SDI II IP Core Design Example User Guide

Last updated for Quartus Prime Design Suite: 16.1
UG-SDI-II-DE
2016.10.31

Subscribe
Send Feedback

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

ALTERA® now part of Intel

Contents

SDI II Design Example Quick Start Guide

1

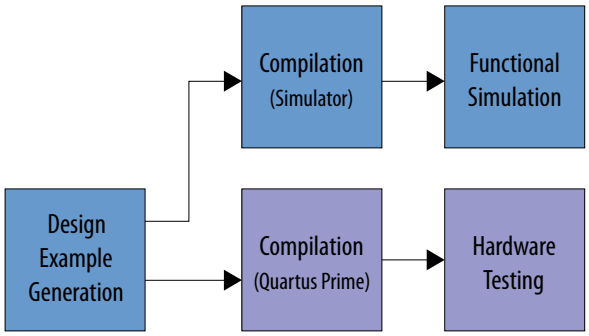
2016.10.31

UG-SDI-II-DE Subscribe Send Feedback

The SDI II IP core design examples for Arria® 10 devices feature a simulating 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 1-1: Development Steps



Related Information

SDI II IP Core User Guide

Directory Structure

The directories contain the generated files for the design examples.

Figure 1-2: Directory Structure for the Design Examples

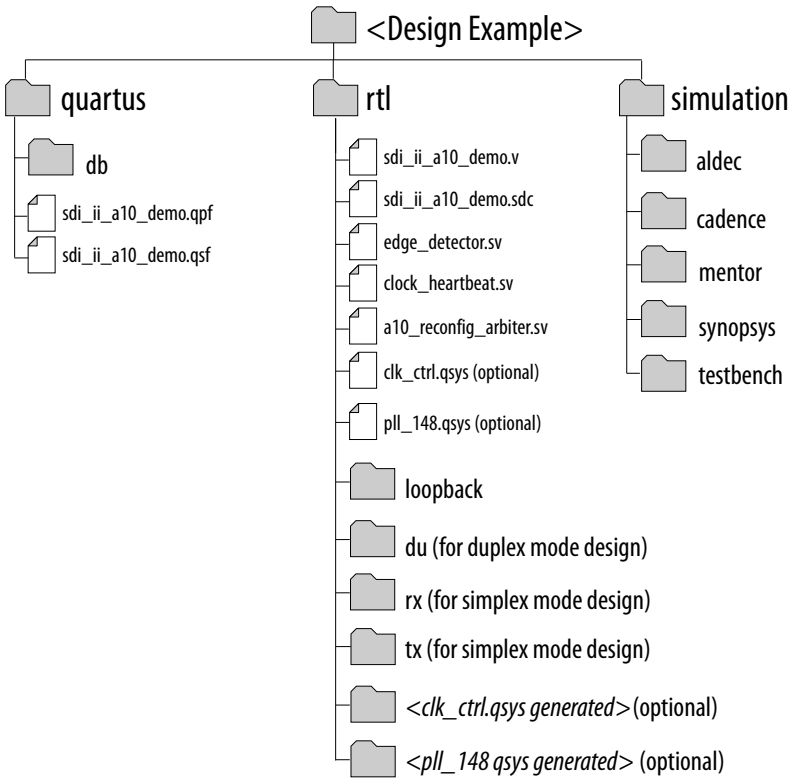


Table 1-1: Other Generated Files in RTL Folder

Folders	Files
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)

Folders	Files
du	/du_top.v
	/sdi_ii_rx_rcfg_a10.sv
	/rcfg_sdi_cdr.sv
	/sdi_du_sys.qsys
	/sdi_rx_phy.qsys
	/tx_pll.qsys
	<qsys generated folder>
rx	/rx_top.v
	/sdi_ii_rx_rcfg_a10.sv
	/rcfg_sdi_cdr.sv
	/sdi_rx_sys.qsys
	<qsys generated folder>
tx	/tx_top.v
	/sdi_tx_sys.qsys
	/tx_pll.qsys
	<qsys generated folder>
	/soft_lockppm_det/soft_lckppm_det

Table 1-2: 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

Folders	Files
synopsys	/vcs/ filelist.f
	/vcs/vcs_setup.sh
	/vcs/vcs_sim.sh
	/vcsmx/synopsys_sim_setup
	/vcsmx/vcsmx_setup.sh
	/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

Hardware and Software Requirements

Altera uses the following hardware and software to test the design examples:

Hardware

- Arria 10 GX FPGA Development Kit
- SDI Signal Generator
- SDI Signal Analyzer
- SubMiniature version B (SMB) to Bayonet Neill–Concelman (BNC) cables for triple rate design or BNC to BNC cables for multi rate design
- VIDIO™ FMC Development Module VIDIO-12G-A (Nextera 12G SDI FMC daughter card) for multi rate design

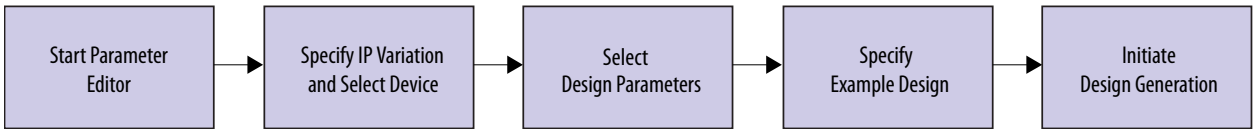
Software

- Quartus® Prime Standard Edition (for hardware testing)
- ModelSim-Altera Edition, ModelSim-Altera Starter Edition, NCSim (Verilog only), Riviera-Pro, or VCS (verilog only)/VCS-MX simulator

Generating the Design

Use the SDI II parameter editor in the Quartus Prime software to generate the design examples.

Figure 1-3: Generating the Design Flow



1. Click **Tools** > **IP Catalog**, and select Arria 10 as the target device family.

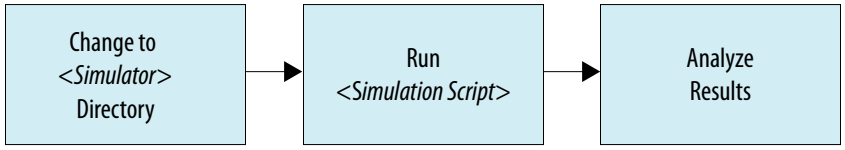
   Note: The design examples only support Arria 10 devices.
2. In the IP Catalog, locate and double-click **SDI II IP Core**. The **New IP Variation** window appears.
3. Specify a top-level name for your custom IP variation. The parameter editor saves the IP variation settings in *<your_ip>*`.qsys`. file
4. You may select a specific Arria 10 device in the **Device** field, or keep the default Quartus Prime software device selection.
5. Click **OK**. The parameter editor appears.
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 **Target Development Kit**, select **Arria 10 GX FPGA Development Kit**. If you select the development kit, then the target device (selected in **step 4**) changes to match the device on the development kit.
8. Click **Generate Example Design**.

Simulating the Design

The 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 1-4: 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.

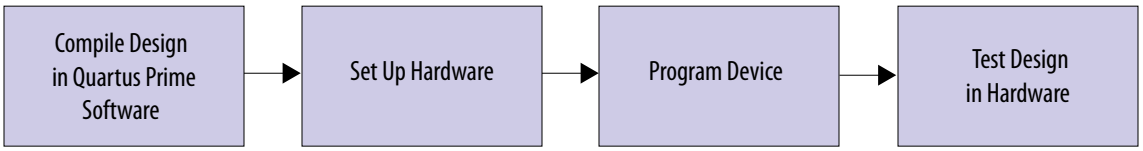
Table 1-3: Steps to Run Simulation

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

A successful simulation ends with the following message:

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

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. Launch the Quartus Prime Standard Edition software and open **quartus/sdi_ii_a10_demo.qpf**.
3. Click **Processing** > **Start Compilation**.
4. After successful compilation, the Quartus Prime Standard Edition software generates a `.sof` file in your specified directory.
5. Configure the selected Arria 10 device on the development board using the generated `.sof` file (**Tools** > **Programmer**).

Connection and Settings Guidelines

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

Connections and Settings for Triple Rate Design

- The on-board SMB RX connector (J20) connects to an external video source.
- The on-board SMB TX connector (J21) connects to 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: If you switch between non-integer frame rate and integer frame rate video formats, switch the Si516_FS (SW6.3) at the back of the board.

Figure 1-5: Switch Settings on the Arria 10 Development Board

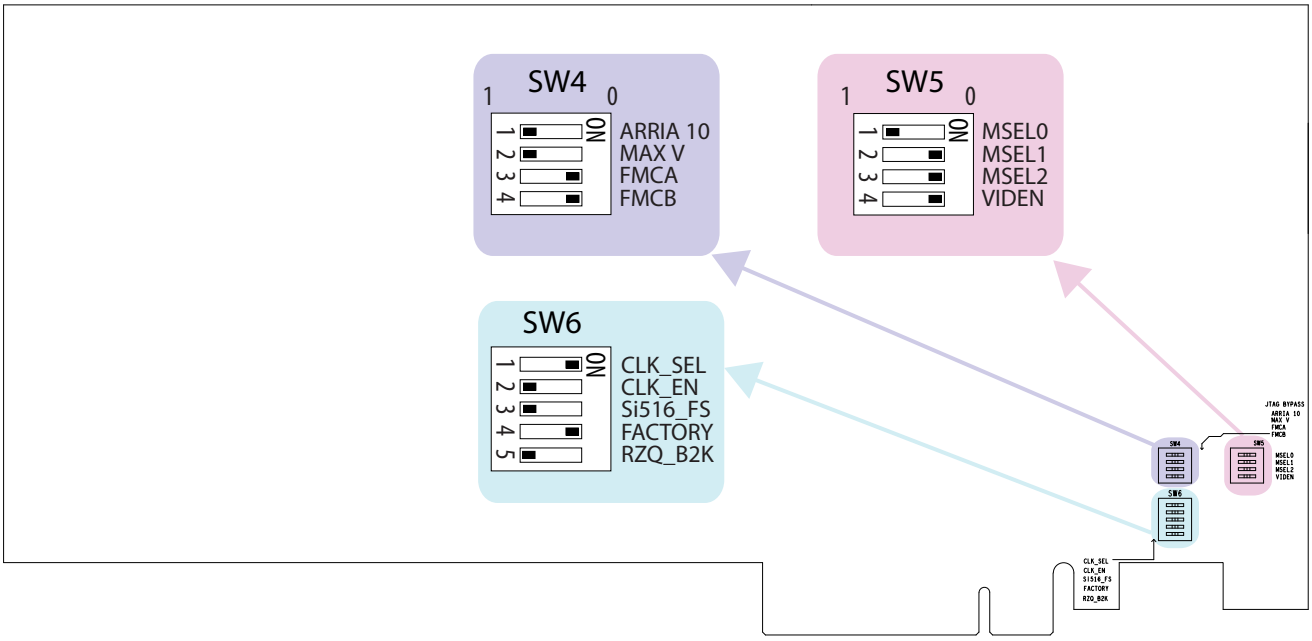


Table 1-4: SW6 DIP Switch Default Settings (Board Button)

Switch	Board Label	Description
1	CLK_SEL	• ON for 100 MHz on-board clock oscillator selection (Default position) • OFF for SMA input clock selection
2	CLK_EN	OFF for setting `CLK_ENABLE` high to the MAX V
3	SI516_FS	• ON for setting the `SDI REFCLK` frequency to 148.35 MHz • OFF for setting the `SDI REFCLK` frequency to 148.5 MHz (Default position)
4	FACTORY	• ON to load factory from flash (Default position) • OFF to load user hardware from flash
5	RZQ_B2K	• ON for setting RZQ resistor of Bank 2K to 99.17 ohm • OFF for setting RZQ resistor of Bank 2K to 240 ohm (Default position)

Connections and Settings for Multi Rate Design

- A VIDIO™ FMC Development Module VIDIO-12G-A (Nextera 12G SDI FMC daughter card) connects to the FMC Port B on the development board.
- The BNC Rx connector (J1/12G In) connects to an external video source.
- The TX connector (J2/12G Out) connects to 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: Change the jumper (J8) position before switching between non-integer frame rate and integer frame rate video formats. Press the push button (PB0) to trigger a device (LMK03328) power cycling through the PDN pin every time you change the jumper (J8) position.

Figure 1-6: Jumper Settings on Nextera 12G-SDI FMC Daughter Card

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

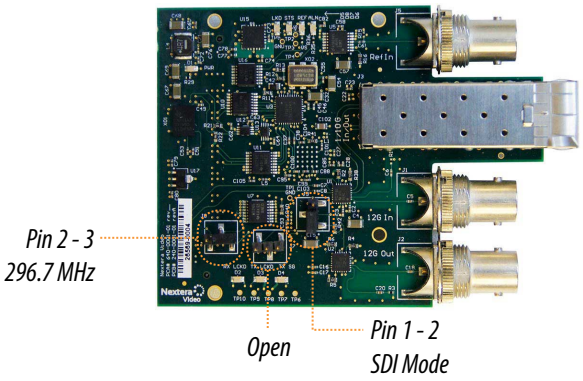


Table 1-5: Jumper Settings

Jumper Block	Description
J7	Programming header
J8	To switch the generated clock frequency for the TX channel: • Pin 1–2 = 297 MHz • Pin 2–3 = 297/1.001 MHz
J9	To select SDI or IP mode: • Pin 1–2 = SDI mode • Pin 2–3 = IP mode

Related Information

- **Arria 10 FPGA Development Kit User Guide**
- **Arria 10 FPGA Development Kit ES Edition User Guide**

SDI II Design Example Detailed Description

2

2016.10.31

UG-SDI-II-DE

The SDI II IP core includes three design examples for Arria 10 devices.

Table 2-1: Arria 10 SDI II Design Examples

Design Example	Data Rate	Channel Mode	VCXO Utilization	Loopback Type
Parallel loopback with external VCXO	Triple-rate (up to 3G-SDI)	Duplex	Yes	Parallel
Parallel loopback without external VCXO	Triple-rate (up to 3G-SDI)	Simplex	No	Parallel
Parallel loopback with external VCXO	Multi-rate (up to 12G-SDI)	Simplex	Yes	Parallel

Note: Designs that use voltage controlled crystal oscillator (VCXO) require an external VCXO, such as Si516. Designs without VCXO use the internal FPGA's fractional PLL.

Features

- All designs use LED status for early debugging stage.
- For triple rate parallel loopback with external VCXO design, you can choose either CMU or fPLL as the TX PLL.
- All designs include RX and TX options. To use RX or TX only components, you remove the irrelevant blocks from respective designs.

User Requirement	Preserve	Remove	Add
RX Only	RX Top	• TX Top • Loopback Top • Transceiver Arbiter	—
TX Only	TX Top	• RX Top • Loopback Top • Transceiver Arbiter	Video Pattern Generator (from simulation folder)

Parallel Loopback without External VCXO Triple-Rate Design Example

The parallel loopback without external VCXO design example demonstrates simplex channel mode for triple-rate SDI.

Figure 2-1: Parallel Loopback without External VCXO Triple-Rate Block Diagram

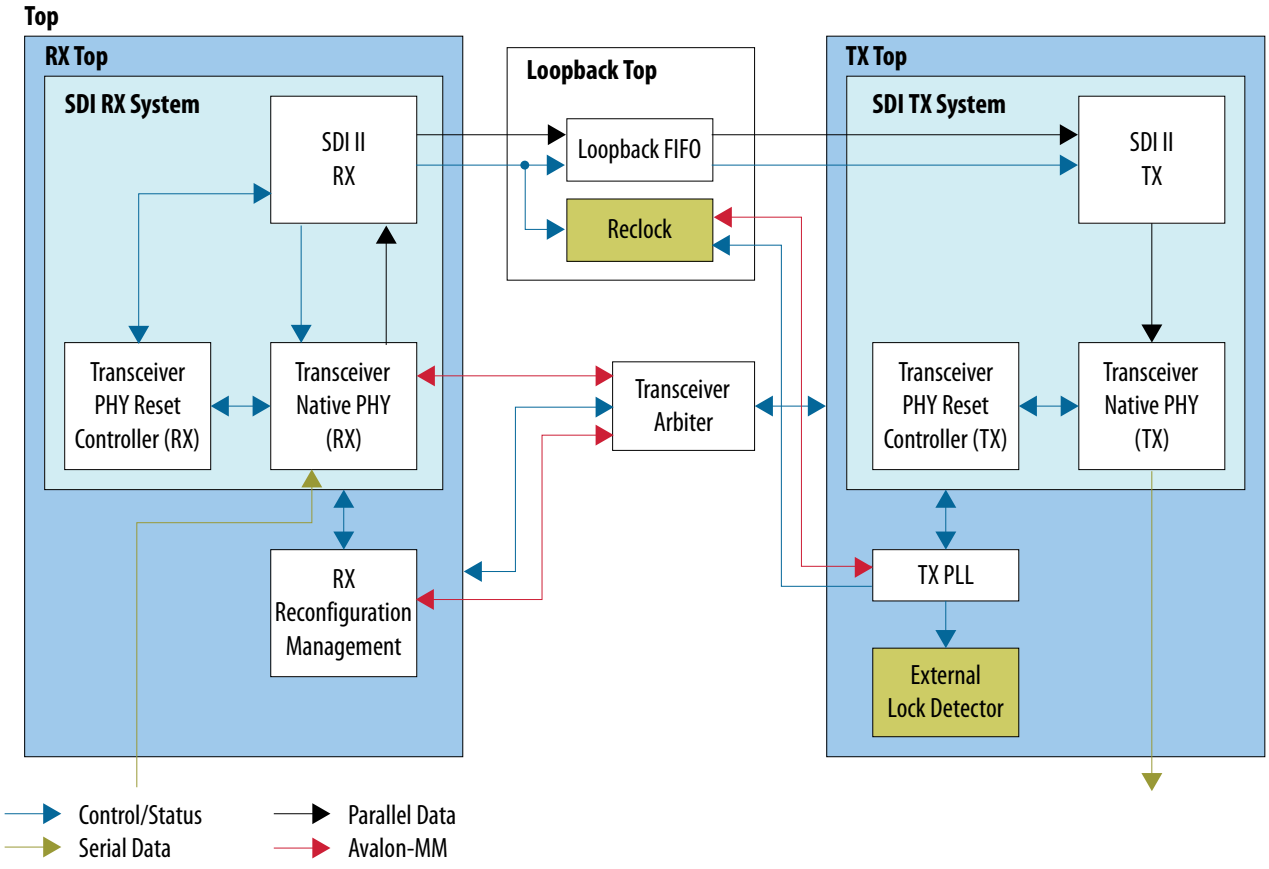
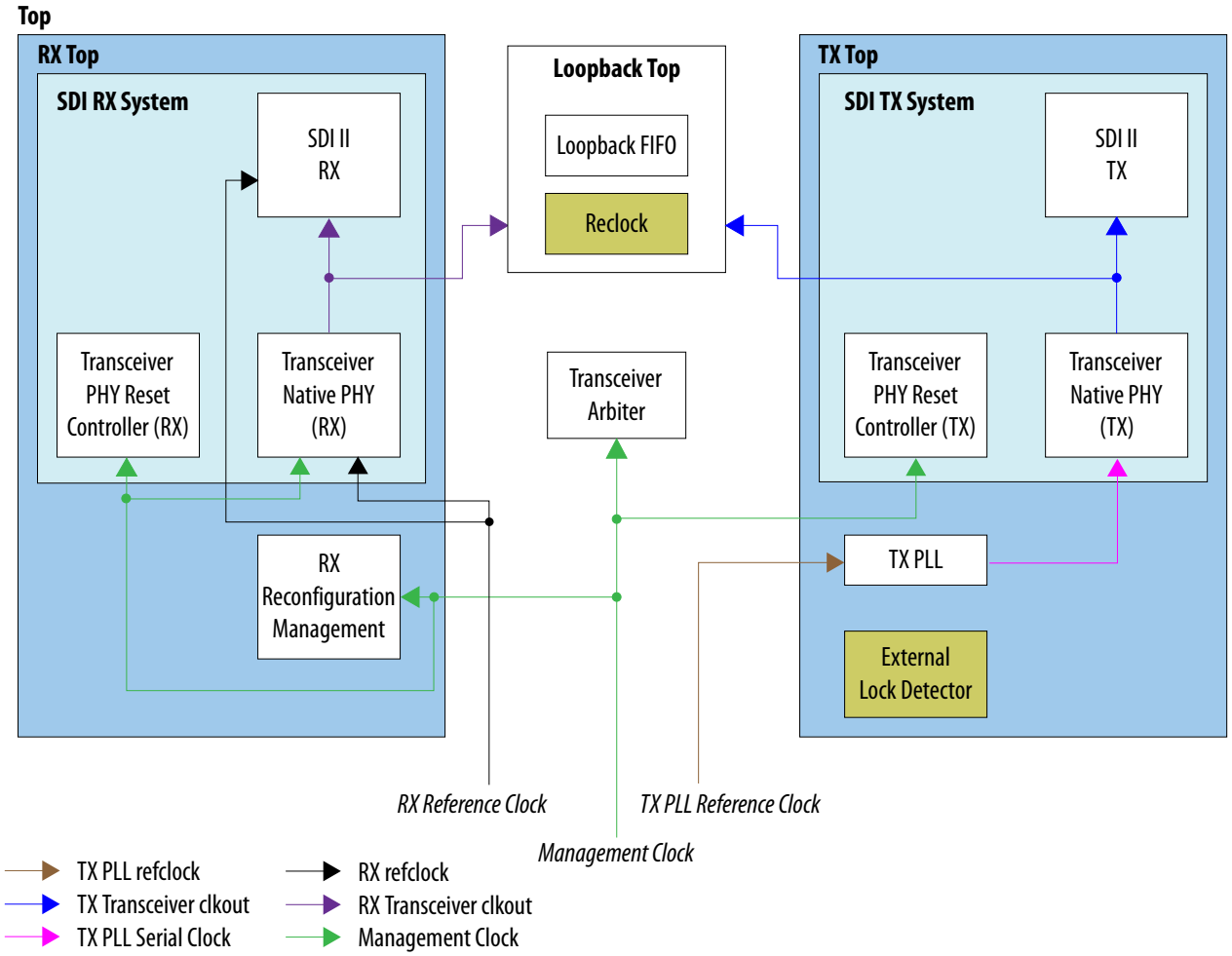


Figure 2-2: Parallel Loopback without External VCXO Triple-Rate Clock Scheme



Parallel Loopback with External VCXO Multi-Rate Design Example

The parallel loopback with external VCXO design example demonstrates simplex channel mode for multi-rate SDI.

Figure 2-3: Parallel Loopback with External VCXO Multi-Rate Block Diagram

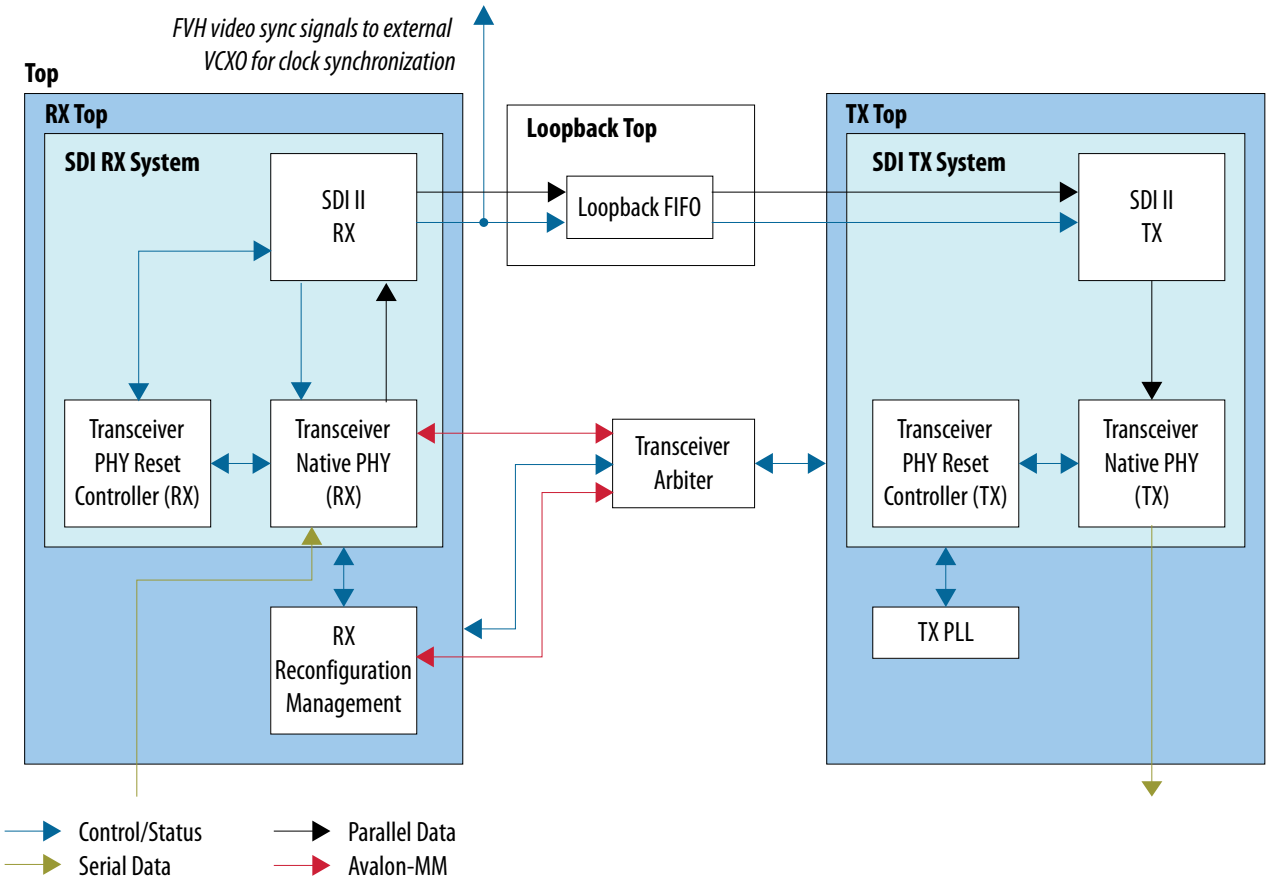
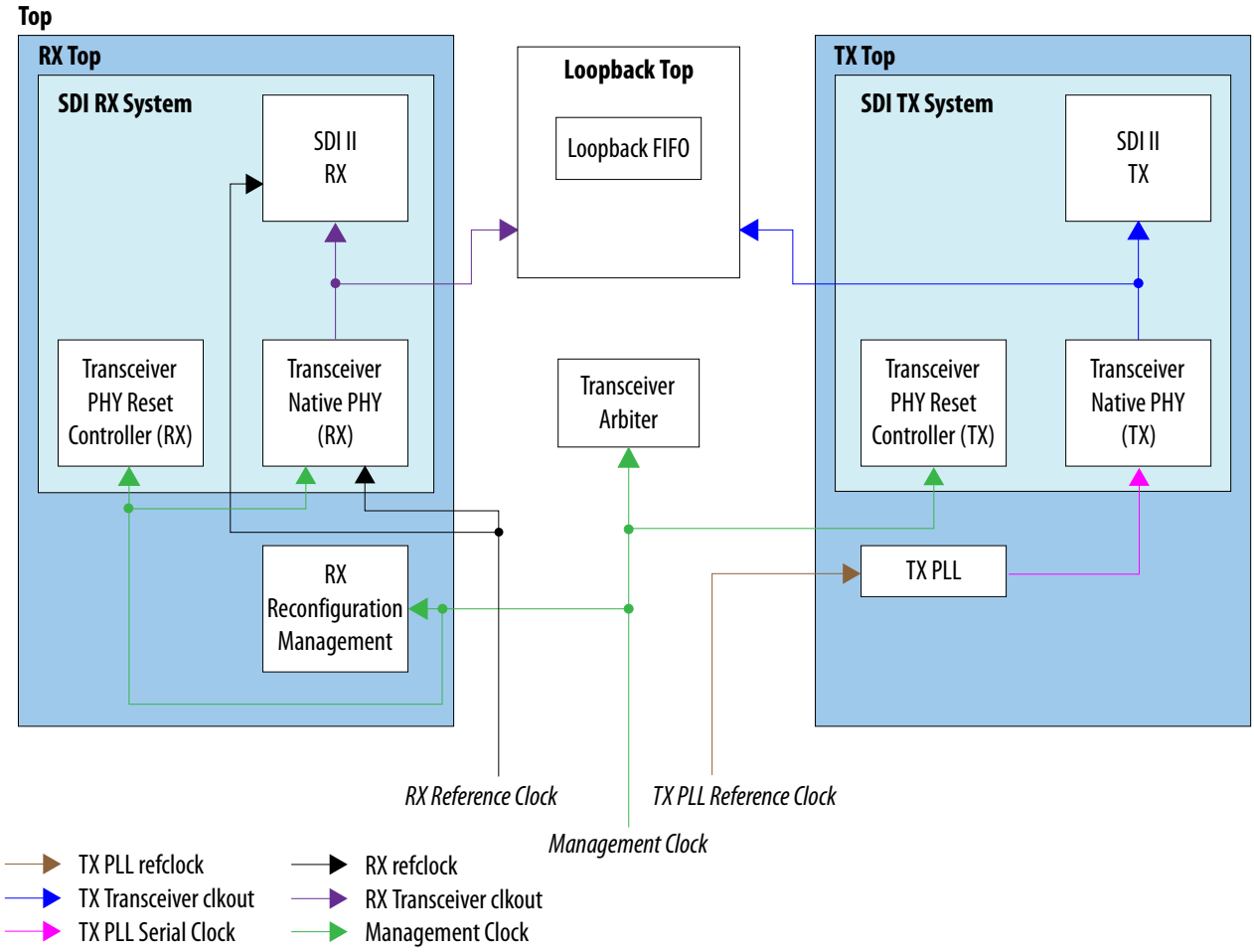


Figure 2-4: Parallel Loopback with External VCXO Multi-Rate Clock Scheme



Parallel Loopback with External VCXO Triple-Rate Design Example

The parallel loopback with external VCXO design example demonstrates duplex channel mode for triple-rate SDI.

Figure 2-5: Parallel Loopback with External VCXO Triple-Rate Block Diagram

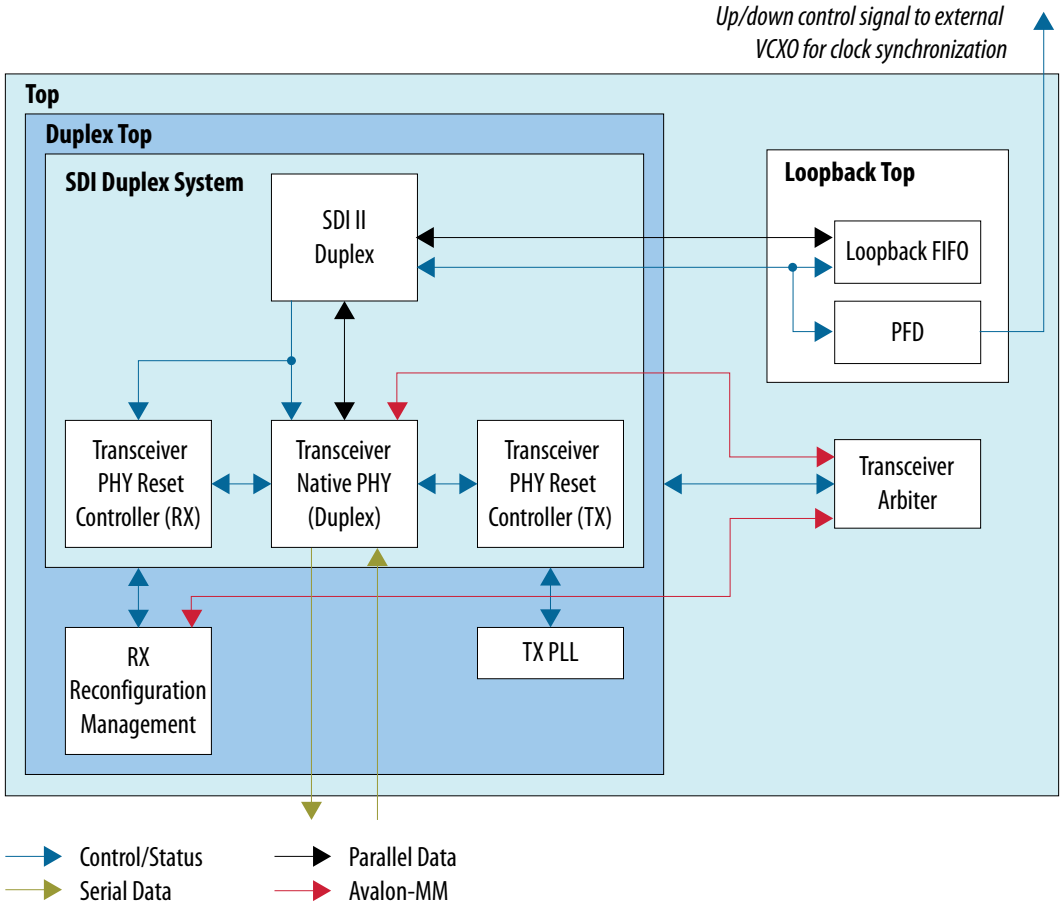
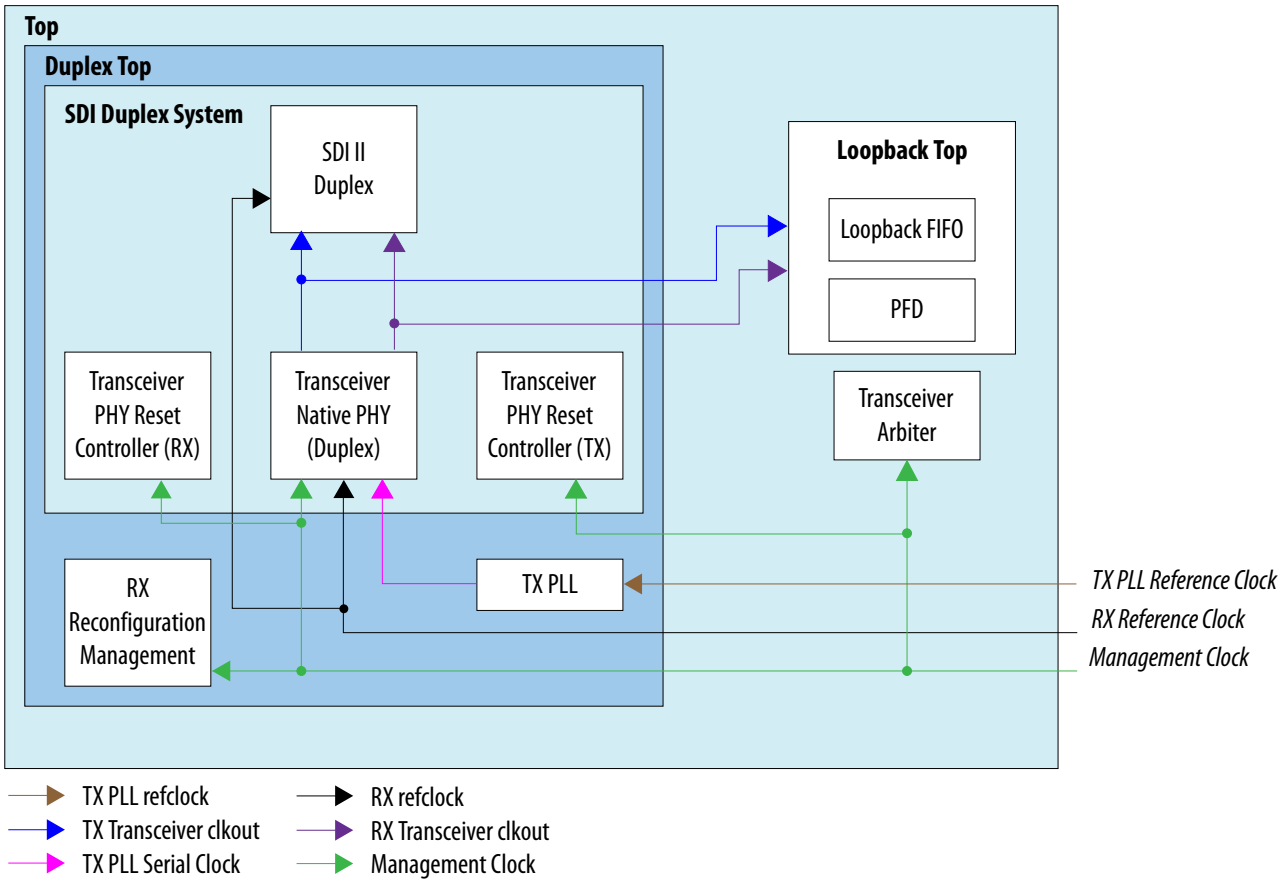


Figure 2-6: Parallel Loopback with External VCXO Triple-Rate Clock Scheme



Note: For duplex parallel loopback 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. For the parallel loopback with external VCXO triple-rate design example, use the only 148.5 MHz on-board oscillator as the TX PLL reference clock. For the RX reference clock, use a 270 MHz clock from another on-board oscillator.

Design Components

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

Table 2-2: Device Under Test (DUT) Components

Design Component	Description

SDI II IP Core	• TX
	  • 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).
	  • In a multi-rate design, the TX core oversamples the received data up to 11.88 Gbps data rate for every video standard. Specify the assignment of the parallel data interface (`tx_parallel_data`) to the transceiver based on the 11.88 Gbps data rate settings.
	• RX
	  • The RX core receives the parallel data from the Transceiver Native PHY IP core and decodes information. This information includes descrambling, realigning data, and extracting the necessary information for user.
	  • For a multi-rate design, due to the difference in data widths recovered for different video standards, rearrange `rx_parallel_data` from the transceiver before passing the data back to the protocol block.

Design Component	Description
Transceiver Native PHY IP Core	• TX Hard transceiver block that receives parallel data from the SDI II IP core and serializes the data before transmission. • For a triple-rate design, enable the simplified data interface option to connect parallel data directly to the `tx_dataout` signal of the SDI II IP core. • For a multi-rate design, disable this option due to the limitation in the 12G-SDI transceiver PHY settings. • RX Hard transceiver block that receives serial data from an external video source. • For a triple-rate design, enable the simplified data interface option to connect parallel data directly to the `rx_datain` signal of SDI II IP core. • For a multi-rate design, disable this option due to the limitation in the 12G-SDI transceiver PHY settings. You must connect the `rx_analogreset_ack` output signal from this block to the RX Reconfiguration Management module to indicate that the transceiver is in reset. **Note:** For the duplex mode transceiver (SDI triple-rate parallel loopback with external VCXO design example), generate a dummy RX only PHY (`sdi_rx_phy.qsys`) to get the transceiver configuration files (`*_CFG0.sv`, `*_CFG1.sv`, …) 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.
Transceiver PHY Reset TX	• TX • The reset input of this controller is triggered from the top level. • The controller generates the corresponding analog and digital reset signal to the Transceiver Native PHY block, according to the reset sequencing inside the block. • Use the `tx_ready` 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. • RX • The reset input of this controller is triggered by the SDI II IP core. • The controller generates the corresponding analog and digital reset signal to the Transceiver Native PHY block according to the reset sequencing inside the block.

Design Component	Description
RX Reconfiguration Management	RX transceiver reconfiguration management that reconfigures the Transceiver Native PHY block to receive different data rates from SD-SDI to 12G-SDI standards. To indicate the status of the transceiver, connect `rx_cal_busy` and `rx_analogreset_ack` from the transceiver to this block.
TX PLL	Transmitter PLL block that provides the serial fast clock to Transceiver Native PHY. • For triple rate design, choose CMU PLL or fPLL. • For multi rate design, use fPLL. Move the TX PLL out from the TX top if you want to merge the PLL between multiple channels.
External Lock Detector	• This block provides the TX PLL lock status signal when fPLL is operating in **SDI_direct** mode for the triple rate parallel loopback without external VCXO design. • When you set the protocol mode of the fPLL to **SDI_direct** and turn on fractional mode, the fPLL IP block does not provide the `pll_locked` status signal. • Include this block if any downstream logic requires the `pll_locked` status.

Table 2-3: 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. • The receiver sends the decoded RX data to the transmitter through this FIFO buffer. • When the receiver locks, the RX data is written to the FIFO buffer. • The transmitter starts reading, encoding, and transmitting the data when half of the FIFO buffer is filled.
Phase Frequency Detector (PFD)	You require this soft PFD block when you use the Arria 10 GX FPGA development kit on-board Si516 VCXO for a parallel loopback design. • This block compares the phase between the receiver and transmitter parallel clocks, and generates an up or down signal, that connects to the Si516 VCXO. • These up/down signals control the voltage of the VCXO, so that the frequencies of both clock domains can be tuned as close as possible to each other.

Component	Description
Reclock	The parallel loopback without external VCXO design requires this module. Similar to the PFD block, 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.

Table 2-4: Common Block

Component	Description
Transceiver Arbiter	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. 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. The transceiver arbiter identifies the requester of a reconfiguration through its Avalon-MM reconfiguration interfaces and ensures that the corresponding `tx_reconfig_cal_busy` or `rx_reconfig_cal_busy` is gated accordingly.

Clocking Scheme Signals

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

Table 2-5: Clocking Scheme Signals

<table>
<tr><th>Clock</th><th>Signal Name in Design</th><th>Description</th></tr>
<tr><td>TX PLL Refclock</td><td>tx_pll_refclk</td><td>TX PLL reference clock, of any frequency that is divisible by the transceiver for that data rate.

<table><tr><th>Design Example</th><th>Recommended Frequency</th></tr><tr><td>Multi rate parallel loopback with external VCXO</td><td>297 MHz</td></tr><tr><td>Triple rate parallel loopback with external VCXO</td><td>148.5 MHz</td></tr><tr><td>Triple rate parallel loopback without external VCXO</td><td>100 MHz</td></tr></table></td></tr>
<tr><td>TX Transceiver Clockout</td><td>tx_vid_clkout</td><td>148.5 MHz clock recovered from the transceiver.</td></tr>
<tr><td>TX PLL Serial Clock</td><td>tx_serial_clk</td><td>Serial fast clock generated by TX PLL. The clock frequency is set based on the data rate.</td></tr>
<tr><td rowspan="2">RX Refclock</td><td>rx_cdr_refclk</td><td>Transceiver clock data recovery (CDR) reference clock, of any frequency that is divisible by the transceiver for that data rate.

The recommended frequency is 148.5 MHz or higher. This clock must be a free-running clock.</td></tr>
<tr><td>rx_core_refclk</td><td>SDI RX core reference clock.

The recommended frequency is 148.5 MHz. This clock must be a free-running clock.</td></tr>
<tr><td>RX Transceiver Clkout</td><td>rx_vid_clkout</td><td>148.5 MHz clock recovered from the transceiver.</td></tr>
</table>

<table>
<tr><th>Clock</th><th>Signal Name in Design</th><th>Description</th></tr>
<tr><td rowspan="2">Management Clock</td><td>rx_rcfg_mgmt_clk</td><td>A free running 100 MHz RX clock for both Avalon-MM interfaces for reconfiguration and PHY reset controller for transceiver reset sequence.

<table><tr><th>Component</th><th>Required Frequency (MHz)</th></tr><tr><td>Avalon-MM reconfiguration</td><td>100 – 125</td></tr><tr><td>Transceiver PHY reset controller</td><td>1 – 500</td></tr></table>
You may also connect this input clock to rx_core_refclk and change the input clock frequency option to 149 MHz in the parameter editor.</td></tr>
<tr><td>tx_rcfg_mgmt_clk</td><td>A free-running 100 MHz TX clock for both Avalon-MM interfaces for reconfiguration and PHY reset controller for transceiver reset sequence.

<table><tr><th>Component</th><th>Required Frequency</th></tr><tr><td>Avalon-MM reconfiguration</td><td>100 – 125 MHz</td></tr><tr><td>Transceiver PHY reset controller</td><td>1 – 500 MHz</td></tr></table>
You may also connect this input clock to tx_pll_refclk (assuming this clock is not connected to a VCXO that will be tuned) and change the input clock frequency option to the correct clock frequency in the parameter editor.</td></tr>
</table>

Interface Signals

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

Table 2-6: Top-Level Signals

Signal	Direction	Width	Description
On-board Oscillator Signals			
clk_fpga_b2_p	Input	1	100 MHz clock for reconfiguration Avalon-MM interfaces.
pcie_ob_refclk_p	Input	1	100 MHz dedicated transceiver reference clock.
refclk_dp_p	Input	1	270 MHz dedicated transceiver reference clock.
refclk_sdi_p	Input	1	148.5 or 148.35 MHz dedicated transceiver reference clock.
User Push Buttons and LEDs			
user_pb0	Input	1	Push button to power down LMK03328 after switching the jumper settings.
cpu_resetn	Input	1	Global reset.
user_led_g	Output	8	Green LED display.
user_led_r	Output	8	Red LED display.
On-board Si516, SDI Cable Driver and Equalizer Related Pins			
sdi_rx_p	Input	1	On-board SDI RX serial data.
sdi_tx_p	Output	1	On-board SDI TX serial data.
sdi_clk148_up	Output	1	Voltage control for Si516.
sdi_clk148_down	Output	1	Voltage control for Si516.
sdi_mf0_bypass	Output	1	On-board SDI RX Equalizer Bypass.
sdi_mf1_auto_sleep	Output	1	On-board SDI RX Equalizer Auto Sleep.
sdi_mf1_mute	Output	1	On-board SDI RX Equalizer Mute.
sdi_tx_sd_hdn	Output	1	On-board SDI TX cable driver slew rate control.
Nextera SDI FMC Daughter Card Pins on FMC Port B			
fmcb_gbtclk_m2c_p0	Input	1	297 or 296.7 MHz dedicated transceiver reference clock from FMC port B.
fmcb_dp_m2c_p2	Input	1	SDI RX serial data from FMC port B.

Nextera SDI FMC Daughter Card Pins on FMC Port B			
fmcb_la_tx_p1	Input	1	RX cable equalizer lock status on Nextera daughter card.
fmcb_dp_c2m_p0	Output	1	SDI TX serial data from FMC port B.
fmcb_la_tx_p12	Output	1	Initialize LMH1983 on Nextera daughter card.
fmcb_la_tx_n12	Output	1	F sync signal LMH1983 on Nextera daughter card.
fmcb_la_tx_p14	Output	1	V sync signal LMH1983 on Nextera daughter card.
fmcb_la_tx_n14	Output	1	H sync signal LMH1983 on Nextera daughter card.
fmcb_la_tx_p15	Output	1	Power-down signal LMH1983 on Nextera daughter card.

Table 2-7: RX/TX/DU Top Signals

Signal	Direction	Width	Description
Clocks			
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.
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.

Reset			
tx_resetn	Input	1	TX core and PHY reset signal.
rx_resetn	Input	1	RX core and PHY reset signal.
tx_rcfg_mgmt_resetn	Input	1	TX reconfiguration reset signal.
rx_rcfg_mgmt_resetn	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.
Video Signal Interfaces (Interface with Video Image and Processing (VIP) Components)			
rx_vid_data	Output	20*N	Receiver parallel video data out. **Note:** *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: • SD-SDI: 1H 4L 1H 5L • HD-SDI: 1H 1L • 3G/6G/12G-SDI: H
rx_vid_std	Output	3	Received video standard. • 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-SDI16 Streams Interleaved
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. **Note:** *N* = 4 (multi-rate design) or 1 (triple-rate design)

Video Signal Interfaces (Interface with Video Image and Processing (VIP) Components)			
rx_vid_vsync	Output	*N*	Vertical blanking interval timing signal. The receiver asserts this signal when the vertical blanking interval is active. **Note:** *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 interfaced frame, 0 means first field (F0) while 1 means second field (F1). For progressive frame, the value is always 0. **Note:** *N* = 4 (multi-rate design) or 1 (triple-rate design)
rx_vid_trs	Output	*N*	On-board SDI TX cable driver slew rate control. **Note:** *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. **Note:** *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: • SD-SDI = 1H 4L 1H 5L • HD-SDI = H • 3G/6G/12G-SDI = H
tx_vid_std	Input	3	Indicates the desired transmit video standard. • 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-SDI16 Streams Interleaved

Video Signal Interfaces (Interface with Video Image and Processing (VIP) Components)			
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. **Note:** *N* = 4 (multi-rate design) or 1 (triple-rate design)
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. **Note:** *N* = 4 (multi-rate design) or 1 (triple-rate design)
sdi_rx_coreclk_is_ntsc_paln	Input	1	To indicate whether rx_coreclk is 148.5 MHz or 148.35 MHz: • 0: 148.5 MHz • 1: 148.35 MHz
sdi_tx_datavalid	Output	1	Data valid signal generated from SDI TX core. The timing (H: High, L: Low) must be synchronous to tx_vid_clkout and has the following settings: • SD-SDI = 1H 4L 1H 5L • HD-SDI = H • 3G/6G/12G-SDI = H
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. **Note:** *N* = 4 (multi-rate design) or 1 (triple-rate design)

Other SDI Video Protocol Interfaces			
sdi_rx_clkout_is_ntsc_paln	Output	1	Indicates that the receiver is receiving video rate at integer or non-integer frame rate: • 0: Integer frame rate • 1: Non-integer frame rate
sdi_rx_format	Output	4*N	Received video transport format. Refer to the *SDI II IP User Guide* for the encoding value. **Note:** N = 4 (multi-rate design) or 1 (triple-rate design)
sdi_rx_crc_error	Output	1	CRC error status signal from protocol.
Transceiver Interfaces			
tx_pll_select	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.
cdr_reconfig_busy	Output	1	RX CDR reconfiguration status.

Transceiver Reconfiguration Interfaces			
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	
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_fpll_rcfg_write	Output	1	Reconfiguration interface signals to fPLL Avalon-MM interface.
tx_fpll_rcfg_read	Output	1	
tx_fpll_rcfg_writedata	Output	32	
tx_fpll_rcfg_address	Input	10	
tx_fpll_rcfg_readdata	Output	32	
tx_fpll_rcfg_waitrequest	Output	1	

Table 2-8: Loopback Top Signals

Signal	Direction	Width	Description
Clocks			
sdi_tx_clkout	Input	1	TX transceiver recovered parallel clock for video data.

Signal	Direction	Width	Description
Clocks			
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. **Note:** *N* = 4 (multi-rate design) or 1 (triple-rate design)
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. **Note:** *N* = 4 (multi-rate design) or 1 (triple-rate design)
sdi_rx_trs_locked	Input	*N*	TRS locked status signal from SDI RX core. **Note:** *N* = 4 (multi-rate design) or 1 (triple-rate design)
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 Related Signals			
sdi_rx_clkout_is_ntsc_paln	Input	1	Indication from SDI RX core that the receiver is receiving video rate at integer or non-integer frame rate.
sdi_tx_datain	Output	20*N	Parallel video data input to SDI TX core. **Note:** N = 4 (multi-rate design) or 1 (triple-rate design)
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.
Voltage Control Signals for On-board Si516			
vcoclk_up	Output	1	Voltage up signal to Si516 to increase the voltage.
vcoclk_down	Output	1	Voltage down signal to Si516 to decrease the voltage.
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 2-9: Transceiver Arbiter Signals

Signal	Direction	Width	Description
On-board Oscillator Signals			
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_waitrequest	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_waitrequest	Output	1	

Signal	Direction	Width	Description
On-board Oscillator Signals			
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	
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.

Hardware Setup

The design examples demonstrate one channel parallel loopback with the RX instance expecting a video source from an external video generator. The video data then transmits through a loopback FIFO before passing to the TX instance.

To run the hardware test, 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.

Figure 2-7: Arria 10 GX Development Board User LEDs

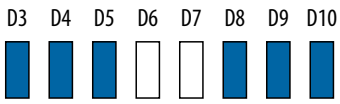


Table 2-10: On-board User LED Functions

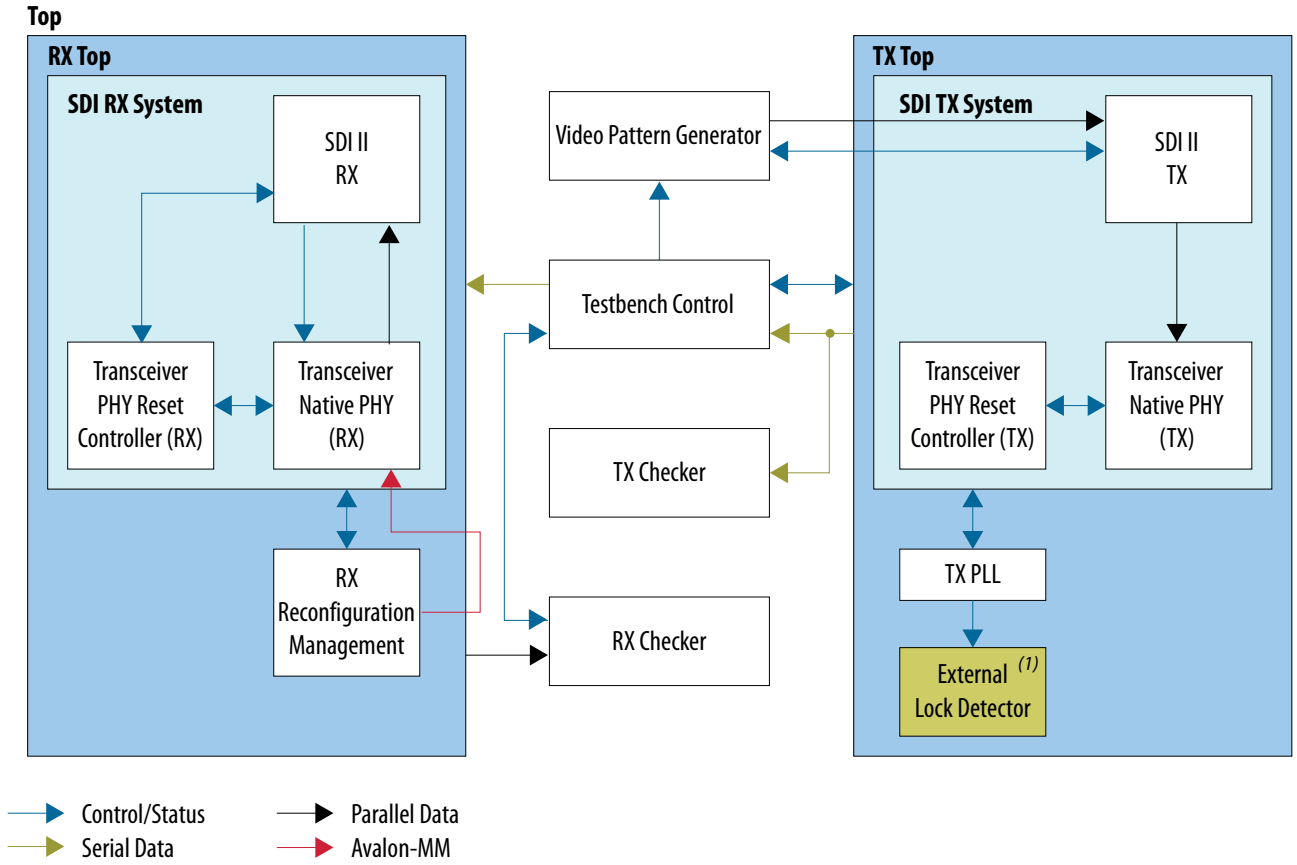
LEDs	Function
D3–D5	Indicates the receiver video standard: • 000: SD-SDI • 001: HD-SDI • 010: 3G Level B • 011: 3G Level A • 100: 6G 8 Streams Interleaved • 101: 6G 4 Streams Interleaved • 110: 12G 16 Streams Interleaved • 111: 12G 8 Streams Interleaved
D6	Shows the slower version of the TX transceiver parallel clock
D7	Shows the slower version of the RX transceiver parallel clock
D8	Illuminates when `align_locked` asserts
D9	Illuminates when `trs_locked` asserts
D10	Illuminates when `frame_locked` asserts

After verifying that the RX core works correctly, connect an SDI signal analyzer to the transmitter output. The same image,which the source generates, should display on the signal analyzer.

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 2-8: Simplex Mode Simulation Testbench Block Diagram



Note (1): This block is only available when you select the Parallel Loopback without External VCXO design.

Figure 2-9: Duplex Mode Simulation Testbench Block Diagram

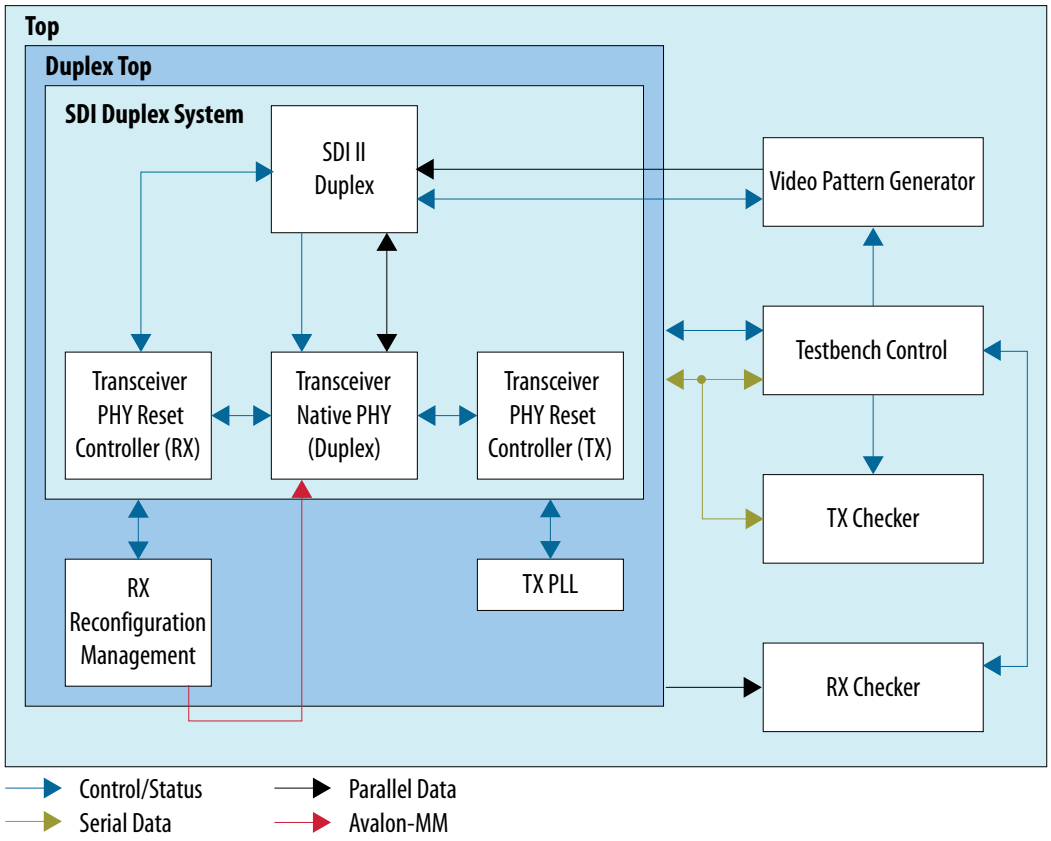


Table 2-11: Testbench Components

Component	Description
Video Pattern Generator	This generator produces configurable color bar or pathological pattern. The generator supports video formats SD-SDI up to 3G-SDI. For 6G/12G-SDI, you can cascade the output of this block to form multiple streams (2 streams for 6G-SDI and 4 streams for 12G-SDI).
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 `trs_locked` 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 2-10: Sequence of Video Standards for Triple-Rate and Multi-Rate Designs

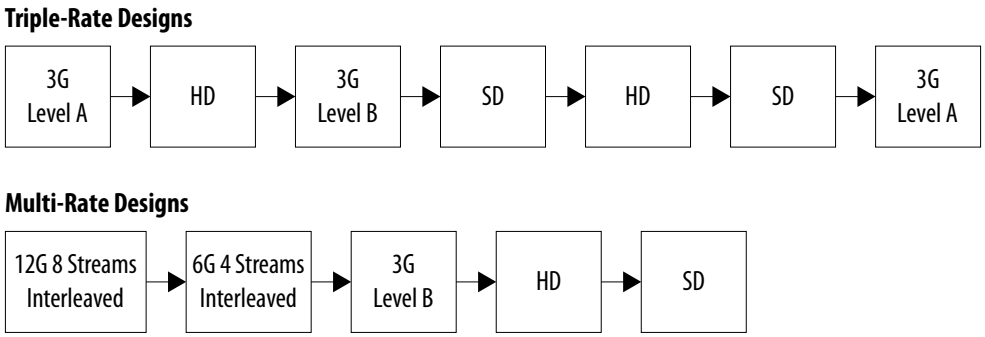
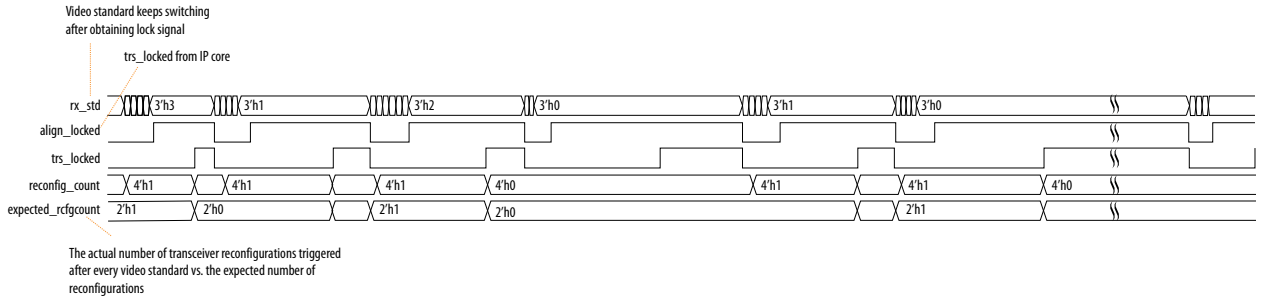


Figure 2-11: Simulation Waveform



A successful simulation ends with the following message:

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

Revision History for SDI II IP Core Design Example User Guide

A

2016.10.31

UG-SDI-II-DE

Subscribe

Send Feedback

Date	Version	Changes
October 2016	2016.10.31	Initial release.

101 Innovation Drive, San Jose, CA 95134