw.sv (optional)
	/sdi_ii_tx_rcfg_c10.sv (optional)
	/sdi_tx_sys.qsys
	/tx_pll.ip
	/tx_pll_alt.ip(optional)
	<qsys generated folder>

**Table 2. Other Generated Files in Simulation Folder**

Folders	Files
aldec	/aldec.do
	/rivierapro_setup.tcl
cadence	/cds.lib
	/hdl.var
	/ncsim.sh
	/ncsim_setup.sh
	<cds_libs folder>
mentor	/mentor.do
	/msim_setup.tcl
synopsys	/vcs/filelist.f
	/vcs/vcs_setup.sh
	/vcs/vcs_sim.sh
	/vcsmx/synopsys_sim_setup
	/vcsmx/vcsmx_setup.sh

**continued...**



Folders	Files
	/vcsmx/vcsmx_sim.sh
testbench	tb_top.v
	rx_checker/sdi_ii_tb_rx_checker.v
	rx_checker/tb_data_compare.v
	rx_checker/tb_dual_link_sync.v
	rx_checker/tb_fifo_line_test.v
	rx_checker/tb_frame_locked_test.sv
	rx_checker/tb_rxsample_test.v
	rx_checker/tb_trs_locked_test.sv
	rx_checker/tb_txpll_test.sv
	rx_checker/tb_vpid_check.v
	tb_control/sdi_ii_tb_control.v
	tb_control/tb_clk_rst.v
	tb_control/tb_data_delay.v
	tb_control/tb_serial_delay.sv
	tb_control/tb_tasks.v
	tb_checker/sdi_ii_tb_tx_checker.v
	tb_checker/tb_serial_check_counter.v
	tb_checker/tb_serial_descrambler.v
	tb_checker/tb_tx_clkout_check.v
	vid_pattgen/sdi_ii_colorbar_gen.v
	vid_pattgen/sdi_ii_ed_vid_pattgen.v
	vid_pattgen/sdi_ii_makeframe.v
	vid_pattgen/sdi_ii_patho_gen.v

## 1.2 Hardware and Software Requirements

Intel uses the following hardware and software to test the design examples.

### Hardware

- Intel Cyclone 10 GX FPGA Development Kit
- SDI Signal Generator
- SDI Signal Analyzer
- Bayonet Neill-Concelman (BNC) to BNC cables
- VIDIO™ FMC Development Module VIDIO-12G-A (Nextera 12G SDI FMC daughter card)



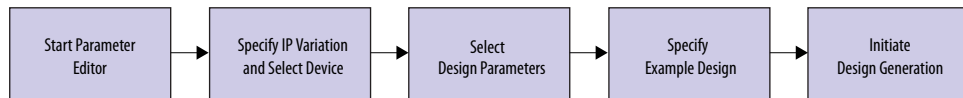
## Software

- Intel Quartus Prime Pro Edition (for hardware testing)
- ModelSim\* - Intel FPGA Edition, ModelSim - Intel FPGA Starter Edition, NCSim, Riviera-Pro, or VCS (Verilog only)/VCS-MX simulator

## 1.3 Generating the Design

Configure the Intel FPGA SDI II parameter editor in the Intel Quartus Prime Pro Edition software to generate the design examples.

**Figure 3. Generating the Design Flow**

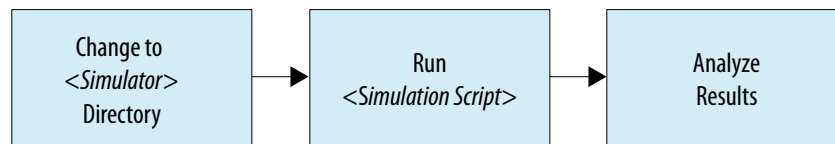


1. Create a project targeting the Intel Cyclone 10 GX device family and select the desired device.
2. In the IP Catalog, locate and double-click **Intel FPGA SDI II**. The **New IP Variant** or **New IP Variation** window appears.
3. Specify a top-level name for your custom IP variation. The parameter editor saves the IP variation settings in a file named `<your_ip>.ip`.
4. Click **OK**. The parameter editor appears.
5. On the **IP** tab, select your desired IP settings. The generated design example is based on your settings.
6. On the **Design Example** tab, select **Simulation** to generate the testbench, and select **Synthesis** to generate the hardware design example.  
You must select at least one of these options to generate the design example files.
7. For **Generate File Format**, select **Verilog** or **VHDL**.
8. For **Target Development Kit**, select the relevant FPGA development kit. You may change the target device using the **Change Target Device** parameter if your board revision does not match the grade of the default targeted device.
9. Click **Generate Example Design**.

## 1.4 Simulating the Design

The Intel FPGA SDI II design example testbench simulates one channel serial loopback design with TX instance connected to an internal video pattern generator. The serial output from the TX instance connects to the RX instance in the testbench. The testbench also includes checkers and control mechanisms.

**Figure 4. Design Simulation Flow**





1. Navigate to the simulation folder of your choice.
2. Run the simulation script for the supported simulator of your choice. The script compiles and runs the testbench in the simulator.
3. Analyze the results.

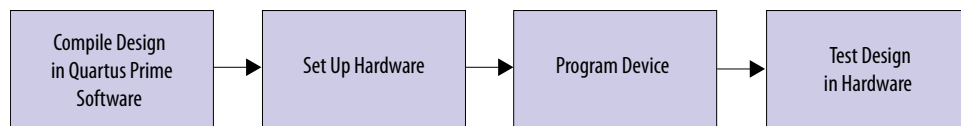
**Table 3. Steps to Run Simulation**

Simulator	Working Directory	Instructions
Riviera-Pro	/simulation/aldec	In the GUI, type: <pre>do aldec.do</pre>
NCSim	/simulation/cadence	In the command line, type: <pre>source ncsim.sh</pre>
ModelSim	/simulation/mentor	In the GUI, type: <pre>do mentor.do</pre>
VCS	/simulation/synopsys/vcs	In the command line, type: <pre>source vcs_sim.sh</pre>
VCS-MX	/simulation/synopsys/ vcsmx	In the command line, type: <pre>source vcsmx_sim.sh</pre>

A successful simulation ends with the following message:

```
#### TRANSMIT TEST COMPLETED SUCCESSFULLY! ####  
#  
#### Channel 1: RECEIVE TEST COMPLETED SUCCESSFULLY! ####
```

## 1.5 Compiling and Testing the Design



To compile and run a demonstration test on the hardware design example, follow these steps:

1. Ensure that the hardware design example generation is complete.
2. Open `quartus/sdi_ii_c10_demo.qpf`.
3. Click **Processing** ► **Start Compilation**.
4. Open the **Clock Controller** parameter editor, and set the clock frequency.





SDI II Design Examples		Si570 (Y2) Tab	Si5332 (U64) Tab
HD/3G-SDI single-rate and triple-rate	Parallel loopback without external VCXO designs	Set <b>Target Frequency</b> to 148.5 MHz.	Set <b>OUT1</b> frequency to 100 MHz.
	Serial loopback designs	Set <b>Target Frequency</b> to 148.3516 MHz if <b>Dynamic Tx clock switching</b> is enabled.	Set <b>OUT1</b> frequency to 148.5 MHz
Multi-rate	Parallel loopback with external VCXO designs	Set <b>Target Frequency</b> to 297 MHz.	Set <b>OUT1</b> frequency to 148.5 MHz.
	Serial loopback designs	Set <b>Target Frequency</b> to 296.7033 MHz if <b>Dynamic Tx clock switching</b> is enabled.	Set <b>OUT1</b> frequency to 148.5 MHz

Figure 5. Si570 (Y2) Tab in the Clock Controller GUI

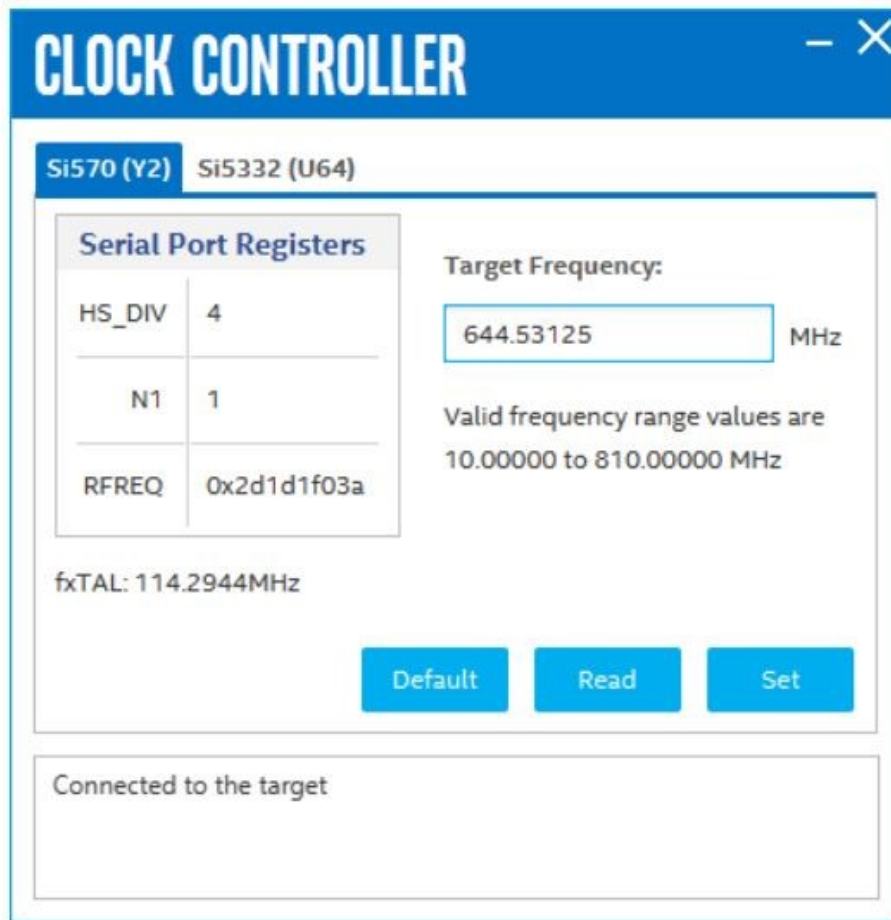
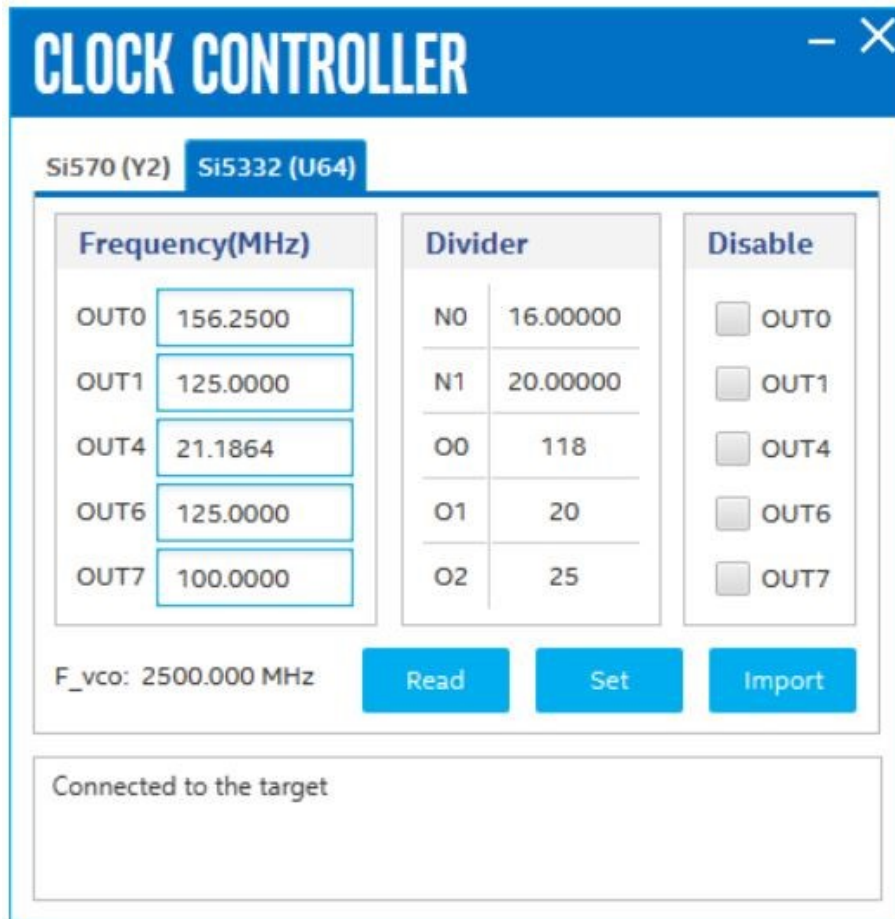


Figure 6. Si5332 (U64) Tab in the Clock Controller GUI



- After successful compilation, the Intel Quartus Prime Pro Edition software generates a .sof file in your specified directory.
- Configure the selected Intel Cyclone 10 GX device on the development board using the generated .sof file (**Tools > Programmer**).
- For serial loopback designs, open the System Console to control the internal video pattern generator. Click **Tools > System Debugging Tools > System Console**.  
*Note:* Close the **Clock Controller** GUI and the Programmer window before you open the System Console.
- After the initialization, type `source ../hwtest/tpg_ctrl.tcl` in the System Console to open the pattern generator control user interface. Select your desired video format.

### 1.5.1 Connection and Settings Guidelines

Before programing with the .sof file, ensure that the connections and settings are correct.



- For parallel loopback design, the on-board BNC RX connector (J1/12G In) connects to an external video source and the on-board BNC TX connector (J2/12G Out) connects to a video analyzer.
- For serial loopback design, the on-board BNC TX connector (J2/12G Out) connects to an on-board BNC RX connector (J1/12G In) or a video analyzer.
- Ensure all switches on the development board are in default position.
- The SDI video analyzer displays the video generated from the source.

*Note:* For parallel loopback designs, change the jumper (J8) position before switching between fractional frame rate and integer frame rate video formats. Press the push button (PB1) to trigger a device (LMK03328) power cycling through the PDN pin every time you change the jumper (J8) position.

**Table 4. SW2 DIP Switch Default Settings (Bottom of the Board)**

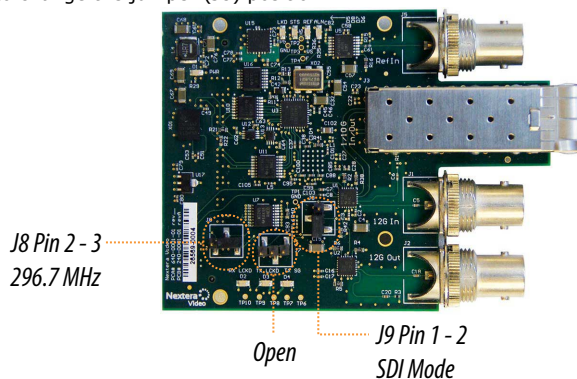
Switch	Board Label	Description
1	CLKBUF_SEL	Set the input source of Si53307(U7): <ul style="list-style-type: none"> <li>• 0: Use CLKIN0 (default)</li> <li>• 1: Use CLKIN1</li> </ul>
2	Si570_OE	Enable the output of Si570 (Y2): <ul style="list-style-type: none"> <li>• 0: Output disabled</li> <li>• 1: Output enabled (default)</li> </ul>

**Table 5. SW3 DIP Switch Default Settings (Bottom of the Board)**

Switch	Board Label	Description
1	Si5340_INSEL0	<ul style="list-style-type: none"> <li>• Si5340_INSEL1, Si5340_INSEL0 = [0,0]</li> <li>• Use crystal as PLL input for Si5332 (default)</li> </ul>
2	Si5340_INSEL1	<ul style="list-style-type: none"> <li>• Si5340_INSEL1, Si5340_INSEL0 = [1,0]</li> <li>• Use external SMB as PLL input for Si5332</li> </ul>

**Figure 7. Jumper Settings on Nextera 12G-SDI FMC Daughter Card**

Refer to these settings to change the jumper (J8) position.





**Table 6. Jumper Settings**

Jumper Block	Description
J7	Programming header
J8	To switch the generated clock frequency for the TX channel: <ul style="list-style-type: none"> <li>Pin 1–2 = 297 MHz</li> <li>Pin 2–3 = 297/1.001 MHz</li> </ul>
J9	To select SDI or IP mode: <ul style="list-style-type: none"> <li>Pin 1–2 = SDI mode</li> <li>Pin 2–3 = IP mode</li> </ul>

**Related Links**

[Intel Cyclone 10 GX FPGA Development Kit](#)

**1.5.2 Design Considerations for Serial Loopback Design**

You need to consider the following issues when instantiating the serial loopback design example:

- Serial loopback design is mainly for image and TX clock switching demonstrations only. To get a more accurate jitter performance with the daughter card components, use the parallel loopback design and connect it to a clean video source.
- To allow segmented frame video format (1080sF30, 1080sF25) and interlaced video format (1080i60, 1080i50) to be correctly differentiated in the external analyzer, Payload ID must be inserted in the serial loopback design.
- The Omnitek Ultra 4K Analyzer (software version 2.1) may not detect 12G-SDI 2160p59.94 in the serial loopback design. If you encounter such problem, upgrade the Omnitek Ultra 4K analyzer to a later version.

**1.6 Intel FPGA SDI II Design Example Parameters**

**Table 7. Intel FPGA SDI II Design Example Parameters for Intel Cyclone 10 GX Devices**

Parameter	Value	Description
<b>Available Design Example</b>		
Select Design	Parallel loopback with external VCXO Parallel loopback without external VCXO Serial loopback	Select the design example to be generated. <ul style="list-style-type: none"> <li>Parallel loopback with external VCXO: Parallel loopback design with an external VCXO to synchronize the clock between RX and TX.</li> <li>Parallel loopback without external VCXO: Parallel loopback design without an external VCXO.</li> </ul> <i>Note:</i> This option is not available for multi-rate designs. <ul style="list-style-type: none"> <li>Serial loopback: A serial loopback design to enables a simple demonstration when you do not have a video source available and to highlight the <b>Dynamic Tx clock switching</b> feature. The IP core generates an internal video pattern generator along with the TX to be transmitted to RX.</li> </ul>
<b>Design Example Options</b>		
Tx PLL type	CMU fPLL	Select the transceiver PLL type.
<i>continued...</i>		



Design Example Options		
		<ul style="list-style-type: none"> <li>CMU PLL only supports data rates up to 3G-SDI.</li> <li>fPLL supports all data rates up to 12G-SDI.</li> </ul> <p><i>Note:</i> This option is available only for multi-rate designs or when you generate a parallel loopback design without an external VCXO.</p>
Dynamic Tx clock switching	Off Tx PLL switching Tx PLL reference clock switching	<ul style="list-style-type: none"> <li>Off: Disable dynamic switching.</li> <li>Tx PLL switching: Instantiates two PLLs, each with its own reference input clock.</li> <li>Tx PLL reference clock switching: Instantiates one PLL with two reference input clocks.</li> </ul> <p>Turn on this option to allow dynamic switching between 1 and 1/1.001 data rates. This option is only available when you select <b>Serial loopback</b>.</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 Intel Quartus Prime compilation and hardware demonstration.

Generated HDL Format		
Generate File Format	Verilog VHDL	Select your preferred HDL format for the generated design example fileset.  <i>Note:</i> 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.

Target Development Kit		
Select Board	No Development Kit Intel Cyclone 10 GX FPGA Development Kit Custom Development Kit	Select the board for the targeted design example. <ul style="list-style-type: none"> <li>No Development Kit: This option excludes all hardware aspects for the design example. The IP core sets all pin assignments to virtual pins.</li> <li>Intel Cyclone 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 <b>Change Target Device</b> parameter if your board revision has a different device variant. The IP core sets all pin assignments according to the development kit.</li> </ul> <p><i>Note:</i> This option is not available if you select <b>Bidirectional</b> mode. The development kit and Nextera daughter card pins are only compatible with simplex mode. For more details, refer to <a href="#">Table 8</a> on page 14.</p> <ul style="list-style-type: none"> <li>Custom Development Kit: This option allows the design example to be tested on a third party development kit with an Intel FPGA. You may need to set the pin assignments on your own.</li> </ul>

Target Device		
Change Target Device	On Off	Turn on this option and select the preferred device variant for the development kit.



**Table 8. Supported Hardware Demonstration for Intel Cyclone 10 GX Devices**

Video Standards	Supported Designs
HD/3G Single Rate and Triple Rate	<ul style="list-style-type: none"><li>• Parallel loopback without external VCXO (Simplex mode only)</li><li>• Serial loopback with the <b>Dynamic Tx clock switching</b> parameter enabled or disabled (Simplex mode only)</li></ul>
Multi Rate	<ul style="list-style-type: none"><li>• Parallel loopback with external VCXO (Simplex mode only)</li><li>• Serial loopback with the <b>Dynamic Tx clock switching</b> parameter enabled or disabled (Simplex mode only)</li></ul>



## 2 Intel FPGA SDI II Design Example Detailed Description

The Intel FPGA SDI II IP core includes these design examples for Intel Cyclone 10 GX devices.

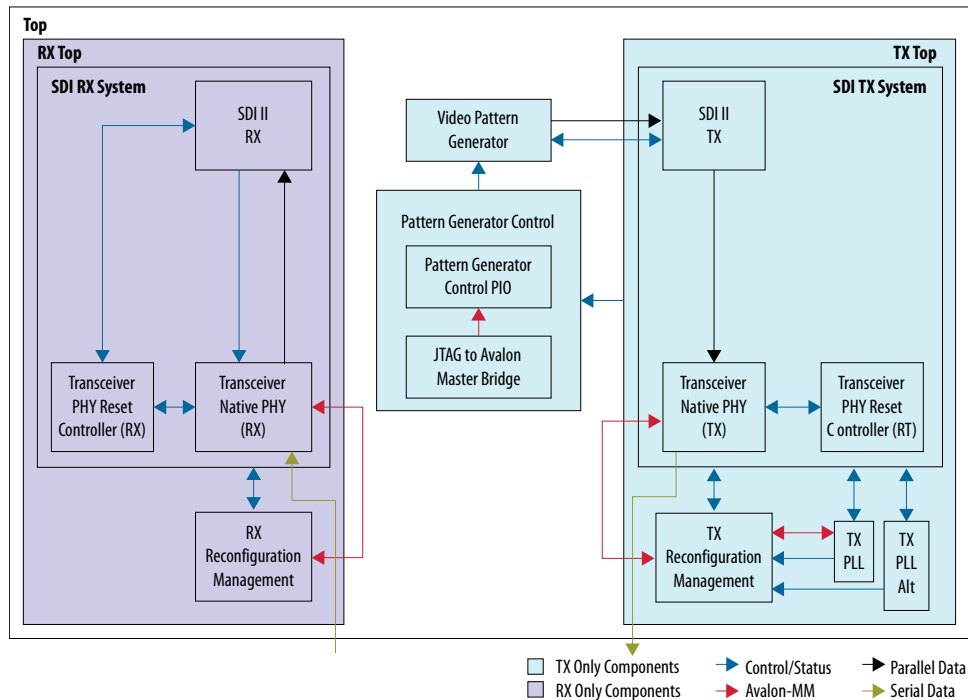
- Parallel loopback with external VCXO
- Parallel loopback without external VCXO
- Serial loopback

### Features

- All designs use LED status for early debugging stage.
- The simplex serial loopback designs include RX and TX options. To use RX or TX only components, remove the irrelevant blocks from the designs.

User Requirement	Preserve	Remove
RX Only	RX Top	– TX Top
TX Only	TX Top	– RX Top

**Figure 8. Components Required for Intel Cyclone 10 GX TX or RX Only Design**



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 or its subsidiaries in the U.S. and/or other countries. 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.

\*Other names and brands may be claimed as the property of others.

**ISO  
9001:2008  
Registered**

## 2.1 Parallel Loopback Design Examples

The parallel loopback design examples demonstrate simplex and duplex channel modes with and without external VCXO.

**Note:** For parallel loopback duplex designs, do not share the TX PLL reference clock with the RX transceiver reference clock. The design logic tunes the TX PLL clock to match the RX recovered clock frequency.

**Figure 9. Parallel Loopback with Simplex Mode Block Diagram**

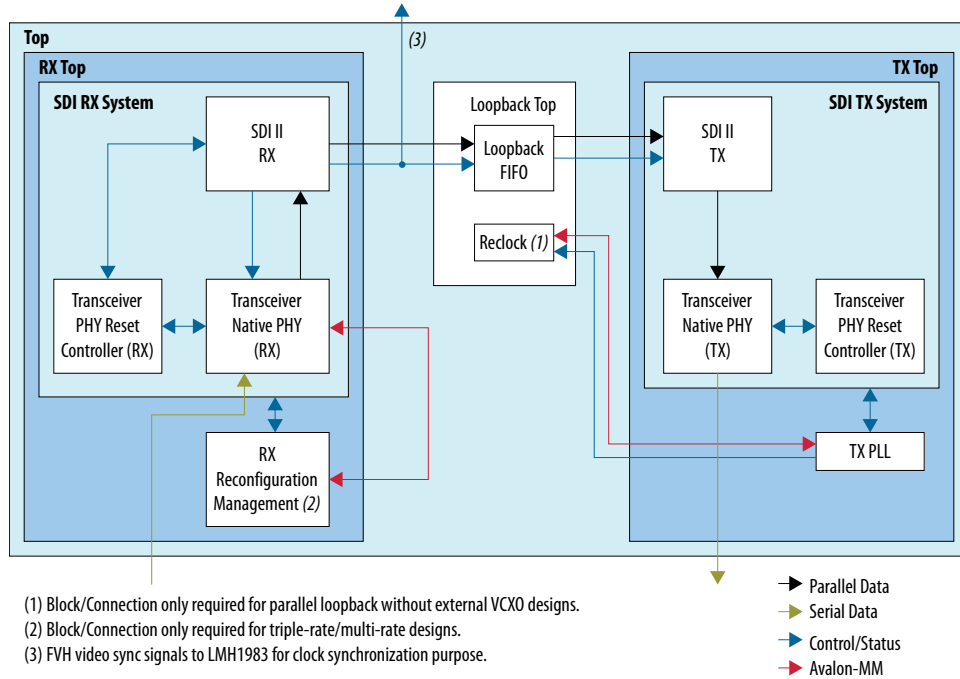






Figure 10. Parallel Loopback with Simplex Mode Clocking Scheme

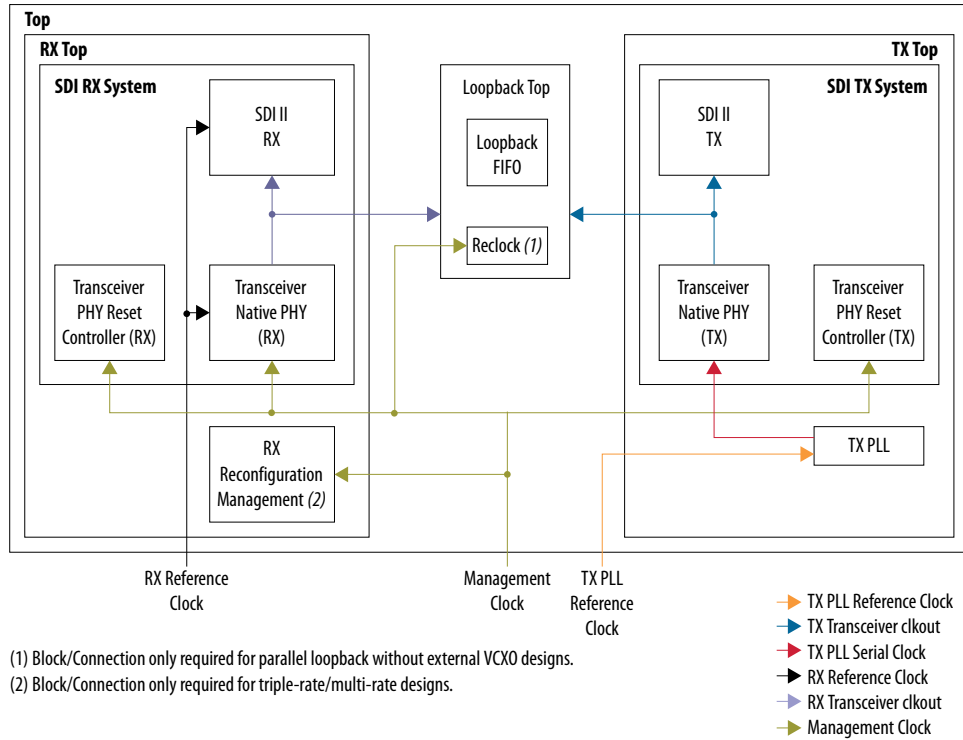


Figure 11. Parallel Loopback with Duplex Mode Block Diagram

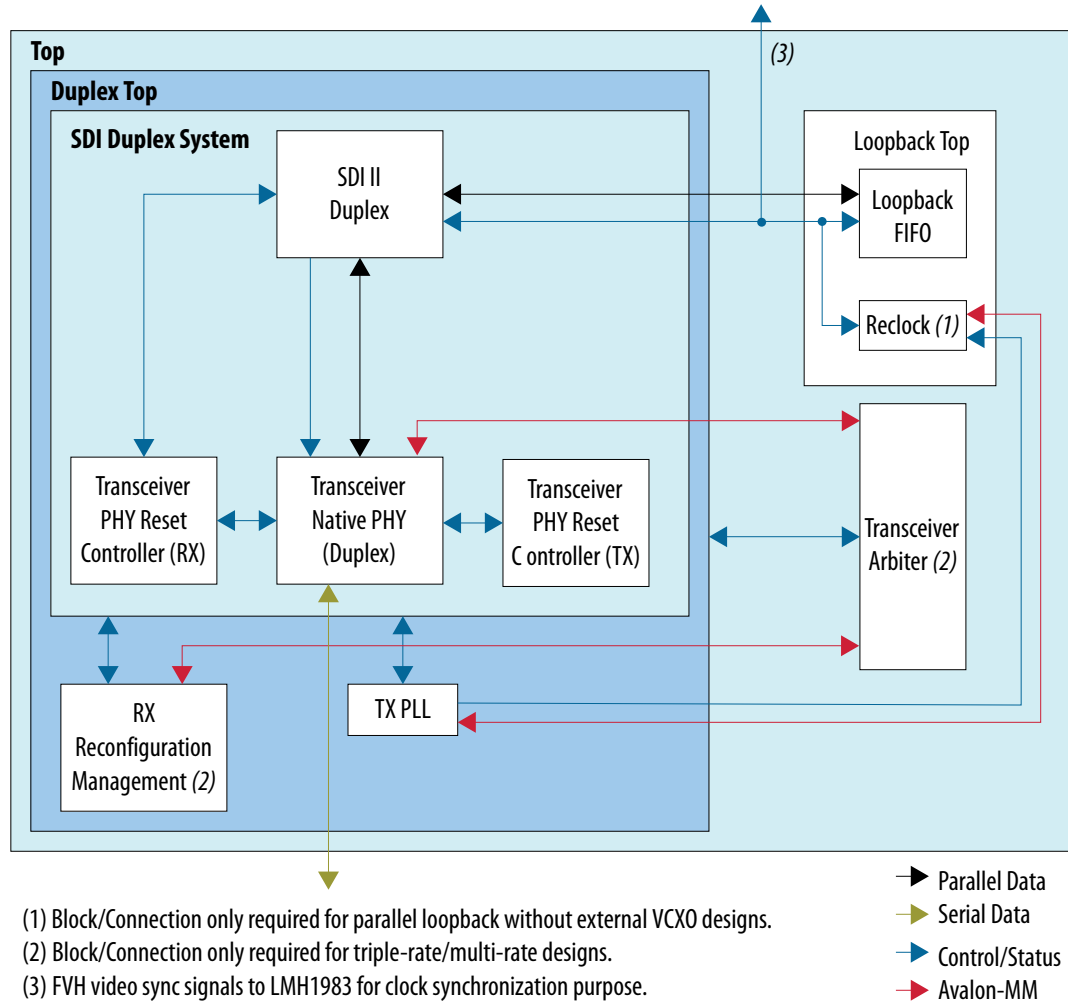
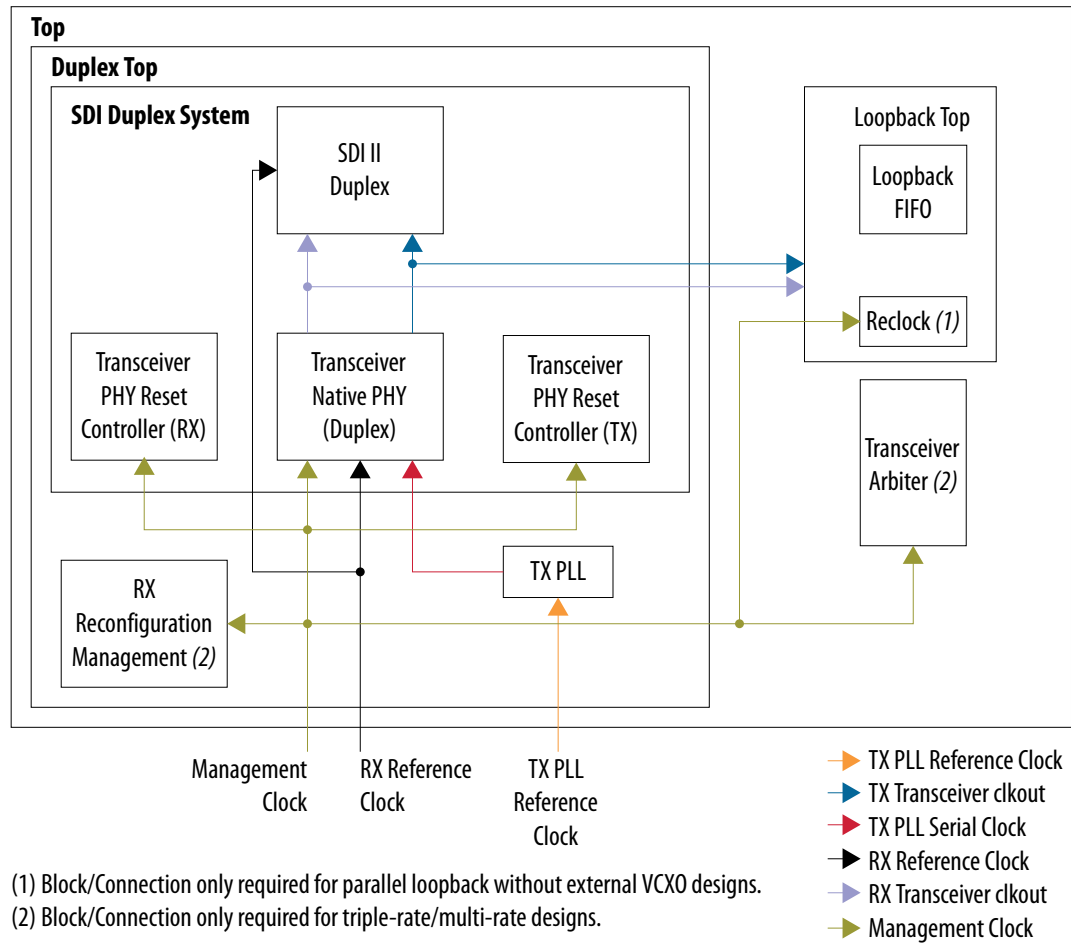




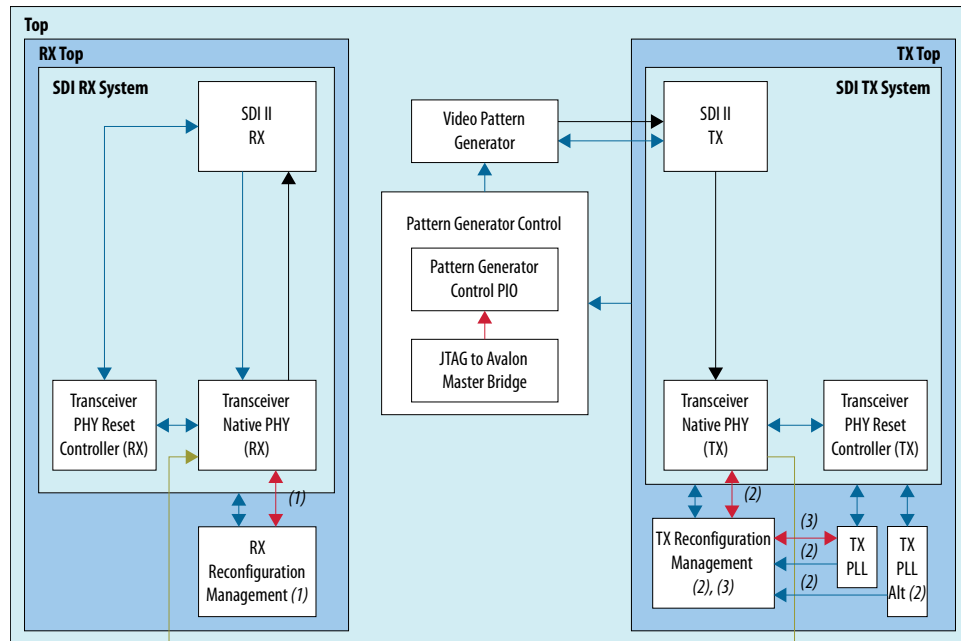
Figure 12. Parallel Loopback with Duplex Mode Clocking Scheme



## 2.2 Serial Loopback Design Examples

The serial loopback design examples demonstrate simplex and duplex channel modes.

Figure 13. Serial Loopback with Simplex Mode Block Diagram



- (1) Block/Connection only required for triple-rate/multi-rate designs.
- (2) Block/Connection only required for triple-rate/multi-rate for TX PLL switching designs.
- (3) Block/Connection only required for triple-rate/multi-rate for TX PLL reference clock switching designs.

- Parallel Data
- Serial Data
- Control/Status
- Avalon-MM



Figure 14. Serial Loopback with Simplex Mode Clocking Scheme

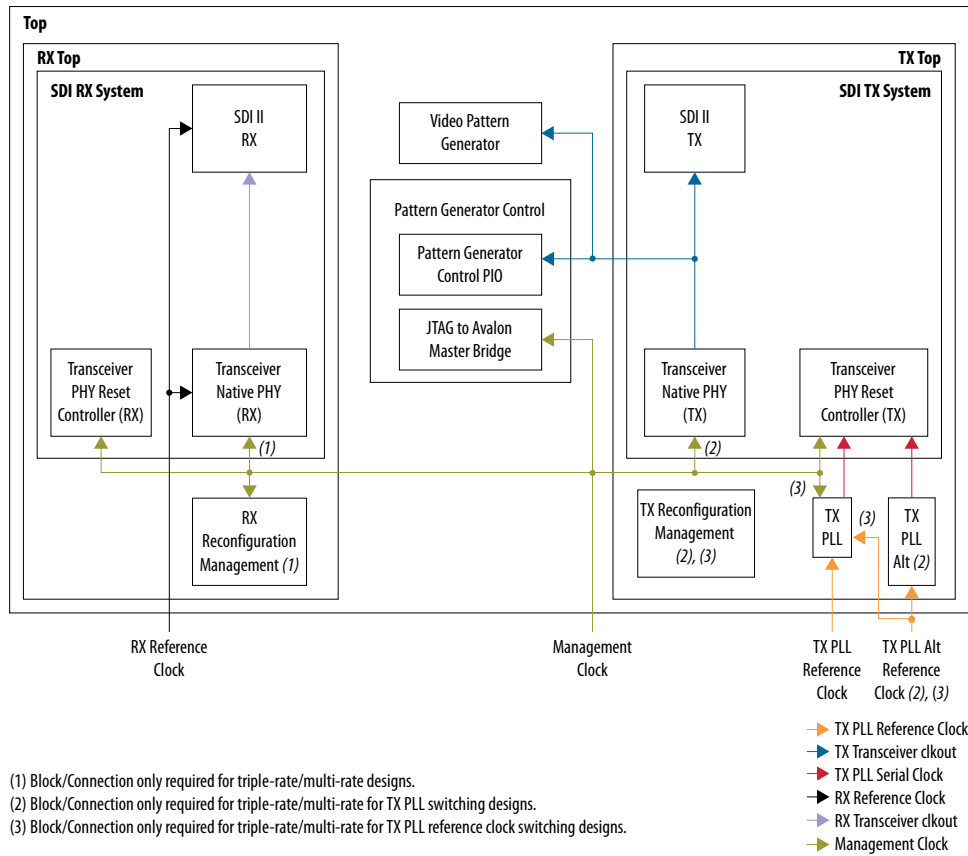
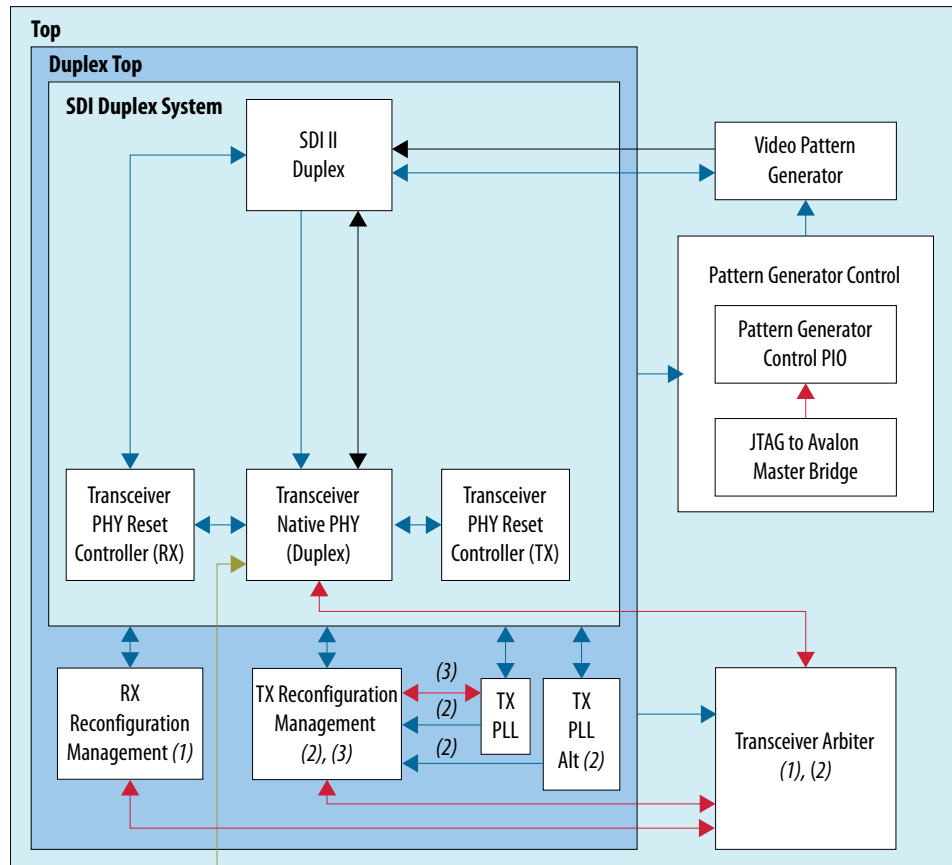


Figure 15. Serial Loopback with Duplex Mode Block Diagram



(1) Block/Connection only required for triple-rate/multi-rate designs.

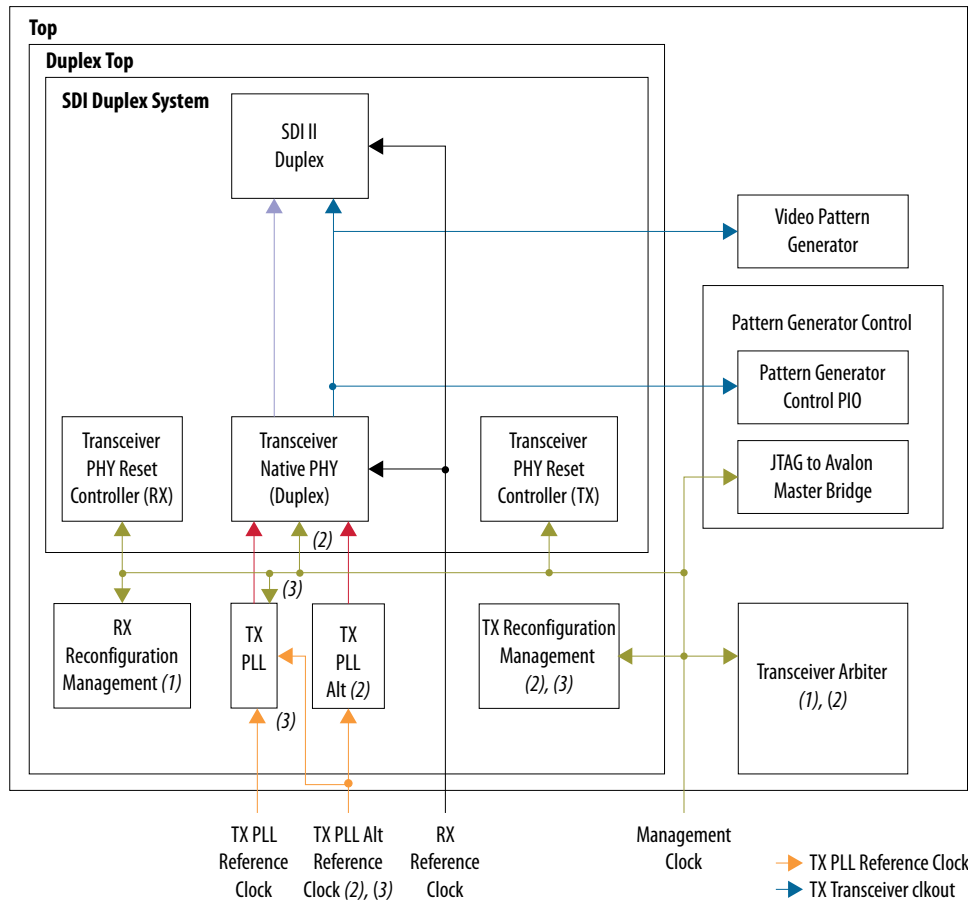
(2) Block/Connection only required for triple-rate/multi-rate for TX PLL switching designs.

(3) Block/Connection only required for triple-rate/multi-rate for TX PLL reference clock switching designs.

- Parallel Data
- Serial Data
- Control/Status
- Avalon-MM



Figure 16. Serial Loopback with Duplex Mode Clocking Scheme



- (1) Block/Connection only required for triple-rate/multi-rate designs.
- (2) Block/Connection only required for triple-rate/multi-rate for TX PLL switching designs.
- (3) Block/Connection only required for triple-rate/multi-rate for TX PLL reference clock switching designs.

- TX PLL Reference Clock
- TX Transceiver clkout
- TX PLL Serial Clock
- RX Reference Clock
- RX Transceiver clkout
- Management Clock

## 2.3 Design Components

The Intel FPGA SDI II IP core design examples require the following components.



**Table 9. Device Under Test (DUT) Components**

Design Component	Description
Intel FPGA SDI II IP Core	<ul style="list-style-type: none"> <li>• TX               <ul style="list-style-type: none"> <li>– The TX core receives the video data from the top level and encodes the necessary information, (e.g. line number (LN), cyclical redundancy check (CRC), payload ID), into the data stream(s).</li> <li>– In a multi-rate design, the TX core oversamples the received data up to 11.88 Gbps data rate for every video standard.</li> <li>– Specify the assignment of the parallel data interface (<code>tx_parallel_data</code>) to the transceiver based on the 11.88 Gbps data rate settings.</li> </ul> </li> <li>• RX               <ul style="list-style-type: none"> <li>– The RX core receives the parallel data from the Intel Cyclone 10 GX Transceiver Native PHY IP core and decodes information. This information includes descrambling, realigning data, and extracting the necessary information for user.</li> <li>– For a multi-rate design, due to the difference in data widths recovered for different video standards, rearrange <code>rx_parallel_data</code> from the transceiver before passing the data back to the protocol block.</li> </ul> </li> </ul>
Intel Cyclone 10 GX Transceiver Native PHY IP Core	<ul style="list-style-type: none"> <li>• TX               <p>Hard transceiver block that receives parallel data from the Intel FPGA SDI II IP core and serializes the data before transmission.</p> <ul style="list-style-type: none"> <li>– For HD/3G-SDI single-rate and triple-rate designs, enable the simplified data interface option to connect parallel data directly to the <code>tx_dataout</code> signal of the Intel FPGA SDI II IP core.</li> <li>– For a multi-rate design, disable this option due to the limitation in the 12G-SDI transceiver PHY settings.</li> </ul> </li> <li>• RX               <p>Hard transceiver block that receives serial data from an external video source.</p> <ul style="list-style-type: none"> <li>– For HD/3G-SDI single-rate and triple-rate designs, enable the simplified data interface option to connect parallel data directly to the <code>rx_datain</code> signal of Intel FPGA SDI II IP core.</li> <li>– For a multi-rate design, disable this option due to the limitation in the 12G-SDI transceiver PHY settings.</li> </ul> <p>You must connect the <code>rx_analogreset_ack</code> output signal from this block to the RX Reconfiguration Management module to indicate that the transceiver is in reset.</p> <p><i>Note:</i> For the duplex mode transceiver (SDI triple-rate parallel loopback with external VCXO design example), generate a dummy RX only PHY (<code>sdi_rx_phy.qsys</code>) to get the transceiver configuration files (<code>*_CFG0.sv</code>, <code>*_CFG1.sv</code>, ...) for RX reconfiguration. The generated configuration files from the duplex mode transceiver may contain some TX registers. You don't need to reconfigure the registers because only the SDI RX core requires transceiver reconfiguration.</p> </li> </ul>
Intel Cyclone 10 GX Transceiver PHY Reset Controller	<ul style="list-style-type: none"> <li>• TX               <ul style="list-style-type: none"> <li>– The reset input of this controller is triggered from the top level.</li> <li>– The controller generates the corresponding analog and digital reset signal to the Intel Cyclone 10 GX Transceiver Native PHY block, according to the reset sequencing inside the block.</li> <li>– Use the <code>tx_ready</code> output signal from the block as a reset signal to the TX core to indicate that the transceiver is up and running, and ready to receive data from the core.</li> </ul> </li> <li>• RX               <ul style="list-style-type: none"> <li>– The reset input of this controller is triggered by the Intel FPGA SDI II IP core.</li> <li>– The controller generates the corresponding analog and digital reset signal to the Intel Cyclone 10 GX Transceiver Native PHY block according to the reset sequencing inside the block.</li> </ul> </li> </ul>

*continued...*





Design Component	Description
RX Reconfiguration Management	RX transceiver reconfiguration management block that reconfigures the Intel Cyclone 10 GX Transceiver Native PHY block to receive different data rates from SD-SDI to 12G-SDI standards. To indicate the status of the transceiver, connect <code>rx_cal_busy</code> and from the transceiver to this block.
TX Reconfiguration Management	TX PLL or transceiver reconfiguration management block that reconfigures the TX PLL or Intel Cyclone 10 GX Transceiver Native PHY block to change the TX clock dynamically for switching between integer and fractional frame rates. The block requires <code>tx_cal_busy</code> , <code>pll_cal_busy</code> , and from the transceiver, and the PLLs to indicate the status of the transceiver in a TX PLL switching design.
TX PLL/TX PLL Alt	Transmitter PLL block that provides the serial fast clock to Transceiver Native PHY. <ul style="list-style-type: none"> <li>For TX PLL switching design, TX PLL is always configured to generate integer frame rate while TX PLL Alt is configured to generate fractional frame rate.</li> <li>For TX PLL reference clock switching design, TX PLL is configured to have reference clock 0 to generate integer frame rate and reference clock 1 to generate fractional frame rate.</li> <li>For single-rate and triple-rate designs, this PLL can be either CMU PLL or fPLL.</li> </ul> Move the TX PLL out from the TX top if you want to merge the PLL between multiple channels.

**Table 10. Loopback Components**

Component	Description
Loopback FIFO	This block contains a dual-clock FIFO (DCFIFO) buffer to handle the data transmission across asynchronous clock domains—the receiver recovered clock and transmitter clock out. <ul style="list-style-type: none"> <li>The receiver sends the decoded RX data to the transmitter through this FIFO buffer.</li> <li>When the receiver locks, the RX data is written to the FIFO buffer.</li> <li>The transmitter starts reading, encoding, and transmitting the data when half of the FIFO buffer is filled.</li> </ul>
Reclock	The parallel loopback without external VCXO design requires this module. This block compares the phase between the receiver and transmitter parallel clocks. The output interfaces of this block connect to the reconfiguration Avalon Memory-Mapped (Avalon-MM) interfaces of an fPLL. If there is any difference in the frequencies between the clock domains, this module generates the necessary signals to reconfigure the fPLL to match the clock frequencies as close as possible. <i>Note:</i> Applicable only for parallel loopback without external VCXO designs.

**Table 11. Video Pattern Generator Components**

Component	Description
Video Pattern Generator	Basic video pattern generator which supports SD-SDI up to 12G-SDI video formats with 4:2:2 YCbCr. The generator enables you to select static video with colorbar pattern or pathological pattern.
Pattern Gen Control PIO	Provides a memory-mapped interface for controlling the video pattern generator.
JTAG to Avalon Master Bridge	Provides System Console host access to the Parallel I/O (PIO) IP core in the design through the JTAG interface.



**Table 12. Common Block**

Component	Description
Transceiver Arbiter	<p>This generic functional block prevents transceivers from recalibrating simultaneously when either RX or TX transceivers within the same physical channel require reconfiguration. The simultaneous recalibration impacts applications where RX and TX transceivers within the same channel are assigned to independent IP implementations.</p> <p>This transceiver arbiter is an extension to the resolution recommended for merging simplex TX and simplex RX into the same physical channel. This transceiver arbiter also assists in merging and arbitrating the Avalon-MM RX and TX reconfiguration requests targeting simplex RX and TX transceivers within a channel as the reconfiguration interface port of the transceivers can only be accessed sequentially. The transceiver arbiter is not required when only either RX or TX transceiver is used in a channel.</p> <p>The transceiver arbiter identifies the requester of a reconfiguration through its Avalon-MM reconfiguration interfaces and ensures that the corresponding <code>tx_reconfig_cal_busy</code> or <code>rx_reconfig_cal_busy</code> is gated accordingly.</p>

## 2.4 Clocking Scheme Signals

The table lists the clocking scheme signals for the Intel FPGA SDI II IP core design examples.

**Table 13. Clocking Scheme Signals**

Clock	Signal Name in Design	Description
TX PLL Refclock	<code>tx_pll_refclk</code>	<p>TX PLL reference clock, of any frequency that is divisible by the transceiver for that data rate.</p> <p><i>Note:</i> You must connect this clock to a dedicated transceiver reference clock pin.</p> <ul style="list-style-type: none"> <li>Parallel loopback with external VCXO <ul style="list-style-type: none"> <li>Use a minimum clock frequency of 148.5 MHz (single-rate/triple-rate) or 297 MHz (multi-rate) to meet jitter performance specification.</li> <li>If you use a higher clock frequency, you would need to modify the TX PLL reference clock value in the TX PLL parameter editor.</li> </ul> </li> <li>Parallel loopback without external VCXO <ul style="list-style-type: none"> <li>The recommended frequency is 100 MHz.</li> </ul> </li> <li>Serial loopback <ul style="list-style-type: none"> <li>For this design, the TX PLL refclock is configured to generate clock for integer frame rate.</li> <li>The minimum clock frequency is 148.5 MHz (single-rate/triple-rate) or 297 MHz (multi-rate) to meet jitter performance specification.</li> <li>If you use a higher clock frequency, you would need to modify the TX PLL reference clock value in the TX PLL parameter editor.</li> </ul> </li> </ul>
TX PLL Alt Refclock	<code>tx_pll_refclk_alt</code>	<p>Second TX PLL reference clock which can be any clock frequency that is divisible by transceiver for that data rate. This clock must be connected to a dedicated transceiver reference clock pin.</p>

*continued...*



Clock	Signal Name in Design	Description
		<ul style="list-style-type: none"> <li>Serial loopback               <ul style="list-style-type: none"> <li>For this design example, TX PLL alt refclock is configured to generate clock for fractional frame rate.</li> <li>The minimum clock frequency is 148.35 MHz (single-rate/triple-rate) or 296.7 MHz (multi-rate) to meet jitter performance specification.</li> <li>If you use a higher clock frequency, you would need to modify the TX PLL reference clock value in the TX PLL parameter editor.</li> </ul> </li> </ul>
TX Transceiver Clockout	tx_vid_clkout	Recovered clock from the transceiver. <ul style="list-style-type: none"> <li>HD-SDI single rate               <ul style="list-style-type: none"> <li>74.25 MHz (default)</li> <li>74.1758 MHz (for the Dynamic TX clock switching feature when you transmit video format with fractional frame rate)</li> </ul> </li> <li>3G-SDI single rate, triple rate or multi rate               <ul style="list-style-type: none"> <li>148.5 MHz (default)</li> <li>148.35 MHz (for the Dynamic TX clock switching feature when you transmit video format with fractional frame rate)</li> </ul> </li> </ul>
TX PLL Serial Clock	tx_serial_clk	Serial fast clock generated by TX PLL. The clock frequency is set based on the data rate.
RX Refclock	rx_cdr_refclk	Transceiver clock data recovery (CDR) reference clock, of any frequency divisible by the transceiver for that data rate. Only a single reference clock frequency is required to support both integer and fractional frame rates. It must be a free running clock connected to the transceiver clock pin. For Intel Cyclone 10 GX devices, the multi-rate designs use a clock frequency of 297 MHz and the single-rate and triple-rate designs use a frequency of 148.5 MHz. Using a higher clock frequency would require a modification of the RX CDR reference clock value in the Intel Cyclone 10 GX Native PHY Transceiver parameter editor. For triple or multi-rate modes, you need to modify the reference clock value for every profile. <i>Note:</i> Do not share the TX PLL reference clock with the RX transceiver reference clock for a parallel loopback design. In parallel loopback designs, the TX PLL clock is tuned to match the RX recovered clock frequency.
	rx_core_refclk	SDI RX core reference clock. The required frequency is 148.5 MHz. This clock must be a free-running clock.
RX Transceiver Clkout	rx_vid_clkout	Recovered clock from the transceiver. <ul style="list-style-type: none"> <li>SD-SDI               <ul style="list-style-type: none"> <li>148.5 MHz (default)</li> </ul> </li> <li>HD-SDI               <ul style="list-style-type: none"> <li>74.25 MHz when receiving integer frame rate</li> <li>74.1758 MHz when receiving fractional frame rate</li> </ul> </li> <li>3G/6G/12-SDI               <ul style="list-style-type: none"> <li>148.5 MHz when receiving integer frame rate</li> <li>148.35 MHz when receiving fractional frame rate</li> </ul> </li> </ul>
Management Clock	rx_rcfg_mgmt_clk	A free-running 100 MHz RX clock used by Avalon-MM interfaces for reconfiguration and by the PHY reset controller for transceiver reset sequence.

*continued...*



Clock	Signal Name in Design	Description	
		<b>Component</b>	<b>Required Frequency (MHz)</b>
		Avalon-MM reconfiguration	100 – 125
		Transceiver PHY reset controller	1 – 500
		You may change the clock frequency as long as it is within the valid frequency range of the Avalon-MM reconfiguration and Transceiver PHY reset controller. For Transceiver PHY reset controller, you may need to modify the input clock frequency in the parameter editor.	
	tx_rcfg_mgmt_clk	A free-running 100 MHz TX clock used by Avalon-MM interfaces for reconfiguration and by the PHY reset controller for transceiver reset sequence.	
		<b>Component</b>	<b>Required Frequency (MHz)</b>
		Avalon-MM reconfiguration	100 – 125
		Transceiver PHY reset controller	1 – 500
		You may change the clock frequency as long as it is within the valid frequency range of the Avalon-MM reconfiguration and Transceiver PHY reset controller. For Transceiver PHY reset controller, you may need to modify the input clock frequency in the parameter editor.	

## 2.5 Interface Signals

The tables list the signals for the Intel FPGA SDI II IP core design examples.

**Table 14. Top-Level Signals**

Signal	Direction	Width	Description
<b>On-board Oscillator Signals</b>			
c10_refclk_2_p	Input	1	100 MHz clock for reconfiguration Avalon-MM interfaces.
usb_refclk_p	Input	1	125 MHz dedicated transceiver clock. Programmable to 148.5/100 MHz from the Clock Controller GUI.
sfp_refclk_p	Input	1	644.53 MHz dedicated transceiver clock. Programmable to 148.3516/148.5/296.7033/297 MHz from the Clock Controller GUI.
<b>User Push Buttons and LEDs</b>			
user_pb0	Input	1	Push button for the LEDs to switch between displaying rx_std or rx lock status.
user_pb1	Input	1	Push button to power down LMK03328 after switching the jumper settings.
user_pb2	Input	1	Push button for global reset.
user_led	Output	4	Green LED display.



Nextera SDI FMC Daughter Card Pins on FMC Port			
fmc_gbtclk_m2c_p0	Input	1	297 or 296.7 MHz dedicated transceiver reference clock from FMC port.
fmc_dp_m2c_p2	Input	1	SDI RX serial data from FMC port.
fmc_la_tx_p1	Input	1	RX cable equalizer lock status on Nextera daughter card.
fmc_dp_c2m_p0	Output	1	SDI TX serial data from FMC port.
fmc_la_tx_p12	Output	1	Initialize LMH1983 on Nextera daughter card.
fmc_la_tx_n12	Output	1	F sync signal LMH1983 on Nextera daughter card.
fmc_la_tx_p14	Output	1	V sync signal LMH1983 on Nextera daughter card.
fmc_la_tx_n14	Output	1	H sync signal LMH1983 on Nextera daughter card.
fmc_la_tx_p15	Output	1	Power-down signal LMH1983 on Nextera daughter card.

**Table 15. RX/TX/DU Top Signals**

Signal	Direction	Width	Description
<b>Clocks</b>			
rx_cdr_refclk	Input	1	RX transceiver reference clock. This clock must be a free-running clock.
rx_core_refclk	Input	1	SDI RX core clock. This clock must be a free-running clock.
tx_pll_refclk	Input	1	TX PLL reference clock. This clock must be a free-running clock.
tx_pll_refclk_alt	Input	1	Secondary TX PLL reference clock. This clock must be a free-running clock.
rx_rcfg_mgmt_clk	Input	1	RX reconfiguration management clock, Avalon-MM interface clock, and PHY reset control input clock. This clock must be a free-running clock.
tx_rcfg_mgmt_clk	Input	1	TX reconfiguration management clock, and Avalon-MM interface clock, and PHY reset control input clock. This clock must be a free-running clock.
rx_vid_clkout	Output	1	RX transceiver recovered parallel clock for video data.
tx_vid_clkout	Output	1	TX transceiver recovered parallel clock for video data.

<b>Reset</b>			
tx_resetrn	Input	1	TX core and PHY reset signal.
rx_resetrn	Input	1	RX core and PHY reset signal.
tx_rcfg_mgmt_resetrn	Input	1	TX reconfiguration reset signal.
rx_rcfg_mgmt_resetrn	Input	1	RX reconfiguration reset signal.
sdi_rx_rst_proto_out	Output	1	Reset signal generated to reset the receiver downstream protocol logic. This generated reset signal is synchronous to rx_vid_clkout clock domain.

<b>Video Signal Interfaces (Interface with Video Image and Processing (VIP) Components)</b>			
rx_vid_data	Output	20*N	Receiver parallel video data out.

*continued...*



Video Signal Interfaces (Interface with Video Image and Processing (VIP) Components)			
			<i>Note:</i> $N = 4$ (multi-rate design) or 1 (triple-rate design)
rx_vid_datavalid	Output	1	Data valid signal generated from SDI RX core. The timing must be synchronous to rx_vid_clkout and has the following settings: <ul style="list-style-type: none"> <li>SD-SDI: 1H 4L 1H 5L</li> <li>HD/3G/6G/12G-SDI: H</li> </ul>
rx_vid_std	Output	3	Received video standard. <ul style="list-style-type: none"> <li>3'b000: SD-SDI</li> <li>3'b001: HD-SDI</li> <li>3'b011: 3G-SDI Level A</li> <li>3'b010 3G-SDI Level B</li> <li>3'b101: 6G-SDI 4 Streams Interleaved</li> <li>3'b100: 6G-SDI 8 Streams Interleaved</li> <li>3'b111: 12G-SDI 8 Streams Interleaved</li> <li>3'b110: 12G-SDI16 Streams Interleaved</li> </ul>
rx_vid_locked	Output	1	Frame locked indicates that the IP core has spotted multiple frames with the same timing.
rx_vid_hsync	Output	$N$	Horizontal blanking interval timing signal. The receiver asserts this signal when the horizontal blanking interval is active. <i>Note:</i> $N = 4$ (multi-rate design) or 1 (triple-rate design)
rx_vid_vsync	Output	$N$	Vertical blanking interval timing signal. The receiver asserts this signal when the vertical blanking interval is active. <i>Note:</i> $N = 4$ (multi-rate design) or 1 (triple-rate design)
rx_vid_f	Output	$N$	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. <i>Note:</i> $N = 4$ (multi-rate design) or 1 (triple-rate design)
rx_vid_trs	Output	$N$	On-board SDI TX cable driver slew rate control. <i>Note:</i> $N = 4$ (multi-rate design) or 1 (triple-rate design)
tx_vid_data	Output	$20*N$	Receiver output signal that indicates current word is timing reference signal (TRS). This signal asserts at the first word of 3FF 000 000 TRS. <i>Note:</i> $N = 4$ (multi-rate design) or 1 (triple-rate design)
tx_vid_datavalid	Input	1	Transmitter parallel data valid. The timing (H: High, L: Low) must be synchronous to tx_pclk clock domain and has the following settings: <ul style="list-style-type: none"> <li>SD-SDI = 1H 4L 1H 5L</li> <li>HD-SDI = H (for single-rate) and 1H 1L (triple-rate/multi-rate)</li> <li>3G/6G/12G-SDI = H</li> </ul>
tx_vid_std	Input	3	Indicates the desired transmit video standard. <ul style="list-style-type: none"> <li>3'b000: SD-SDI</li> <li>3'b001: HD-SDI</li> <li>3'b011: 3G-SDI Level A</li> <li>3'b010 3G-SDI Level B</li> <li>3'b101: 6G-SDI 4 Streams Interleaved</li> </ul>

**continued...**



Video Signal Interfaces (Interface with Video Image and Processing (VIP) Components)			
			<ul style="list-style-type: none"> <li>3'b100: 6G-SDI 8 Streams Interleaved</li> <li>3'b111: 12G-SDI 8 Streams Interleaved</li> <li>3'b110: 12G-SDI16 Streams Interleaved</li> </ul>
tx_vid_trs	Input	1	Transmitter TRS input. For use in LN, CRC, or payload ID insertion. Assert on the first word of both end of active video (EAV) TRS and start of active video (SAV) TRS.

Other SDI Video Protocol Interfaces			
sdi_tx_enable_crc	Input	1	Enable CRC insertion for all SDI video standards, except SD-SDI.
sdi_tx_enable_ln	Input	1	Enable LN insertion for all SDI video standards, except SD-SDI.
sdi_tx_ln	Input	11*N	LN insertion in the data stream when sdi_tx_enable_ln = 1. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
sdi_tx_ln_b	Input	11*N	LN insertion in the data stream when sdi_tx_enable_ln = 1. Only for 3G level B, 6G 8 streams interleaved, and 12G 16 streams interleaved. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
sdi_tx_vpid_overwrite	Input	1	Enable this signal to overwrite the existing payload ID embedded in the data stream.
sdi_tx_line_f0	Input	11*N	Indicates the line number to be inserted with the payload ID.
sdi_tx_line_f1	Input	11*N	
sdi_tx_vpid_byte1	Input	8*N	Payload ID byte to be inserted in the payload ID field.
sdi_tx_vpid_byte2	Input	8*N	
sdi_tx_vpid_byte3	Input	8*N	
sdi_tx_vpid_byte4	Input	8*N	
sdi_tx_vpid_byte1_b	Input	8*N	
sdi_tx_vpid_byte2_b	Input	8*N	
sdi_tx_vpid_byte3_b	Input	8*N	
sdi_tx_vpid_byte4_b	Input	8*N	
sdi_rx_coreclk_is_ntsc_paln	Input	1	To indicate whether rx_coreclk is 148.5 MHz or 148.35 MHz: <ul style="list-style-type: none"> <li>0: 148.5 MHz</li> <li>1: 148.35 MHz</li> </ul>
sdi_tx_datavalid	Output	1	Data valid signal generated from SDI TX core. The timing (H: High, L: Low) is synchronous to tx_vid_clkout and has the following settings: <ul style="list-style-type: none"> <li>SD-SDI = 1H 4L 1H 5L</li> <li>HD-SDI = H (for single-rate) and 1H 1L (triple-rate/multi-rate)</li> <li>3G/6G/12G-SDI = H</li> </ul>

continued...



Other SDI Video Protocol Interfaces			
sdi_rx_align_locked	Output	1	Alignment locked indicating the IP core has spotted a TRS and word alignment performed.
sdi_rx_trs_locked	Output	$N$	TRS locked indicating the IP core has spotted six consecutive TRS with same timing. <i>Note:</i> $N = 4$ (multi-rate design) or 1 (triple-rate design)
sdi_rx_clkout_is_ntsc_paln	Output	1	Indicates that the receiver is receiving video rate at integer or fractional frame rate: <ul style="list-style-type: none"> <li>0: Integer frame rate</li> <li>1: Fractional frame rate</li> </ul>
sdi_rx_format	Output	$4*N$	Received video transport format. Refer to the <i>SDI II IP User Guide</i> for the encoding value. <i>Note:</i> $N = 4$ (multi-rate design) or 1 (triple-rate design)
sdi_rx_ap	Output	$N$	Active picture interval timing signal. This signal asserts when the active picture interval is active.
sdi_rx_eav	Output	$N$	Receiver output signal that indicates current TRS is EAV. This signal is asserted at the fourth word of TRS, which is the XYZ word.
sdi_rx_ln	Output	$11*N$	Received line number from protocol.
sdi_rx_ln_b	Output	$11*N$	
sdi_rx_crc_error_c	Output	$N$	CRC error status signal from protocol.
sdi_rx_crc_error_y	Output	$N$	
sdi_rx_crc_error_c_b	Output	$N$	
sdi_rx_crc_error_y_b	Output	$N$	
sdi_rx_line_f0	Output	$11*N$	
sdi_rx_line_f1	Output	$11*N$	
sdi_rx_vpid_byte1	Output	$8*N$	
sdi_rx_vpid_byte2	Output	$8*N$	
sdi_rx_vpid_byte3	Output	$8*N$	
sdi_rx_vpid_byte4	Output	$8*N$	
sdi_rx_vpid_checksum_error	Output	$N$	
sdi_rx_vpid_valid	Output	$N$	
sdi_rx_vpid_byte1_b	Output	$8*N$	
sdi_rx_vpid_byte2_b	Output	$8*N$	
sdi_rx_vpid_byte3_b	Output	$8*N$	
sdi_rx_vpid_byte4_b	Output	$8*N$	
sdi_rx_vpid_checksum_error_b	Output	$N$	
sdi_rx_vpid_valid_b	Output	$N$	





Transceiver Interfaces			
tx_pll_refclk_sel	Input	1	Indicate which of pll_locked signals to be monitored for TX PHY reset controller's reset sequencing. Always set to 1'b0 if only one PLL is in use.
tx_rcfg_cal_busy	Input	1	Transceiver calibration status to TX PHY reset controller.
rx_rcfg_cal_busy	Input	1	Transceiver calibration status to RX PHY reset controller and Rx reconfiguration management module.
gxb_rx_serial_data	Input	1	RX transceiver serial data.
gxb_tx_serial_data	Output	1	TX transceiver serial data.
gxb_rx_ready	Output	1	RX transceiver status.
gxb_tx_ready	Output	1	TX transceiver status.
gxb_rx_cal_busy	Output	1	Calibration status signal from RX transceiver.
gxb_tx_cal_busy	Output	1	Calibration status signal from TX transceiver.
tx_pll_locked	Output	1	TX PLL lock status.
tx_pll_locked_alt	Output	1	TX PLL alt lock status.
cdr_reconfig_busy	Output	1	RX CDR reconfiguration status.
tx_reconfig_busy	Output	1	TX PLL/transceiver reconfiguration status.

Transceiver Reconfiguration Interfaces			
gxb_du_rcfg_write	Input	1	Reconfiguration interface signals from transceiver arbiter to duplex mode transceiver.
gxb_du_rcfg_read	Input	1	
gxb_du_rcfg_address	Input	10	
gxb_du_rcfg_writedata	Input	32	
gxb_du_rcfg_readdata	Output	32	
gxb_du_rcfg_waitrequest	Output	1	
gxb_rx_rcfg_write	Input	1	Reconfiguration interface signals from transceiver arbiter to RX transceiver.
gxb_rx_rcfg_read	Input	1	
gxb_rx_rcfg_address	Input	10	
gxb_rx_rcfg_writedata	Input	32	
gxb_rx_rcfg_readdata	Output	32	
gxb_rx_rcfg_waitrequest	Output	1	
gxb_tx_rcfg_write	Input	1	Reconfiguration interface signals from transceiver arbiter to TX transceiver.
gxb_tx_rcfg_read	Input	1	
gxb_tx_rcfg_address	Input	10	

*continued...*



Transceiver Reconfiguration Interfaces			
gxb_tx_rcfg_writedata	Input	32	
gxb_tx_rcfg_readdata	Output	32	
gxb_tx_rcfg_waitrequest	Output	1	
rx_rcfg_readdata	Input	32	Reconfiguration interface signals from RX reconfiguration management module to transceiver arbiter.
rx_rcfg_waitrequest	Input	1	
rx_rcfg_write	Output	1	
rx_rcfg_read	Output	1	
rx_rcfg_address	Output	10	
rx_rcfg_writedata	Output	32	
tx_rcfg_readdata	Input	32	
tx_rcfg_waitrequest	Input	1	
tx_rcfg_write	Output	1	
tx_rcfg_read	Output	1	
tx_rcfg_address	Output	10	
tx_rcfg_writedata	Output	32	
tx_fpll_rcfg_write	Input	1	Reconfiguration interface signals to fPLL Avalon-MM interface.
tx_fpll_rcfg_read	Input	1	
tx_fpll_rcfg_writedata	Input	32	
tx_fpll_rcfg_address	Input	10	
tx_fpll_rcfg_readdata	Output	32	
tx_fpll_rcfg_waitrequest	Output	1	

**Table 16. Loopback Top Signals**

Signal	Direction	Width	Description
<b>Clocks</b>			
sdi_tx_clkout	Input	1	TX transceiver recovered parallel clock for video data.
sdi_rx_clkout	Input	1	RX transceiver recovered parallel clock for video data.
sdi_reclk_sysclk	Input	1	Input clock for reclock module (without external VCXO solution). This clock should be the same as fPLL reconfig_clk.



Resets			
sdi_rx_rst_proto	Input	1	Reset signal from SDI RX core to indicate that the protocol is currently held in reset.
sdi_reclk_rst	Input	1	Reset signal to reclock module (without external VCXO solution).

SDI Related Signals			
sdi_rx_dataout	Input	20*N	Receiver recovered parallel video data. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
sdi_rx_dataout_valid	Input	1	Data valid signal generated from SDI RX core.
sdi_rx_std	Input	3	Received video standard from SDI RX core.
sdi_rx_trs	Input	N	Receiver output signal from SDI II IP core that indicates current word is TRS. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
sdi_rx_trs_locked	Input	N	TRS locked status signal from SDI RX core. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
sdi_rx_frame_locked	Input	1	Frame locked status signal from SDI RX core.
sdi_tx_dataout_valid	Input	1	Data valid signal generated from SDI TX core.
sdi_rx_h	Input	1	Horizontal blanking interval timing signal extracted from SDI RX core.
sdi_rx_format	Input	4	Received video transport format.
sdi_rx_clkout_is_ntsc_paln	Input	1	Indication from SDI RX core that the receiver is receiving video rate at integer or fractional frame rate.
sdi_tx_datain	Output	20*N	Parallel video data input to SDI TX core. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
sdi_tx_datain_valid	Output	1	Data valid for the transmitter parallel data to SDI TX core.
sdi_tx_trs	Output	1	Transmitter TRS input to indicate that the current word is a TRS to SDI TX core.
sdi_tx_std	Output	3	Indicates the desired transmit video standard to SDI TX core.

fPLL Reconfiguration Signals			
pll_locked	Input	1	PLL lock status signal.
pll_reconfig_readdata	Input	32	Reconfiguration interface signals to fPLL Avalon-MM interface.
pll_reconfig_waitrequest	Input	1	
pll_reconfig_write	Output	1	
pll_reconfig_read	Output	1	
pll_reconfig_writedata	Output	32	
pll_reconfig_address	Output	10	



**Table 17. Transceiver Arbiter Signals**

Signal	Direction	Width	Description
<b>On-board Oscillator Signals</b>			
clk	Input	1	Reconfiguration clock. This clock should be sharing the same clock as reconfiguration management blocks.
reset	Input	1	Reset signal. This reset should be sharing the same reset as reconfiguration management blocks.
rx_rcfg_en	Input	1	RX reconfiguration enable signal.
tx_rcfg_en	Input	1	TX reconfiguration enable signal.
rx_rcfg_ch	Input	2	Indicates which channel to be reconfigured on RX. Always assign to 2'b00 for SDI case.
tx_rcfg_ch	Input	2	Indicates which channel to be reconfigured on TX. Always assign to 2'b00 for SDI case.
rx_reconfig_mgmt_write	Input	1	Reconfiguration Avalon-MM interfaces from RX reconfiguration management.
rx_reconfig_mgmt_read	Input	1	
rx_reconfig_mgmt_address	Input	10	
rx_reconfig_mgmt_writedata	Input	32	
rx_reconfig_mgmt_readdata	Output	32	
rx_reconfig_mgmt_waistrequest	Output	1	
tx_reconfig_mgmt_write	Input	1	Reconfiguration Avalon-MM interfaces from TX reconfiguration management.
tx_reconfig_mgmt_read	Input	1	
tx_reconfig_mgmt_address	Input	10	
tx_reconfig_mgmt_writedata	Input	32	
tx_reconfig_mgmt_readdata	Output	32	
tx_reconfig_mgmt_waistrequest	Output	1	
reconfig_write	Output	1	Reconfiguration Avalon-MM interfaces to transceiver.
reconfig_read	Output	1	
reconfig_address	Output	10	
reconfig_writedata	Output	32	
rx_reconfig_readdata	Input	32	

*continued...*



Signal	Direction	Width	Description
<b>On-board Oscillator Signals</b>			
rx_reconfig_waitrequest	Input	1	
tx_reconfig_readdata	Input	1	
tx_reconfig_waitrequest	Input	1	
rx_cal_busy	Input	1	Calibration status signal from RX transceiver.
tx_cal_busy	Input	1	Calibration status signal from TX transceiver.
rx_reconfig_cal_busy	Output	1	Calibration status signal to RX transceiver PHY reset control.
tx_reconfig_cal_busy	Output	1	Calibration status signal from TX transceiver PHY reset control.

<b>Video Pattern Generator Signals</b>			
clk	Input	1	Clock signal. This clock must be connected to the tx_vid_clkout input signal on TX/Du top.
rst	Input	1	Reset signal. This reset signal should be synchronized with the tx_vid_clkout clock signal from the TX/Du top.
bar_100_75n	Input	1	Enable this signal to generate 100% color-bar pattern. Disable to generate 75% color-bar pattern.
enable	Input	1	This signal acts as a data valid signal to this module. This signal should be connected to the sdi_tx_datavalid signal from the TX/Du top.
patho	Input	1	Enable this signal to generate pathological pattern.
blank	Input	1	Enable this signal to generate blank signal.
no_color	Input	1	Enable this signal to generate bar with no color.
sgmt_frame	Input	1	Enable this signal to generate payload ID for segmented frame video format when generating 1080i50 or 1080i60 video.
tx_std	Input	3	Indicates the desired transmit video standard. This input signal must match tx_vid_std on the TX/Du top.
tx_format	Input	4	Indicates the desired transmit video format.
dl_mapping	Input	1	Enable this signal to generate data streams with dual-link mapping. <i>Note:</i> Applicable only for HD dual link or 3G Level B dual link video standard.
ntsc_paln	Input	1	Enable this signal to generate payload ID for fractional frame rate video format. Disable to generate integer frame rate video format.
dout	Output	20*N	Data output signal to be connected to the tx_vid_data input signal on the TX/Du top. <i>Note:</i> N = 4 (multi-rate design) or 1 (triple-rate design)
dout_valid	Output	1	Data valid output signal to be connected to the tx_vid_datavalid input signal on the TX/Du top.
<b>continued...</b>			



Video Pattern Generator Signals			
trs	Output	1	TRS output signal to be connected to the tx_vid_trs input signal on the TX/Du top.
ln	Output	11*N	Line number output signal to be connected to the sdi_tx_ln input signal on the TX/Du top. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
dout_b	Output	20*N	Data output signal for link B (HD dual link). <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
dout_valid_b	Output	1	Data valid output signal for link B (HD dual link).
trs_b	Output	1	TRS output signal for link B (HD dual link).
ln_b	Output	11*N	Line number output signal to be connected to the sdi_tx_ln_b input signal on the TX/Du top. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
vpid_byte1	Input	8*N	The payload ID output signal to be connected to sdi_tx_vpid_byte1 input signal on TX/Du top. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
vpid_byte2	Input	8*N	The payload ID output signal to be connected to sdi_tx_vpid_byte2 input signal on TX/Du top. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
vpid_byte3	Input	8*N	The payload ID output signal to be connected to sdi_tx_vpid_byte3 input signal on TX/Du top. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
vpid_byte4	Input	8*N	The payload ID output signal to be connected to sdi_tx_vpid_byte4 input signal on TX/Du top. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
vpid_byte1_b	Input	8*N	The payload ID output signal to be connected to sdi_tx_vpid_byte1_b input signal on TX/Du top. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
vpid_byte2_b	Input	8*N	The payload ID output signal to be connected to sdi_tx_vpid_byte2_b input signal on TX/Du top. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
vpid_byte3_b	Input	8*N	The payload ID output signal to be connected to sdi_tx_vpid_byte3_b input signal on TX/Du top. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
vpid_byte4_b	Input	8*N	The payload ID output signal to be connected to sdi_tx_vpid_byte4_b input signal on TX/Du top. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
line_f0	Output	11*N	The line number output signal to be inserted with the payload ID. This signal must connect to sdi_tx_line_f0 input signal on TX/Du top. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>
line_f1	Output	11*N	The line number output signal to be inserted with the payload ID. This signal must connect to sdi_tx_line_f1 input signal on TX/Du top. <i>Note: N = 4 (multi-rate design) or 1 (triple-rate design)</i>



Pattern Generator Control Module Signals			
avmm_clk_in_clk	Input	1	Clock signal to Avalon-MM interface.
tx_clkout_in_clk	Input	1	Clock signal to Parallel I/O (PIO) IP. This clock must share the same clock as video pattern generator.
avmm_clk_reset_n	Input	1	Reset signal to Avalon-MM interface.
pattgen_rst_reset_in_0	Input	1	Input reset signals to a reset synchronizer which synchronizes the reset to the tx_clkout_in_clk clock domain.
pattgen_rst_reset_in_1	Input	1	
pattgen_rst_reset_out	Input	1	Output reset from the reset synchronizer. This reset synchronizes to the tx_clkout_in_clk clock domain and connects to the video pattern generator's input reset.
pattgen_ctrl_pio_out_port	Output	12	Output control signal from PIO to control the video pattern generator.

## 2.6 Video Pattern Generator Parameters

Customize the video pattern generator parameters according to your design.

**Table 18. Video Pattern Generator Parameters**

Parameter	Valid Value	Default Value	Description
OUTW_MULTP	1, 4	1	Defines the width of the output ports. Select 4 for a multi-rate design, otherwise select 1.
SD_BIT_WIDTH	10, 20	10	Defines the generated SD interface bit width. This value must match with the SD interface bit width parameter of the SDI II TX core in the same design.
TEST_GEN_ANC	0, 1	0	Select 1 to generate the ancillary data packet in output stream. The module inserts the embedded Data ID (DID) packet with 10'h242 if TEST_GEN_VPID is not enabled.
TEST_GEN_VPID	0, 1	0	Select 1 to generate the payload ID packet in output streams. The module inserts the embedded Data ID (DID) packet with 10'h241.

## 2.7 Hardware Setup

To run the hardware test for parallel loopback designs, connect an SDI video generator to the receiver input pin.

- Connect an external video analyzer to the TX instance to verify full functionality.
- To validate if the RX core locks to the signal and receives the video data correctly, use the on-board LEDs that display the RX status.



To run the hardware test for serial loopback designs, connect the transmitter output pin directly to the receiver input pin.

- To validate if the RX core locks to the signal and receives the video data correctly, use the on-board LEDs that display the RX status.
- You may also connect an SDI signal analyzer to the transmitter output pin to view the generated image.

**Table 19. On-board User LED Functions**

SWS ON/OFF	Function
ON	D21, D20, and D19 indicate the receiver video standard: <ul style="list-style-type: none"><li>• 000: SD-SDI</li><li>• 001: HD-SDI</li><li>• 010: 3G Level B</li><li>• 011: 3G Level A</li><li>• 100: 6G 8 Streams Interleaved</li><li>• 101: 6G 4 Streams Interleaved</li><li>• 110: 12G 16 Streams Interleaved</li><li>• 111: 12G 8 Streams Interleaved</li></ul>
OFF	<ul style="list-style-type: none"><li>• D21: Illuminates when <code>align_locked</code> asserts</li><li>• D20: Illuminates when <code>trs_locked</code> asserts</li><li>• D19: Illuminates when <code>frame_locked</code> asserts</li></ul>

## 2.8 Simulation Testbench

The simulation testbench checks for the assertion of the `trs_locked` signal. The testbench also detects the number of transceiver reconfiguration triggered after every video standard switching.





Figure 17. Simplex Mode Simulation Testbench Block Diagram

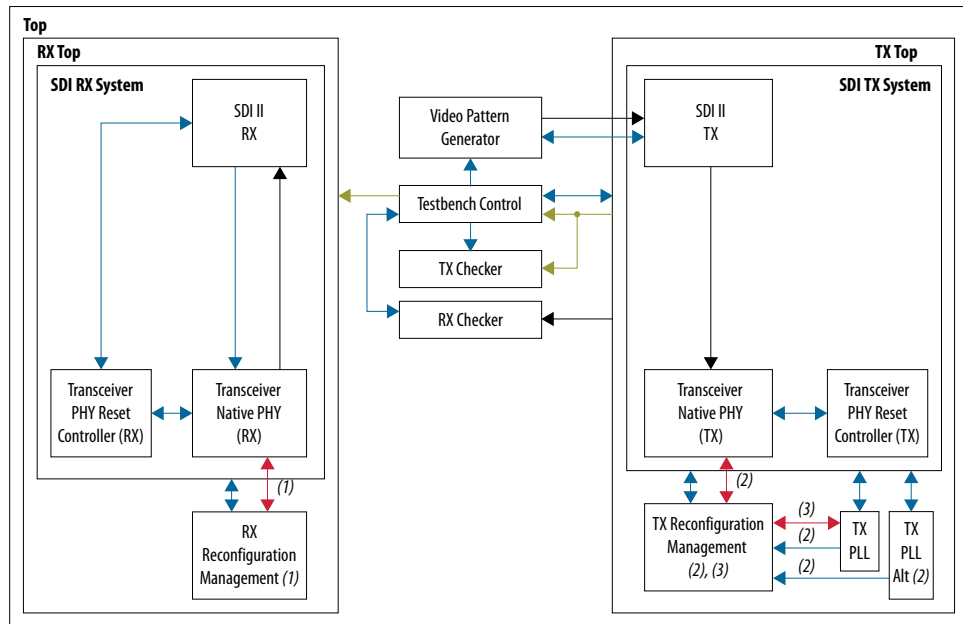
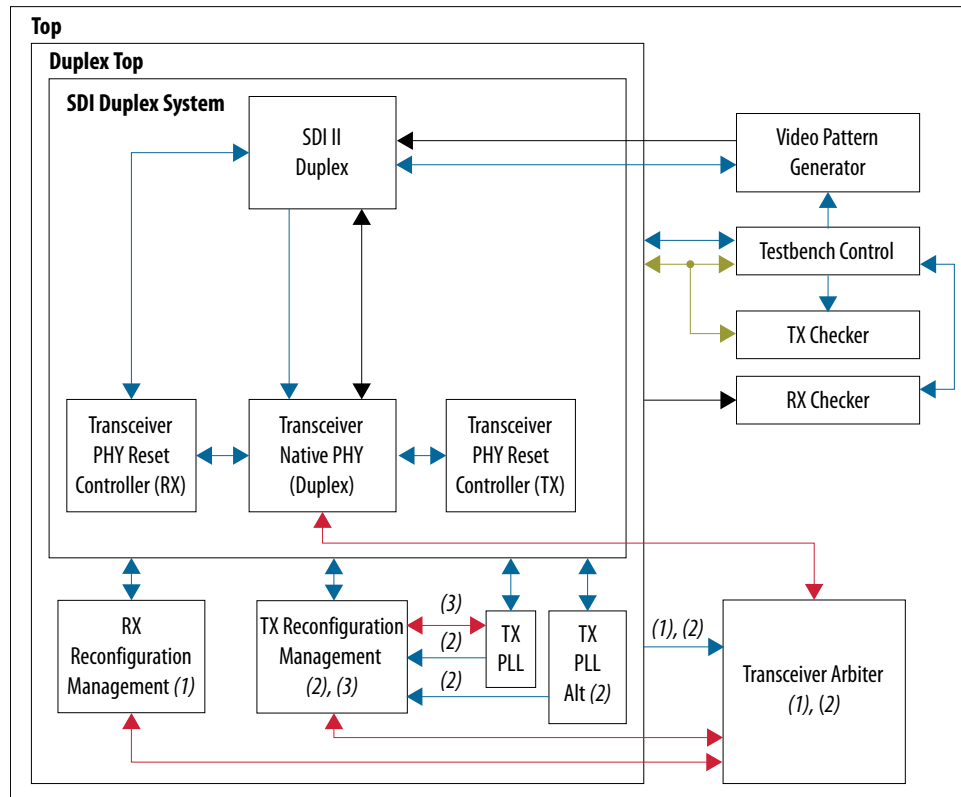


Figure 18. Duplex Mode Simulation Testbench Block Diagram



- (1) Block/Connection only required for triple-rate/multi-rate designs.
- (2) Block/Connection only required for TX PLL switching designs.
- (3) Block/Connection only required for TX PLL reference clock switching designs

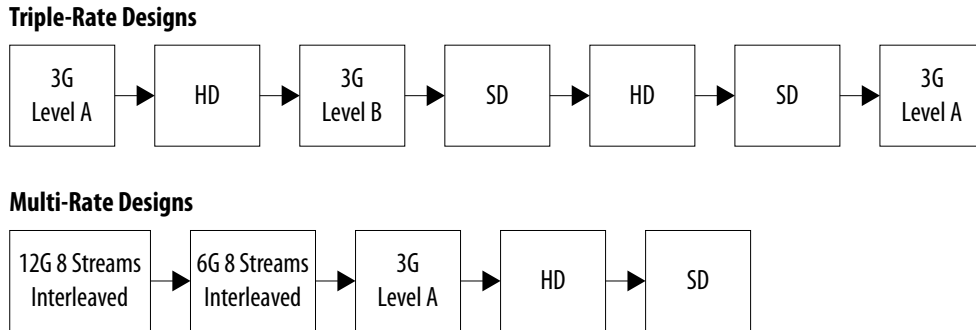
- Parallel Data
- Serial Data
- Control/Status
- Avalon-MM

Table 20. Testbench Components

Component	Description
Testbench Control	This block controls the test sequence of the simulation and generates the necessary stimulus signals to the TX and video pattern generator blocks.
RX Checker	This checker detects the <code>trs_locked</code> signal from the RX protocol and compares the actual number of transceiver reconfigurations performed versus the expected number.
TX Checker	This checker verifies if the TX serial data contains a valid TRS signal.



**Figure 19. Sequence of Video Standards for Triple-Rate and Multi-Rate Designs**



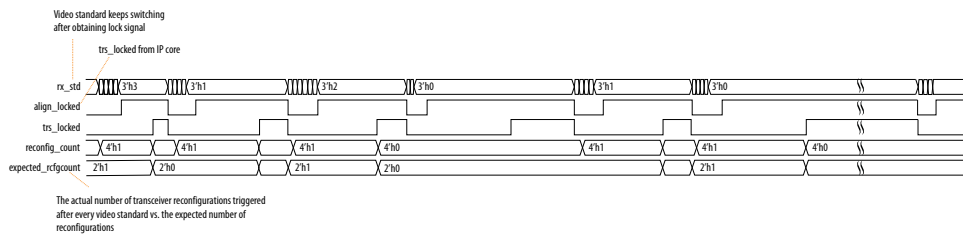
For single-rate designs, only one video standard is tested:

- HD-SDI single-rate—HD
- 3G-SDI single-rate—3G Level A

If you enable the **Dynamic Tx Clock Switching** parameter, only one video standard is being tested with 2 different TX PHY reference clocks to demonstrate the switching:

- HD-SDI single-rate—HD
- 3G-SDI single-rate/triple-rate—3G Level A
- Multi-rate—12G 8 streams interleaved

**Figure 20. Simulation Waveform**



A successful simulation ends with the following message:

```

##### TRANSMIT TEST COMPLETED SUCCESSFULLY! #####
#
##### Channel 1: RECEIVE TEST COMPLETED SUCCESSFULLY! #####
  
```



## A Revision History for Intel FPGA SDI II Design Example User Guide for Intel Cyclone 10 GX Devices

---

Date	Version	Changes
December 2017	2017.12.25	Initial release.

---

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 or its subsidiaries in the U.S. and/or other countries. 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.

\*Other names and brands may be claimed as the property of others.

**ISO  
9001:2008  
Registered**