XAPP1076 (v1.0) December 15, 2010

Implementing Triple-Rate SDI with Spartan-6 FPGA GTP Transceivers

Author: Reed Tidwell

Summary

The triple-rate serial digital interface (SDI) supporting the SMPTE SD-SDI, HD-SDI, and 3G-SDI standards is widely used in professional broadcast video equipment. SDI interfaces are used in broadcast studios and video production centers to carry uncompressed digital video, along with embedded ancillary data, such as multiple audio channels.

Spartan®-6 FPGA GTP transceivers are well-suited for implementing triple-rate SDI receivers and transmitters. This document describes how to implement triple-rate SDI interfaces using Spartan-6 FPGAs.

Introduction

The Spartan-6 FPGA triple-rate SDI reference design supports SD-SDI, HD-SDI, and 3G-SDI (both level A and level B). If an application only requires support for a subset of these SDI standards, the triple-rate SDI interface is still used. There is little overhead in the basic triple-rate SDI datapath for the unused SDI standards.

Differences from Virtex-5 FPGA GTP Transceivers

The GTP transceivers in Spartan-6 devices are similar to the GTP transceivers in Virtex®-5 LXT devices. The differences important to SDI interfaces are described in this section.

Virtex-5 FPGA GTP transceivers in come in GTP_DUAL tiles, with one PMA PLL (used for reference clock multiplication) shared by the two transmitters and two receivers in the tile. The shared PMA PLL limits the independence of the receivers and transmitters in the tile. Spartan-6 FPGA GTP transceivers also come in GTP_DUAL tiles; however, Spartan-6 FPGA GTP transceivers have a separate PMA PLL available for each lane (receiver/transmitter pair). This provides two independent clocks that are shared by the transmitters and receivers in the tile. This allows more flexibility in using a tile for both transmitting and receiving; however, it is still not recommended to use a transmitter and receiver with a common clock, unless the reference clock does not change. The receivers should use a different clock from the transmitters because if the transmitter clock changes, the receiver stream incurs errors. It is possible, in Spartan-6 devices, to have transmitters in the same tile transmitting at different bit rates, something that was impossible in Virtex-5 devices because of the single PMA PLL.

The clock recovered by a Virtex-5 FPGA GTP receiver stops if the input serial bitstream stops. This situation often requires using a BUFGMUX to substitute a free-running clock for the recovered clock when it stops, because many video applications cannot tolerate a stoppage in the recovered video clock. The recovered clocks from Spartan-6 FPGA GTP receivers do not stop when the input bitstream stops. This eliminates the need to use BUFGMUXes to swap in a free-running clock.

In GTP_DUAL tiles of Virtex-5 FPGA GTP transceivers, the reference clock provided to the PMA PLL could be changed dynamically, but only through the DRP port. The reference clock multiplexer for each PMA PLL in the Spartan-6 FPGA GTP transceiver is dynamically controlled by reference clock selection ports, rather than through the DRP.

The Virtex-5 FPGA GTP receiver requires a clock and data recovery (CDR) reset when the input bitstream stops, such as when the SDI cable is unplugged. The preferred method to

generate this reset in SDI interfaces is to use the carrier detect signal from the SDI cable equalizer. The Spartan-6 FPGA GTP receiver does not require the carrier detect signal from the SDI cable equalizer to generate a CDR reset when the cable is unplugged.

In Virtex-5 FPGA GTP transceivers, the fundamental user clock frequency is at a two word per clock rate. For SDI, this means the user clock rate is 1/20th the serial bit rate. In the Spartan-6 FPGA GTP transceiver, the user clock frequency is one word per clock. For SDI, that means 1/10th the serial bit rate. The GTP transceiver uses a 20-bit parallel interface. Thus, two clocks must be provided to the GTP transceiver: USRCLK at 1/10th the serial bit rate, and USRCLK2 at 1/20th the serial bit rate.

The slowest speed grade, -2, of Spartan-6 devices does not support the 20-bit interface at the speed required for 3G-SDI and oversampled SD-SDI. Therefore, the reference design described in this application note can only be implemented on speed grades of -3 or faster.

Supported Video Formats

The triple-rate SDI receiver and transmitter support the video formats shown in Table 1. The receiver and transmitter do not convert between video and SDI data streams for all video formats. The formats that require conversion between SDI data streams and video using external formatting modules are marked as *External* in the SDI Stream Mapping column. Those that are directly supported are marked with Not required in the SDI Stream Mapping column.

Table 1: **Supported Video Formats**

Interface	Video Standard	Sampling Structure/ Bit Depth	Frame/Field Rate (Hz)	SDI Stream Mapping
SD-SDI SMPTE 259-C	PAL	4:2:2 Y'Cb'Cr' 10-bit or 8-bit	50	Not required
	NTSC	4:2:2 Y'Cb'Cr' 10-bit or 8-bit	59.94	Not required
HD-SDI SMPTE 292	SMPTE 274	4:2:2 Y'Cb'Cr' 10-bit	1080p: 23.98, 24, 25, 29.97, 30 1080i: 50, 59.94, 60 1080PsF: 23.98, 24, 25, 29.97, 30	Not required
	SMPTE 296	4:2:2 Y'Cb'Cr' 10-bit	720p: 23.98, 24, 25. 29.97, 30, 50, 59.94, 60	Not required
	SMPTE 260	4:2:2 Y'Cb'Cr' 10-bit	1035i: 59.94, 60	Not required
	SMPTE 295	4:2:2 Y'Cb'Cr' 10-bit	1080i: 50	Not required
3G-SDI Level A SMPTE 425-A	SMPTE 274	4:2:2 Y'Cb'Cr' 10-bit	1080p: 50, 59.94, 60	Not required
		4:4:4 or 4:4:4:4 Y'Cb'Cr' or RGB 10-bit	1080p: 23.98, 24, 25, 29.97, 30 1080i: 50, 59.94, 60 1080PsF: 23.98, 24, 25, 29.97, 30	External
		4:4:4 Y'Cb'Cr' or RGB 12-bit	1080p: 23.98, 24, 25, 29.97, 30 1080i: 50, 59.94, 60 1080PsF: 23.98, 24, 25, 29.97, 30	External
		4:2:2 Y'Cb'Cr' 12-bit	1080p: 23.98, 24, 25, 29.97, 30 1080i: 50, 59.94, 60 1080PsF: 23.98, 24, 25, 29.97, 30	External
	SMPTE 296	4:4:4 or 4:4:4:4 Y'Cb'Cr' or RGB 10-bit	720p: 23.98, 24, 25. 29.97, 30, 50, 59.94, 60	External
	SMPTE 428-9	4:4:4 X'Y'Z' 12-bit	2048 x 1080p: 24	External
3G-SDI Level B SMPTE 425-B	SMPTE 372	See Dual-Link HD-SDI SMPTE 372 in this table.		
	2 X HD-SDI streams	See HD-SDI SMPTE 292 in this table.		
Dual-Link HD-SDI SMPTE 372	SMPTE 274	4:2:2 Y'Cb'Cr' 10-bit	1080p: 50, 59.94, 60	External
		4:4:4 or 4:4:4:4 Y'Cb'Cr' or RGB 10-bit	1080p: 23.98, 24, 25, 29.97, 30 1080i: 50, 59.94, 60 1080PsF: 23.98, 24, 25, 29.97, 30	External
		4:4:4 Y'Cb'Cr' or RGB 12-bit	1080p: 23.98, 24, 25, 29.97, 30 1080i: 50, 59.94, 60 1080PsF: 23.98, 24, 25, 29.97, 30	External
		4:2:2 Y'Cb'Cr' 12-bit	1080p: 23.98, 24, 25, 29.97, 30 1080i: 50, 59.94, 60 1080PsF: 23.98, 24, 25, 29.97, 30	External
	SMPTE 428-9	4:4:4 X'Y'Z' 12-bit	2048 x 1080p: 24	External

Triple-Rate SDI Receiver Reference Design

A triple-rate SDI receiver is implemented by an `s6_sdi_rx_light_20b` module, combined with a Spartan-6 FPGA GTP receiver and an `s6gtp_sdi_control` module, as shown in Figure 1. Optionally, an EDH processor can be included to provide SD-SDI error checking and lock-detection features.

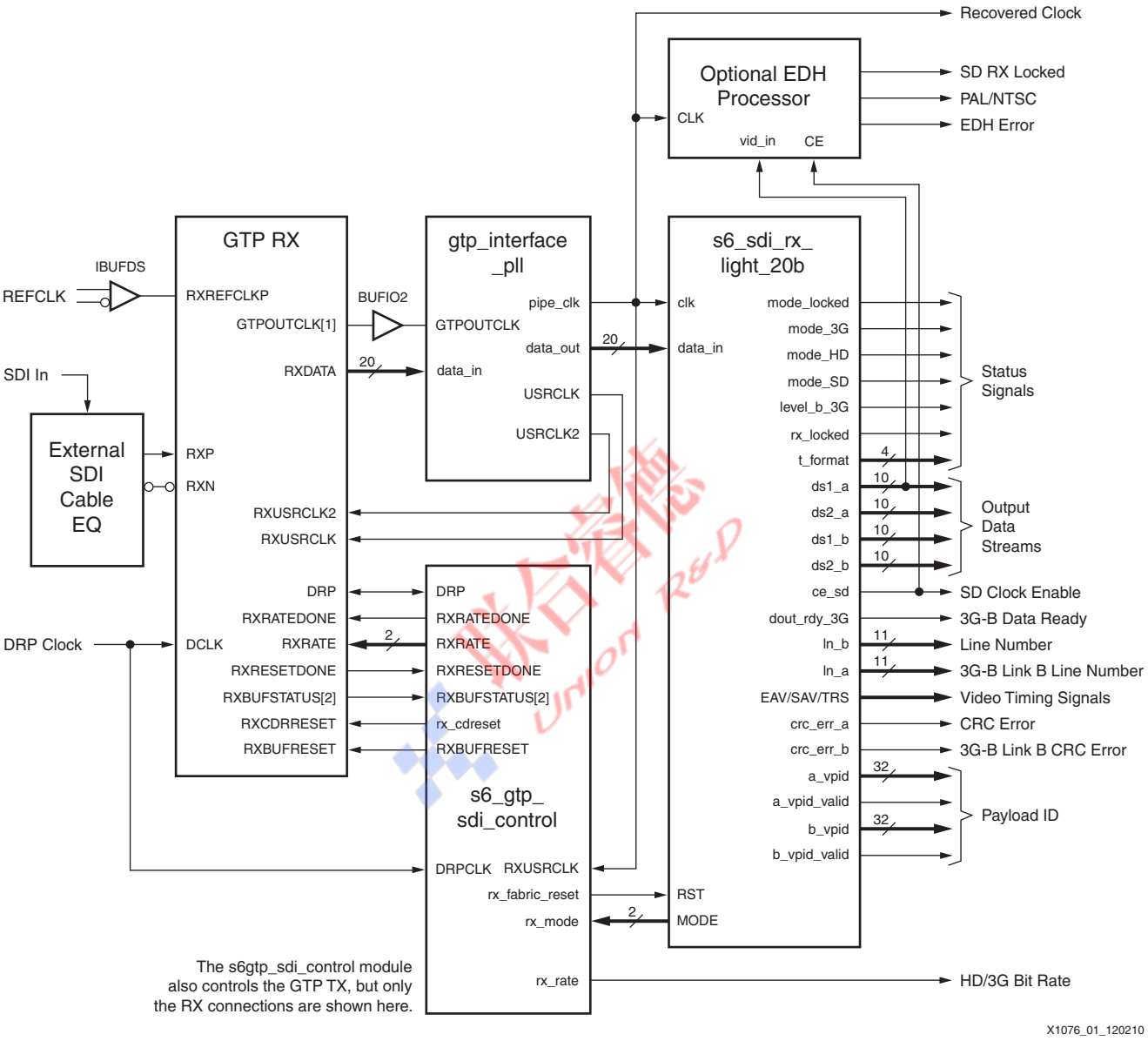


Figure 1: **Spartan-6 FPGA GTP Transceiver Triple-Rate SDI Receiver Block Diagram**

The triple-rate SDI receiver has these features:

- Uses a 20-bit GTP transceiver interface, supported on -3 and faster Spartan-6 devices
- A single reference clock frequency is required to receive the five supported bit rates:
  - 270 Mb/s SD-SDI (SMPTE 259)
  - 1.485 Gb/s HD-SDI (SMPTE 292)
  - 148.5/1.001 Gb/s HD-SDI (SMPTE 292)
  - 2.97 Gb/s 3G-SDI (SMPTE 424)
  - 2.97/1.001 Gb/s 3G-SDI (SMPTE 424)

- The receiver has a bit rate detector that can distinguish between the two HD-SDI and two 3G-SDI bit rates. An output signal from the receiver indicates which bit rate is being received.
- The receiver automatically detects the SDI standard of the input signal (3G-SDI, HD-SDI, or SD-SDI) and reports the current SDI mode on an output port.
- The receiver detects and reports the video transport format (e.g., 1080p 30 Hz, 1080i 50 Hz).
- The receiver supports both 3G-SDI level A and level B formats, and automatically detects whether 3G-SDI data streams are level A or B. The 3G-SDI level is reported on an output port.
- The receiver performs CRC error checking for HD-SDI and 3G-SDI modes.
- Optional EDH error checking for SD-SDI is available.
- Line numbers are captured and output from the triple-rate SDI receiver. For 3G-SDI level B format, line numbers are captured for each of the two HD-SDI streams carried on the 3G-SDI interface.
- SMPTE 352 video payload ID packets are captured for all SDI modes. All captured SMPTE 352 packet data is available on output ports for one or two streams (for those formats that require SMPTE 352 packets in both streams).
- The pipeline clock and the GTP transceiver user clocks are created in the GTP interface module.
- All ancillary data embedded in the SDI data streams is preserved and output from the receiver on the video data stream outputs.

Table 2 describes the I/O ports of the `s6_sdi_rx_light_20b` module.

Table 2: **I/O Ports of the `s6_sdi_rx_light_20b` Module**

Port Name	I/O	Width	Description
clk	In	1	The user connects this clock input to the pipe_clk output of the `gtp_interface_pll` module for the GTP RX. The clock frequency is 148.5 MHz (or 148.5/1.001 MHz) for 3G-SDI, and 74.25 MHz (or 74.25/1.001 MHz) for HD-SDI and SD-SDI.
rst	In	1	This is an asserted High asynchronous reset input. The falling edge of this reset signal must meet the reset recovery time of all flip-flops relative to the next rising edge of clk. Usually, this input can be tied to ground because a reset is not required. After FPGA configuration, this module is in a fully operational mode.
data_in	In	20	The user connects this port to the data_out_20 port of the `gtp_interface_pll` module.
frame_en	In	1	This input enables the SDI framer function. When this input is High, the framer automatically readjusts the output word alignment to match the alignment of each timing reference signal (TRS), EAV or SAV. Normally, this input should always be High. However, if controlled properly, this input can be used to implement TRS alignment filtering. For example, if the nsp output is connected to the frame_en input, the framer ignores a single misaligned TRS, keeping the existing word alignment until the new word alignment is confirmed by a second matching TRS. It is important to turn off any TRS filtering during the synchronous switching lines by driving the frame_en input High on the synchronous switching lines.
ce_sd	Out	NUM_SD_CE	This output port consists of NUM_SD_CE identical copies of the clock enable signal generated by the SD-SDI data recovery unit in SD-SDI mode. In SD-SDI mode, this output is asserted at a 27 MHz rate. In HD-SDI and 3G-SDI modes, this output is always High.

Table 2: **I/O Ports of the `s6_sdi_rx_light_20b` Module *(Cont'd)***

Port Name	I/O	Width	Description
mode	Out	2	This output port indicates the current SDI mode of the receiver: • `00` = HD-SDI • `01` = SD-SDI • `10` = 3G-SDI When the receiver is not locked, the mode port changes values as the receiver searches for the correct SDI mode. During this time, the mode_locked output is Low. When the receiver detects the correct SDI mode, the mode_locked output goes High. This output port must be connected to the `s6gtp_sdi_control` module's rx_mode input.
mode_HD mode_SD mode_3G	Out	1	These three output ports are decoded versions of the mode port. They are provided for convenience. Unlike the mode port, which changes continuously as the receiver seeks to identify and lock to the incoming signal, these outputs are all forced Low when the receiver is not locked. The mode output matching the current SDI mode of the receiver is High when mode_locked is High.
mode_locked	Out	1	When this output is Low, the receiver is actively searching for the SDI mode that matches the input data stream. During this time, the mode output port changes frequently. When the receiver locks to the correct SDI mode, the mode_locked output goes High.
t_format	Out	4	This output port indicates the video transport timing format of the SDI signal in HD and 3G modes only. Refer to Table 3, page 8 for encoding. The output port is only valid when rx_locked is High. This output port is not valid in SD-SDI mode.
rx_locked	Out	1	This output is High when the receiver is locked to the incoming signal. In HD-SDI and 3G-SDI modes, this output is driven by the transport format detector that generates the t_format output. In SD-SDI mode, this output is identical to the mode_locked signal. Because this output is driven by the transport format detector in HD-SDI and 3G-SDI modes, there can be more than one video frame time of delay from when the receiver is actually locked to the input signal until rx_locked is asserted. During this time, the transport format detector determines the transport format.
nsp	Out	1	When this output is High, the framer has detected a TRS at a new word alignment. If frame_en is High, this output is only asserted briefly. If frame_en is Low, this output remains High until the framer is allowed to readjust to the new TRS alignment (by asserting frame_en High during the occurrence of a TRS).
ln_a	Out	11	The current line number captured from the LN words of the Y data stream is output on this port. This output is valid in HD-SDI and 3G-SDI modes, but not in SD-SDI mode. In 3G-SDI level B mode, the output value is the line number from the Y data stream of link A or HD-SDI signal 1. For any case where the interface line number is not the same as the picture line number, such as for 1080p 60 Hz carried on 3G-SDI level B or dual-link HD-SDI, the output value is the interface line number, not the picture line number.
a_vpid	Out	32	All four user data bytes of the SMPTE 352 packet from data stream 1 are output on this port in this format: most-significant byte to least-significant byte – byte4, byte3, byte2, byte1. This output port is valid only when a_vpid_valid is High. This port is potentially valid in any SDI mode, if there are SMPTE 352 packets embedded in the SDI signal. In 3G-SDI level A mode, the output data is the VPID data captured from data stream 1 (luma). In 3G-SDI level B mode, the output data is the VPID data captured from data stream 1 of link A (dual-link streams) or HD-SDI signal 1 (dual HD-SDI signals).
a_vpid_valid	Out	1	This output is High when a_vpid is valid.

Table 2: **I/O Ports of the `s6_sdi_rx_light_20b` Module *(Cont'd)***

Port Name	I/O	Width	Description
b_vpid	Out	32	All four user data bytes of the SMPTE 352 packet from data stream 2 are output on this port in this format: most-significant byte to least-significant byte – byte4, byte3, byte2, byte1. This output is valid only in 3G-SDI mode and only when b_vpid_valid is High. In 3G-SDI level A mode, the output data is the VPID data captured from data stream 2 (chroma). In 3G-SDI level B mode, the output data the VPID data captured from data stream 1 of link B (dual-link streams) or HD-SDI signal 2 (dual HD-SDI signals).
b_vpid_valid	Out	1	This output is High when b_vpid is valid.
crc_err_a	Out	1	This output is asserted High for one sample period when a CRC error is detected on the previous video line. For 3G-SDI level B mode, this output indicates CRC errors on data stream 1 only. There is a second output called crc_err2 that indicates CRC errors on data stream 2 for 3G-SDI level B mode. This output is not valid in SD-SDI mode. The crc_err_a output is asserted High for a single clock cycle when a CRC error is detected.
ds1_a	Out	10	The recovered data stream 1 is output on this port. The contents of this data stream are dependent on the SDI mode: • SD-SDI: Multiplexed Y/C data stream • HD-SDI: Y data stream • 3G-SDI level A: Data stream 1 • 3G-SDI level B: Data stream 1 of link A or of HD-SDI signal 1
ds2_a	Out	10	The recovered data stream 2 is output on this port. The contents of this data stream are dependent on the SDI mode: • SD-SDI: Not used • HD-SDI: C data stream • 3G-SDI level A: Data stream 2 • 3G-SDI level B: Data stream 2 of link A or of HD-SDI signal 1
eav	Out	1	This output is asserted High for one sample time when the XYZ word of an EAV is present on the data stream output ports.
sav	Out	1	This output is asserted High for one sample time when the XYZ word of an SAV is present on the data stream output ports.
trs	Out	1	This output is asserted High for four consecutive sample times because all four words of an EAV or SAV are output on the data stream ports.
Outputs Used Only in 3G-SDI Level B Mode			
dout_rdy_3G	Out	NUM_3G_DRDY	In 3G-SDI level B mode, the output data rate is 74.25 MHz, but the clock frequency is 148.5 MHz. The dout_rdy_3G outputs are asserted at a 74.25 MHz rate in 3G-SDI level B mode. This output is always High in all other modes, allowing it to be used as a clock enable to downstream modules.
level_b_3G	Out	1	In 3G-SDI mode, this output is asserted High when the input signal is level B and Low when it is level A.
ds1_b	Out	10	In 3G-SDI mode, the recovered data stream 1 of link B or of HD-SDI signal 2 is output on this port.
ds2_b	Out	10	In 3G-SDI mode, the recovered data stream 2 of link B or of HD-SDI signal 2 is output on this port.
crc_err_b	Out	1	This output is the CRC error indicator for link B or HD-SDI signal 2. The crc_err_b output is asserted High for a single clock cycle when a CRC error is detected.
ln_b	Out	11	This output port is only valid in 3G-SDI level B mode. It outputs the line number for the Y data stream of link B or the HD-SDI signal 2. For any case where the interface line number is not the same as the picture line number, the line number output on this port is the interface line number, not the picture line number.

Table 3 shows the encoding of the t_format output port. This port indicates the transport format (not always the same as the picture format) calculated by the receiver by counting words per line and active lines per field or frame. The t_format port can uniquely identify most video transport formats, but cannot distinguish between transport formats that have identical timing. For example, it cannot differentiate between 1080i 60 Hz and 1080PsF 30 Hz (1080p 30 Hz video carried on a 1080i 60 Hz transport) because there is no difference in the transport timing.

In dual-link HD-SDI and 3G-SDI level B modes, the 1080p 50 and 60 Hz video formats are carried on an interlaced transport. The module reports them as being 1080i 50 Hz and 1080i 60 Hz, respectively. However, in 3G-SDI level A mode, these two video formats are carried progressively and are uniquely identified as 1080p 50 Hz and 1080p 60 Hz (codes `1110` and `1101`, respectively).

The transport format detector does not distinguish between frame rates that are 1000 ppm different, such as 1080p 60 Hz and 1080p 59.94 Hz. However, the rx_rate output of the `s6gtp_sdi_control` module can be used to distinguish between these otherwise identical frame rates.

Table 3: **t_format Output Port Encoding for the `s6_sdi_rx_light_20b` Module**

t_format	Standard	Video Format (Frame Rate for p and Field Rate for i)
0000	SMPTE 260M	1035i 59.94 Hz and 60 Hz
0001	SMPTE 295M	1080i 50 Hz
0010	SMPTE 274M	1080i 59.94 Hz and 60 Hz 1080PsF 29.97 Hz and 30 Hz
0011	SMPTE 274M	1080i 50 Hz 1080PsF 25 Hz
0100	SMPTE 274M	1080p 29.97 Hz and 30 Hz 1080p 59.94 Hz and 60 Hz (3G-SDI level B only)
0101	SMPTE 274M	1080p 25 Hz 1080p 50 Hz (3G-SDI level B only)
0110	SMPTE 274M	1080p 23.98 Hz and 24 Hz
0111	SMPTE 296M	720p 59.94 Hz and 60 Hz
1000	SMPTE 274M	1080PsF 23.98 Hz and 24 Hz
1001	SMPTE 296M	720p 50 Hz
1010	SMPTE 296M	720p 29.97 Hz and 30 Hz
1011	SMPTE 296M	720p 25 Hz
1100	SMPTE 296M	720p 23.98 Hz or 24 Hz
1101	SMPTE 274M	1080p 59.94 Hz or 60 Hz (3G-SDI level A only)
1110	SMPTE 274M	1080p 50 Hz (3G-SDI level A only)
1111	Reserved	

Table 4 describes the parameters (Verilog) or generics (VHDL) of the `s6_sdi_rx_light_20b` module.

Table 4: **Parameters for the `s6_sdi_rx_light_20b` Module**

Parameter	Default	Description
NUM_SD_CE	2	This parameter specifies the number of identical SD-SDI clock enable output signals provided on the ce_sd port. It specifies the width of the ce_sd port. All bits of the ce_sd port are identical, but separately generate copies. This parameter must never be set to less than 1.
NUM_3G _DRDY	2	This parameter specifies the number of identical 3G-SDI level B data ready outputs provided by the module on the dout_rdy_3G port. It specifies the width of the dout_rdy_3G port. All bits of the dout_rdy_3G port are identical, but separately generated copies. This parameter must never be set to less than 1.
ERRCNT_WIDTH	4	This parameter is used by the SDI mode detection function. It specifies the number of bits used in the SDI mode detection error counter. This counter must be wide enough to support counting up to MAX_ERRS_LOCKED and MAX_ERRS_UNLOCKED values.
MAX_ERRS_LOCKED	15	This parameter is used by the SDI mode detection function. It specifies the maximum number of consecutive lines with errors allowed while the receiver is locked to an SDI signal. When MAX_ERRS_LOCKED consecutive lines with errors are detected, the receiver moves to the unlocked state and begins searching for the new SDI mode. Reducing this value causes the SDI receiver to respond more quickly to an unlocked condition, reducing the time it takes to relock when the SDI input signal changes modes. However, reducing this value can also cause the receiver to be more prone to prematurely moving to an unlocked state during a burst of errors on the SDI signal.
MAX_ERRS_UNLOCKED	2	This parameter is used by the SDI mode detection function. When the receiver is actively searching for the SDI mode, this parameter specifies the maximum number of consecutive lines with errors allowed before the SDI mode search continues by moving to the next mode. Increasing this value gives the SDI receiver more time to lock to the SDI signal as it tries to lock in each SDI mode. However, testing has shown that a value of 2 provides adequate lock time while also minimizing the SDI mode detection search time.

GTP RX Interface Module

Figure 2 shows a block diagram of the `gtp_interface_pll` module. This module abstracts the clock creation and buffering so that alternate clock creation and distribution schemes can be employed by the user, if desired. The 20-bit data is routed through this module to facilitate synchronization between clock domains, if it is required for the alternate clocking schemes.

The module creates the two synchronized user clocks required for interfacing to the GTP transceiver: usrclk and usrclk2. Pipe_clk is identical to usrclk2.

The input clock is the RX recovered clock from the GTP transceiver. Table 5 describes the inputs and outputs of this module. Figure 3 shows the `gtp_interface_pll` module in the context of a triple-rate SDI receiver. This module is also used for the GTP transmitter, but it requires a separate instance, and the usage and clock frequencies are different (see GTP TX Interface Module, page 22).

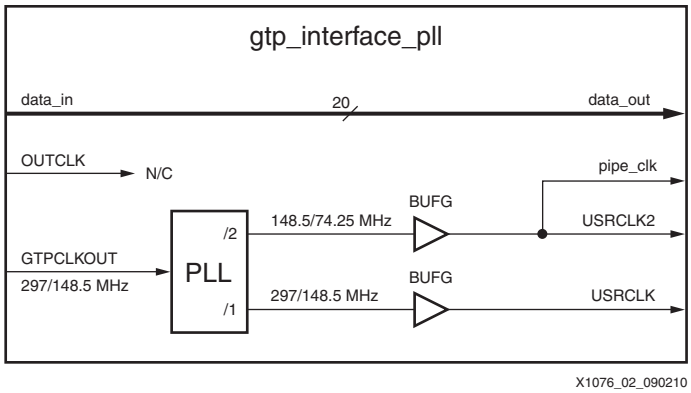


Figure 2: **Block Diagram of the gtp_interface_pll Module**

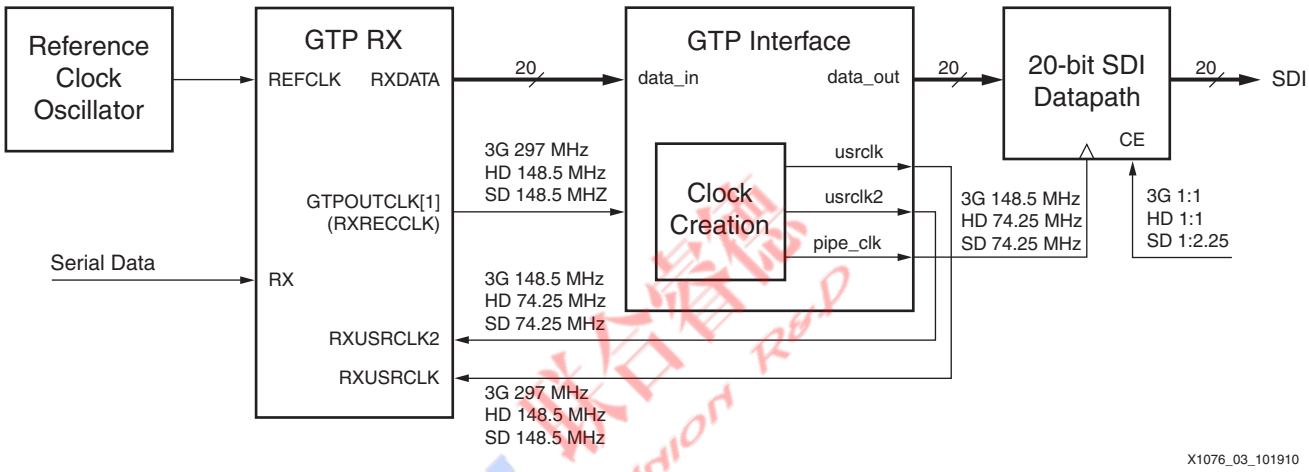


Figure 3: **GTP Interface Module in RX Context**

Table 5: **I/O Ports of the gtp_interface_pll Module**

Port Name	I/O	Width	Description
outclk	In	1	Not used. Output clocks using dedicated clock routing come through gtpclkout.
gtpoutclk	In	1	PLL input clock on dedicated clock routing. Nominally, 297 MHz for 3G-SDI and 148.5 MHz for HD-SDI and SD-SDI. This should be connected to the GTPOUTCLK[1] port of the GTP transceiver through a BUFIO2 buffer.
pll_reset_in	In	1	Asynchronous PLL reset. 1 = Reset PLL. 0 = Do not reset PLL.
data_in	In	20	20-bit input data. This should be connected to the RXDATA port of the GTP transceiver.
usrclk	Out	1	Output user clock at one times the input clock rate. Nominally, 297 MHz for 3G-SDI, and 148.5 MHz for HD-SDI and SD-SDI. This is used to drive the USRCLK input of the GTP transceiver.
usrclk2	Out	1	Output user clock at one-half times the input clock rate. Nominally, 148.5 MHz for 3G-SDI, and 74.25 MHz for HD-SDI and SD-SDI. This output drives the USRCLK2 input of the GTP transceiver.
pipe_clk	Out	1	Pipeline clock for downstream pixel processing. This must be at the same frequency as usrclk2. In the reference design, it is the same clock.

Table 5: **I/O Ports of the gtp_interface_pll Module *(Cont'd)***

Port Name	I/O	Width	Description
data_out	Out	20	20-bit output data. This should be connected to the data_in port of the S6_sdi_rx_light module.
pll_locked_out	Out	1	Locked flag from the PLL. 1 = Locked. 0 = Unlocked.

Operation of the Triple-Rate SDI Receiver in the Various SDI Modes

The triple-rate SDI receiver automatically determines the standard of the incoming SDI mode (SD-SDI, HD-SDI, 3G-SDI level A, or 3G-SDI level B). It does this by sequentially trying to lock to each SDI mode until it finds the correct SDI mode. When locked to the correct SDI mode, the receiver configures itself for correct operation in that mode. The recovered clock frequency and the number and data rate of the output data streams depend on the SDI mode.

Triple-Rate SDI Clocking

The GTP receiver's CDR unit requires a reference clock, which can be either 74.25 MHz or 74.25/1.001 MHz. Some integer multiples of these frequencies are also supported. In particular, 148.5 MHz and 148.5/1.001 MHz are also commonly used. Only a single reference clock frequency is required to support SD-SDI at 270 Mb/s, HD-SDI at both bit rates, and 3G-SDI at both bit rates. The frequency of the reference clock should not change. Any change in the reference clock frequency requires resetting the GTP RX, including the RX PMA PLL.

The GTP receiver recovers a clock in 3G-SDI and HD-SDI modes, but not in SD-SDI mode. The recovered clock from the GTP transceiver is output on bit 1 of the GTPOUTCLK port. This port is connected to dedicated clock resources and is used in place of RXRECCLK to carry the recovered clock. The frequency of the recovered clock depends on the current SDI mode of the receiver. In HD-SDI mode, it is 148.5 MHz or 148.5/1.001 MHz. In 3G-SDI mode, it is 297 MHz or 297/1.001 MHz. In SD-SDI mode, the CDR unit is locked to the reference clock and GTPOUTCLK[1] is either 148.5 MHz or 148.5/1.001 MHz, depending on the frequency of the reference clock.

The GTPOUTCLK[1] output of the GTP receiver usually should be buffered by a BUFIO2 buffer. The output of this clock buffer drives the gtpoutclk1 input of the `gtp_interface_pll` module. The `gtp_interface_pll` module is shown in Figure 2. This module creates two synchronous clocks from the RX recovered clock by dividing the clock by 1 and 2 using a PLL. These clocks are buffered by global clock buffers. The usrclk output is at the recovered clock rate. The usrclk2 output is at one-half the recovered clock rate. These two clocks are required by the GTP RX interface. They must be synchronous with each other and with the data going into the GTP interface. The pipe_clk is identical to usrclk2 at one-half the rate of the recovered clock, and thus can be used to clock the 20-bit `s6_sdi_rx_light_20b` module and any other downstream logic. It can also be used to drive any additional downstream modules that need to be clocked by the recovered clock.

Internally, most of the logic in the `s6_sdi_rx_light_20b` module runs at the pipe clock frequency for HD-SDI and 3G-SDI (some portions run at half the pipe clock frequency in 3G-SDI level B mode). For SD-SDI, the internal data rate is 27 MHz. Clock enables are used in the `s6_sdi_rx_light_20b` module to run the various sections at the proper rates when the data rate is different from the clock frequency.

The `s6_sdi_rx_light_20b` module also supplies one or more copies of the SD-SDI clock enable on the ce_sd output port. These clock enables can be used, in conjunction with the recovered clock, to clock logic downstream from the `s6_sdi_rx_light_20b` module at the 27 MHz SD-SDI data rate. The NUM_SD_CE parameter/generic specifies the number of identical copies of ce_sd output by the module. NUM_SD_CE must never be set to less than 1.

The source of the ce_sd clock enable signal is the SD-SDI data recovery unit (DRU) located inside the `s6_sdi_rx_light_20b` module. Typically, the cadence of a ce_sd signal is 3/3/3/2 cycles of the clock. However, the cadence varies occasionally to make up for differences between the actual recovered data rate and the frequency of the reference clock. In all modes except SD-SDI, ce_sd is always High.

The `s6_sdi_rx_light_20b` module also outputs a dout_rdy_3G data ready signal. In 3G-SDI level B mode, dout_rdy_3G toggles at half the recovered clock frequency because the output data rate is 74.25 MHz and the clock frequency is 148.5 MHz. In all modes except 3G-SDI level B, dout_rdy_3G is always High. The NUM_3G_DRDY parameter/generic specifies the number of identical copies of the dout_rdy_3G signal output by the module. NUM_3G_DRDY should never be set to less than 1.

The ce_sd clock enable signals can be used as clock enables to those datapaths that only operate at the SD-SDI data rate, such as the optional SD-SDI EDH processor. The ce_sd signal is always High when not in SD-SDI mode. The dout_rdy_3G signal can be used as a clock enable to those datapaths that operate in HD-SDI and 3G-SDI modes, but not in SD-SDI mode. In SD-SDI mode, dout_rdy_3G is always High. If a datapath must operate correctly in all three SDI modes, it should use a clock enable that is the AND of a ce_sd signal and a dout_rdy_3G signal.

SD-SDI Output Timing

When the triple-rate SDI receiver operates in SD-SDI mode, the frequency of rxpipeclk is 74.25 MHz or 74.25/1.001 MHz, depending on the reference clock frequency. The recovered SD-SDI data stream is output on the ds1_a port with the Y and C components interleaved at a 27 MHz data rate. The timing signals trs, eav, and sav are valid.

In SD-SDI mode, pipe_clk is not a recovered clock because the GTP receiver is locked to the reference clock and asynchronously samples the input bitstream. pipe_clk is, therefore, an exact multiple of the reference clock supplied to the GTP receiver.

A data recovery unit (DRU) in the `s6_sdi_rx_light_20b` module recovers the actual data stream from the oversampled data. The DRU provides a data ready signal that is asserted when it has a 10-bit data word ready. The `s6_sdi_rx_light_20b` module outputs this data ready signal on the ce_sd output. On average, this output is asserted once every 2.75 clock cycles by using a 3/3/3/2 clock cycle cadence. The output cadence is occasionally altered when the DRU needs to catch up to the actual data rate. This occurs because the GTP transceiver reference clock is a local clock and is asynchronous to the actual timing of the incoming SD-SDI bitstream. High amounts of jitter on the SD-SDI can also cause the cadence to vary.

Figure 4 shows the timing of the SD video and timing signals output by the `s6_sdi_rx_light_20b` module. The outputs only change on the rising edge of clk when ce_sd is High. The timing of the EAV sequence is shown. The sav output signal has the same timing as the eav signal except that it is asserted during SAV sequences.

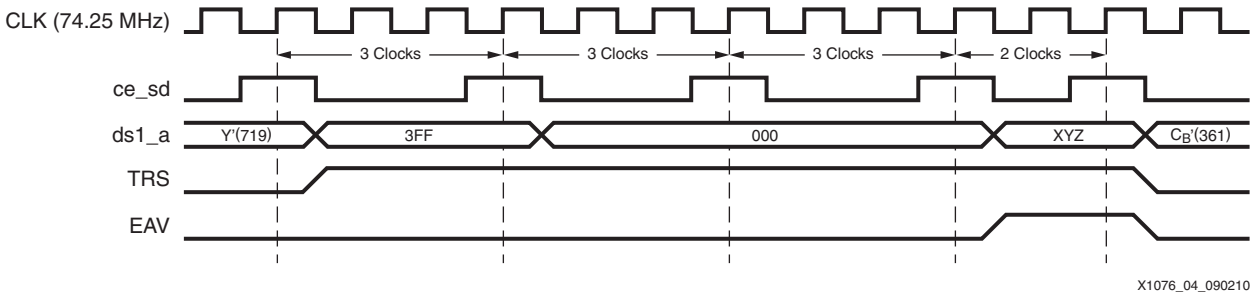


Figure 4: **SD-SDI RX Timing Diagram**

HD-SDI Output Timing

When the triple-rate SDI receiver operates in HD-SDI mode, pipe_clk is based on a true recovered clock and runs at 74.25 MHz or 74.25/1.001 MHz, depending on the HD-SDI bit rate. The Y and C data streams of the HD-SDI signal are output on the ds1_a and ds2_a ports, respectively, along with the timing signals trs, eav, and sav. The line number is output on the ln_a port. In HD-SDI mode, the line number changes during the CRC0 word. Figure 5 shows the timing of the HD-SDI outputs.

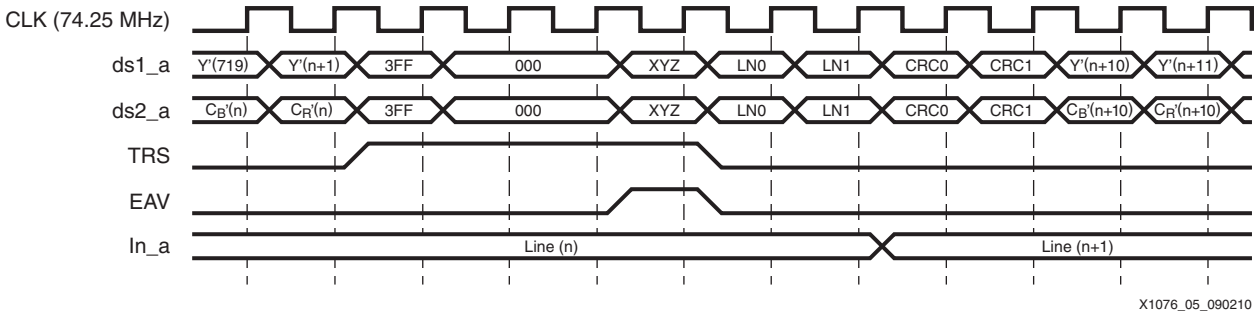


Figure 5: **HD-SDI RX Timing Diagram**

3G-SDI Level A Output Timing

When the triple-rate SDI receiver operates in 3G-SDI level A mode, the frequency of pipe_clk is 148.5 MHz or 148.5/1.001 MHz, depending on the 3G-SDI bit rate.

Figure 6 shows the output timing of the `s6_sdi_rx_light_20b` module when receiving a 1080p 50, 59.94, or 60 Hz signal in 3G-SDI level A mode. These video formats do not require further unpacking of the data streams, because ds1_a carries the luma component and ds2_a carries the multiplexed chroma components.

Other video formats, such as 4:4:4 10-bit or 12-bit, have identical timing to that shown in Figure 6, but the video samples are packed as specified by the SMPTE 425 so that it takes two consecutive words on each data stream to carry a single video sample. The `s6_sdi_rx_light_20b` module does not unpack these other video formats, but outputs the two data streams exactly as described in the 3G-SDI level A data stream mapping sections of SMPTE 425.

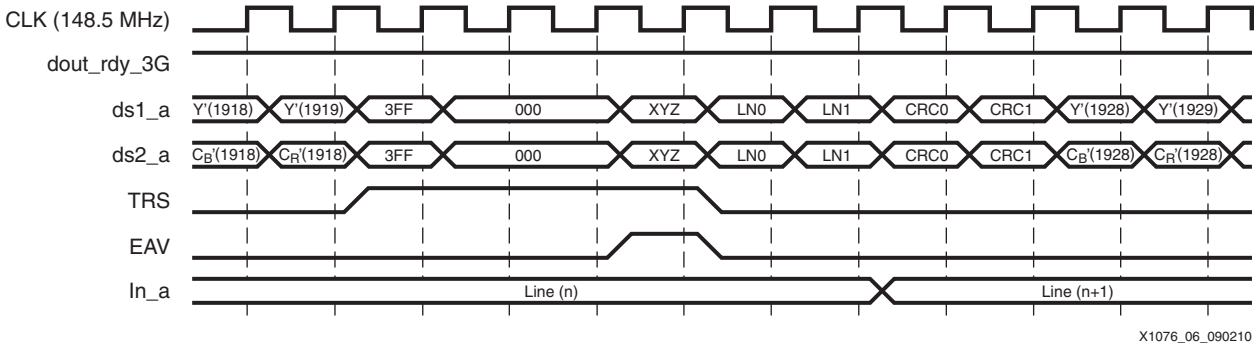


Figure 6: **3G-SDI Level A RX Timing (1080p 50 Hz or 60 Hz)**

3G-SDI Level B Output Timing

When the triple-rate SDI receiver operates in 3G-SDI level B mode, the frequency of pipe_clk is 148.5 MHz or 148.5/1.001 MHz, depending on the 3G-SDI bit rate. However, in this mode, there are four 10-bit data streams output by the `s6_sdi_rx_light_20b` module. The data rate of these data streams is half the frequency of pipe_clk, therefore, the dout_rdy_3G signal is asserted every other clock cycle and acts as a clock enable.

Figure 7 shows the output timing of the `s6_sdi_rx_light_20b` module when receiving a 3G-SDI level B signal.

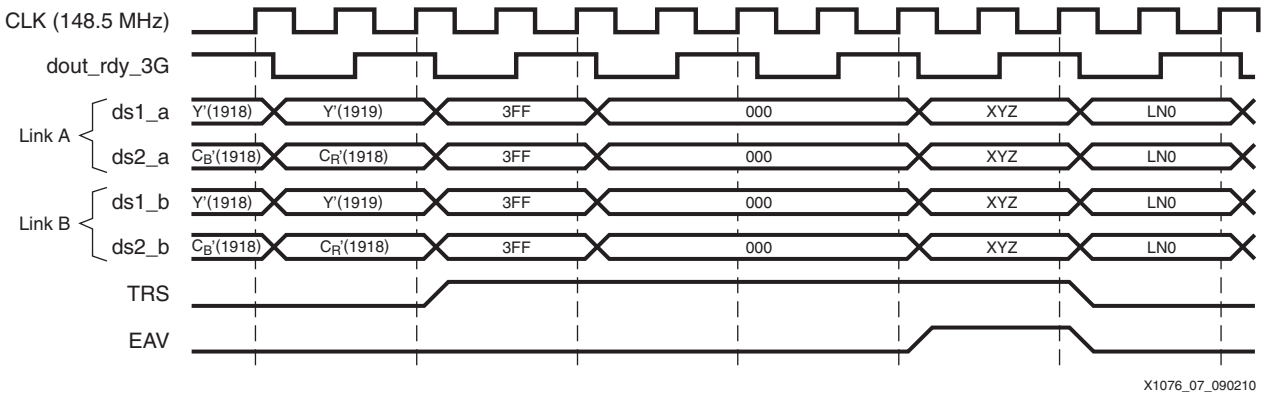


Figure 7: **3G-SDI Level B RX Timing**

Both line number output ports are active. When the level B signal is transporting SMPTE 372 dual-link data streams, the values on both ln_a and ln_b are identical and indicate the interface line number, not the picture line number. When the level B signal is transporting two independent HD-SDI signals, the two line number values are not necessarily the same, depending on whether the two HD-SDI signals are frame locked or not. The ln_a and ln_b ports, not shown in Figure 7, change at the same relative position as they do for 3G-SDI level A and HD-SDI, as the CRC0 word is output on the data streams.

When the 3G-SDI level B signal carries SMPTE 372 dual-link data streams, the link A data streams are output on ds1_a and ds2_a and the link B data streams are output on ds1_b and ds2_b. These four links carry video mapped as required by SMPTE 372. The `s6_sdi_rx_light_20b` module does not unpack the data streams into video, but outputs them as SMPTE 372 data streams.

When the 3G-SDI level B signal carries two independent HD-SDI streams, the first HD-SDI stream is output with the luma component on ds1_a and the multiplexed chroma components on ds2_a. The second HD-SDI stream is output with the luma component on ds1_b and the multiplexed chroma component on ds2_b. These two HD-SDI streams are horizontally synchronized so that their EAVs and SAVs always line up exactly, therefore there is only a single set of eav, sav, and trs timing signals.

Other Triple-Rate SDI RX Design Considerations

To incorporate the triple-rate receiver into a complete system, the designer should take into account some additional considerations. This section describes those considerations.

HD-SDI and 3G-SDI Bit Rate Detection

The `s6gtp_sdi_control` module contains a bit rate detector. The rx_rate output of the `s6gtp_sdi_control` module is `0` when the incoming bit rate is 1.485 Gb/s or 2.97 Gb/s. It is `1` when the incoming bit rate is 1.485/1.001 Gb/s or 2.97/1.001 Gb/s. The bit rate detector depends on a known constant frequency clock supplied to the `s6gtp_sdi_control` module on its dclk port. Refer to SDI GTP Transceiver Control Module, page 32 for a detailed description of this module.

Dual-Link HD-SDI

To implement a dual-link HD-SDI receiver, two triple-rate SDI receivers are paired together with one receiving link A and the other receiving link B. Typically, the received data streams for the two links are skewed, so the skew must be removed. Next, the data streams can be unpacked into video streams, if desired.

Deskewing the data streams involves watching the EAV or SAV signals from each link and delaying the data streams of the link whose EAV becomes asserted first by an appropriate amount to match the timing of the other link. The Spartan-6 FPGA SRL elements make perfect delay devices for implementing this deskew function.

Processing Embedded Audio and Other Ancillary Data

The data streams that are output from the triple-rate SDI receiver always have all ancillary data intact, including embedded audio packets. Modules designed to process ancillary data can be connected to the data streams, clock enables, and other timing signals output by the triple-rate SDI receiver.

SMPTE 352 Video Payload ID Packets

The triple-rate SDI receiver module captures SMPTE 352 packets present in the data streams for all SDI modes. For SD-SDI and HD-SDI, the four data bytes of the SMPTE 352 packet are output on the a_vpid port. The a_vpid_valid port indicates when valid SMPTE 352 packets have been captured. This output has some hysteresis so that SMPTE 352 packets can be missing from a few fields or frames before the valid signal is negated. During the time that the valid output is asserted and new SMPTE 352 packets are not found, the data from the last valid SMPTE 352 packet received is output on the a_vpid port.

The 3G-SDI standard requires SMPTE 352 packets in both data streams. The triple-rate SDI receiver captures the SMPTE 352 packets from both streams, outputting the data from the packet in data stream 1 on a_vpid and the data from the packet in data stream 2 on b_vpid. The module also provides individual a_vpid_valid and b_vpid_valid outputs.

SD-SDI EDH Error Detection

The triple-rate SDI receiver detects CRC errors in HD-SDI and 3G-SDI modes, but it does not contain an EDH error checker. However, either of the two EDH processors described in Chapters 6 and 7 of *Audio/Video Connectivity Solutions for Virtex-II Pro and Virtex-4 FPGAs* [Ref 1] can be connected downstream from the `s6_sdi_rx_light_20b` module to check the SD-SDI data stream for EDH errors.

SD-SDI Data Recovery Unit

The triple-rate SDI receiver uses a DRU to recover the SD-SDI data. This DRU is based on the digital PLL data recovery design described in *Dynamically Programmable DRU for High-Speed Serial I/O* [Ref 2], but has been optimized for SD-SDI and ported to the Spartan-6 FPGA. This SD-SDI DRU has been optimized for use with 5.5X oversampled 270 Mb/s data. By optimizing the DRU specifically for SD-SDI, it is much smaller than the general-purpose implementation of the DRU. The SD-SDI DRU has also been optimized so that the reference clock used for SD-SDI reception can be either 74.25 MHz or 74.25/1.001 MHz.

Electrical Interface

The GTP receiver must be connected to the SDI connector through an SDI cable equalizer. The equalizer serves two purposes. It equalizes the SDI signal to compensate for signal distortion and attenuation, and it converts the single-ended 75Ω SDI signal to a differential signal compatible with the GTP receiver input. Most, if not all, SDI cable equalizers currently available do not have a common mode output directly compatible with the Spartan-6 FPGA GTP transceiver inputs. Therefore AC coupling is required to shift the common mode voltage of the signal as it enters the GTP receiver. The GTP receiver has built-in AC coupling, but these internal capacitors are not adequate to support the long run lengths of the SDI signals. The internal AC coupling must be bypassed, and external AC coupling capacitors must be used. The value of these external capacitors is usually about 4.7 µF. Nothing else needs to be done to set the common mode voltage because the GTP receiver provides a correct termination voltage to set the common mode voltage on its inputs.

Triple-Rate SDI Transmitter Reference Design

The Spartan-6 FPGA SDI transmitter provides the basic operations necessary to support SD-SDI, HD-SDI, dual-link HD-SDI, and 3G-SDI transmission. The transmitter does not perform video mapping for 3G-SDI or dual-link HD-SDI. Video formats that require mapping must be mapped into SDI data streams prior to the triple-rate SDI TX module. These video formats include those in Table 1 marked “external” in the “SDI Stream Mapping column: 3G-SDI level A formats except 1080p 50 Hz, 59.94 Hz, and 60 Hz, all 3G-SDI level B formats, and all dual-link HD-SDI formats.

The Spartan-6 FPGA GTP transceiver triple-rate SDI transmitter has these features:

- A 20-bit GTP interface, supported on -3 and faster Spartan-6 devices.
- Only two reference clock frequencies are required to support all SDI modes:
  - 148.5 MHz for SD-SDI at 270 Mb/s, HD-SDI at 1.485 Gb/s, and 3G-SDI at 2.97 Gb/s.
  - 148.5/1.001 MHz for HD-SDI at 1.485/1.001 Gb/s and 3G-SDI at 2.97/1.001 Gb/s.
- Direct support for 3G-SDI level A transmission of 1080p 50 Hz, 59.94 Hz, and 60 Hz video.
- Transmission of preformatted dual-link HD-SDI streams via either dual-link HD-SDI or 3G-SDI level B formats.
- With the addition of a 3G-SDI level A mapping module, supports all 3G-SDI level A compatible video formats.
- Direct support for transmission of two independent HD-SDI streams via 3G-SDI level B format.
- EDH packets generated and updated for SD-SDI mode.
- CRC values can be generated and inserted for HD-SDI and 3G-SDI modes.
- Line number words inserted for HD-SDI and 3G-SDI modes.
- SMPTE 352M video payload ID packets can be generated and inserted in all SDI modes.

A block diagram of the Spartan-6 FPGA GTP Transceiver triple-rate SDI transmitter is shown in Figure 8.

The transmitter datapath consists of three modules: `triple_sdi_vpid_insert`, `triple_sdi_tx_output_20b`, and `gtp_interface_pll`. These three modules, combined with the GTP transceiver and the `s6gtp_sdi_control` module, form the triple-rate SDI transmitter design. One `s6gtp_sdi_control` module is required per GTP transceiver (RX/TX pair). If both the RX and the TX in a GTP transceiver are used for SDI, only one `s6gtp_sdi_control` module is required for that transceiver. Figure 8 only shows the transmitter connections of the GTP transceiver wrapper and the `s6gtp_sdi_control` module.

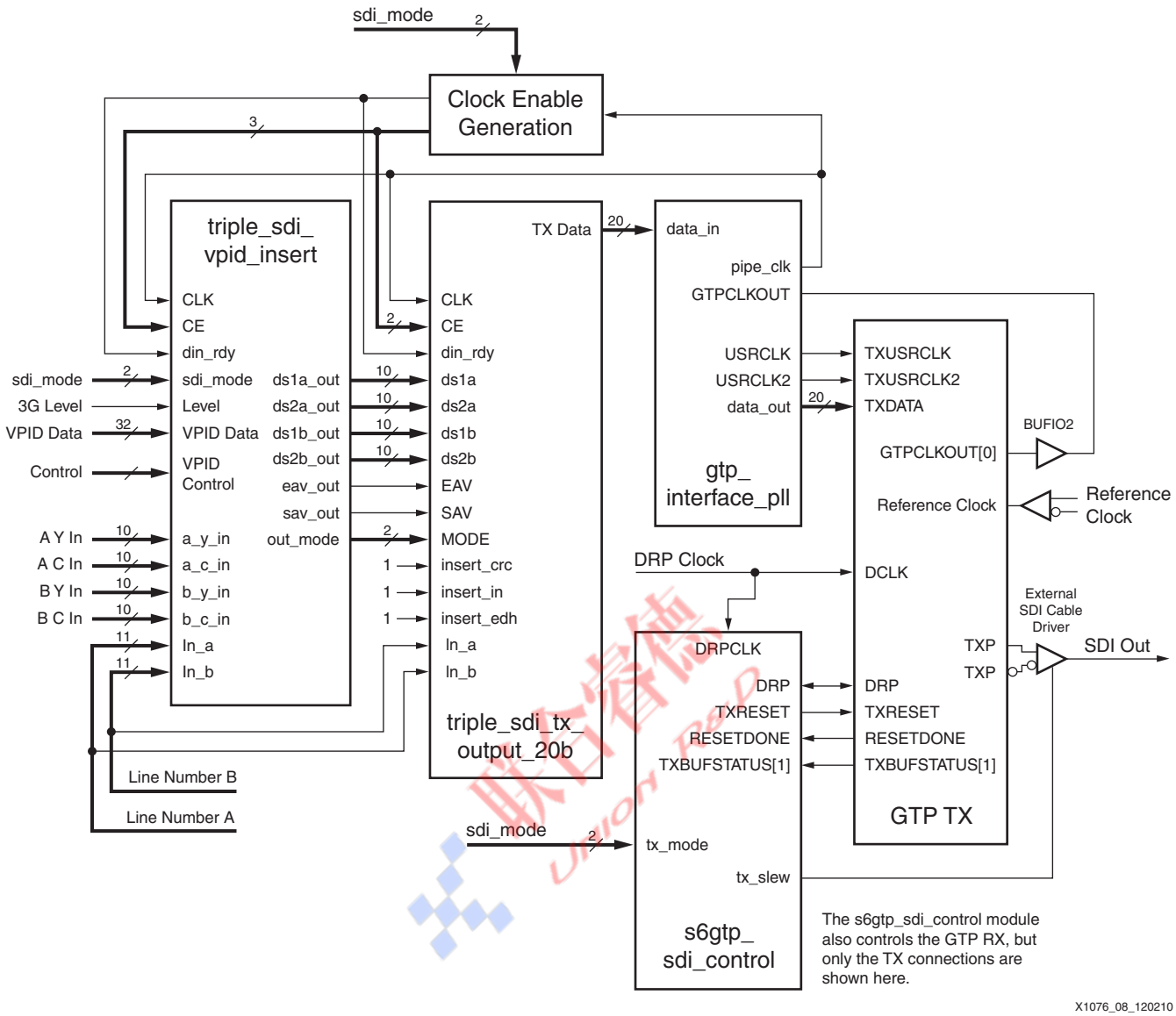


Figure 8: **Spartan-6 FPGA GTP Transceiver Triple-Rate SDI Transmitter Block Diagram**

Triple-Rate SDI Transmitter Datapath Modules

The triple-rate SDI transmitter datapath consists of two datapath modules, the `triple_sdi_vpid_insert` module, which inserts SMPTE 352 video payload ID packets into the data streams, and the `triple_sdi_tx_output_20b` module, which creates a final output data stream suitable for input to the TXDATA port of the GTP transmitter.

Triple-Rate SDI VPID Insert Module

Table 6 describes the ports of the `triple_sdi_vpid_insert` module. This module is the front-end of the triple-rate SDI transmitter. Its main function is to insert SMPTE 352 video payload ID packets into the data streams, but it also provides some additional functions required by the `triple_sdi_tx_output_20b` module, such as detecting and identifying EAV and SAV sequences and generating the out_mode signal that connects to the mode port of the output module. Even if SMPTE 352 packets do not need to be inserted in a particular application (these packets are required for 3G-SDI and dual-link HD-SDI modes), the `triple_sdi_vpid_insert` module is still required to perform these additional functions. If the `triple_sdi_vpid_insert` module's enable input port is wired Low, the actual

SMPTE 352 packet generation and insertion logic is optimized out of the design, and only the additional required functions of the module are retained by the synthesizer.

Table 6: **I/O Ports of the `triple_sdi_vpid_insert` Module**

Port Name	I/O	Width	Description
clk	In	1	This clock input must be driven by the pipe_clk. It must have a frequency of 74.25 MHz or 74.25/1.001 MHz for HD-SDI, 148.5 MHz or 148.5/1.001 MHz for 3G-SDI, and 148.5 MHz for SD-SDI modes.
ce	In	1	The clock enable must be asserted at a 27 MHz rate for SD-SDI mode (with a mandatory 5/6/5/6 clock cycle cadence). For all other SDI modes, the clock enable is always High.
din_rdy	In	1	For SD-SDI, HD-SDI, and level A 3G-SDI modes, this input must be kept High at all times. For level B 3G-SDI mode, this input must be asserted every other clock cycle.
rst	In	1	This asynchronous reset input resets the module when High. The falling edge of this reset signal must meet the reset recovery time of all flip-flops relative to the next rising edge of clk.
sdi_mode	In	1	This input port is used to select the transmitter SDI mode: • `00` = HD-SDI (including dual-link HD-SDI) • `01` = SD-SDI • `10` = 3G-SDI • `11` = Invalid
level	In	1	In 3G-SDI mode, this input determines whether the module inserts SMPTE 352 packets for level A (level = Low) or for level B (level = High). Because these two 3G-SDI levels have different requirements for placement of SMPTE 352 packets, this input must be properly controlled, otherwise the data streams generated by the module are not legal.
enable	In	1	When this input is High, SMPTE 352 packets are inserted into the data streams, otherwise the packets are not inserted. SMPTE 352 packets are mandatory in 3G-SDI and dual-link HD-SDI modes.
overwrite	In	1	If this input is High, SMPTE 352 packets already present in the data streams are overwritten. If this input is Low, existing SMPTE 352 packets are not overwritten. When transporting SMPTE 372 dual-link data streams on a 3G-SDI level B interface, existing SMPTE 352 packets in the data streams must be updated to indicate that the interface is 3G-SDI rather than HD-SDI mode. This module updates these packets only when overwrite is High.
byte1	In	8	This value is inserted as the first user data word of the SMPTE 352 packet. It must be valid during the entire HANC interval.
byte2	In	8	This value is inserted as the second user data word of the SMPTE 352 packet. It must be valid during the entire HANC interval.
byte3	In	8	This value is inserted as the third user data word of the SMPTE 352 packet. It must be valid during the entire HANC interval.
byte4a	In	8	This value is inserted as the fourth user data word of the SMPTE 352 packet. This word is used for the SMPTE 352 packets inserted into SD-SDI, HD-SDI, and 3G-SDI level A data streams. For 3G-SDI level B and dual-link HD-SDI modes, this value is used for the SMPTE 352 packet inserted into Y channel of link A only. This input must be valid during the entire HANC interval.
byte4b	In	8	This value is inserted as the fourth user data word of SMPTE 352 packets inserted in the Y channel of link B for 3G-SDI level B and dual-link HD-SDI modes only. This input value is not used for SD-SDI, HD-SDI, or 3G-SDI level A modes. This input must be valid during the entire HANC interval.

Table 6: **I/O Ports of the `triple_sdi_vpid_insert` Module *(Cont'd)***

Port Name	I/O	Width	Description
ln_a	In	11	The current line number must be provided to the module through this port if SMPTE 352 packet insertion is enabled. If SMPTE 352 packet insertion is disabled in a particular SDI mode, SD-SDI for example, valid line numbers do not need to be supplied on this port in that SDI mode. SD-SDI only uses 10-bit line numbers, so bit 10 of the port must be `0` in SD-SDI mode. The line number must be valid at least one clock cycle before the start of the HANC space (by the XYZ word of the EAV) and must remain valid during the entire HANC interval. This input is the only line number input used for SD-SDI, HD-SDI, and 3G-SDI level A modes. For 3G-SDI level B mode, a second line number input port, ln_b, is also provided. The line numbers input on the ln_a port are used only to identify the line for the purposes of inserting SMPTE 352 packets. For video formats where the picture line number is different from the transport line number, the value supplied on this port must be the transport line number.
ln_b	In	11	This second line number input port is used only for 3G-SDI level B format. This additional line number port allows the two separate HD-SDI signals to be vertically unsynchronized when level B carries two independent HD-SDI signals. This input port has the same timing and other requirements described for ln_a.
line_f1	In	11	The SMPTE 352 packet is inserted in the HANC space of the line number specified by this input port. For interlaced video, this input port specifies a line number in field 1. For progressive video, this specifies the only line in the frame where the packet is inserted. The input value must be valid during the entire HANC interval.
line_f2	In	11	For interlaced video, a SMPTE 352 packet is inserted on the line number in field 2 indicated by this value. For progressive video, this input port must be disabled by holding the line_f2_en port Low. The input value must be valid during the entire HANC interval.
line_f2_en	In	1	This input controls whether or not SMPTE 352 packets are inserted on the line indicated by line_f2. For interlaced video, this input must be High. For progressive video, this input must be Low. For progressive video transported on an interlaced transport, such as 1080p 60 Hz transported by either 3G-SDI level B or dual-link HD-SDI, SMPTE 352 packets must be inserted into both fields of the interlaced transport, so this input must be High. This input must be valid during the entire HANC interval.
a_y_in	In	10	This is the data stream A Y input. The data on this port depends on the SDI mode: • SD-SDI: The multiplexed Y/C data stream enters the module on this port. • HD-SDI: The Y data stream enters the module on this port. • 3G-SDI level A: Data stream 1, as defined by SMPTE 425, enters the module on this port. • Dual-link HD-SDI or 3G-SDI level B transporting dual-link HD-SDI: The Y data stream of link A enters the module on this port. • 3G-SDI level B transporting dual HD-SDI signals: The Y data stream of HD-SDI signal 1 enters the module on this port.
a_c_in	In	10	This is the data stream A C input. The data on this port depends on the SDI mode: • HD-SDI and 3G-SDI level A: The C data stream enters the module on this port. • Dual-link HD-SDI or 3G-SDI level B transporting dual-link HD-SDI: The C data stream of link A enters the module on this port. • 3G-SDI level B transporting dual HD-SDI signals: The C data stream of HD-SDI signal 1 enters the module on this port.
b_y_in	In	10	This is the data stream B Y input: The data stream on this port depends on the SDI mode: • Dual-link HD-SDI or 3G-SDI level B carrying dual-link HD-SDI: The Y data stream of link B enters the module on this port. • 3G-SDI level B carrying dual HD-SDI signals: The Y data stream of HD-SDI signal 2 enters the module on this port. • For other SDI modes, this input port is unused.

Table 6: **I/O Ports of the `triple_sdi_vpid_insert` Module *(Cont'd)***

Port Name	I/O	Width	Description
b_c_in	In	10	This is the data stream B C input: The data stream on this port depends on the SDI mode: • Dual-link HD-SDI or 3G-SDI level B carrying dual-link HD-SDI: the C data stream of link B enters the module on this port. • 3G-SDI level B carrying dual HD-SDI signals: The C data stream of HD-SDI signal 2 enters the module on this port. • For other SDI modes, this input port is unused.
ds1a_out	Out	10	Data stream 1 of link A is output on this port.
ds2a_out	Out	10	Data stream 2 of link A is output on this port.
ds1b_out	Out	10	Data stream 1 of link B is output on this port. This port is only used in 3G-SDI level B and dual-link HD-SDI modes.
ds2b_out	Out	10	Data stream 2 of link B is output on this port. This port is only used in 3G-SDI level B and dual-link HD-SDI modes.
eav_out	Out	1	This output is High when the XYZ word of an EAV is output on the data stream outputs.
sav_out	Out	1	This output is High when the XYZ word of an SAV is output on the data stream outputs.
out_mode	Out	2	This output port must be connected to the mode input port of the `triple_sdi_tx_output_20b` module.

Triple-Rate SDI Output Module

Table 7 describes the ports of the `triple_sdi_tx_output_20b` module. This module is the back end of the triple-rate SDI transmitter datapath. It takes in one, two, or four data streams, depending on the SDI mode, from the `triple_sdi_vpid_insert` module, optionally generates and inserts EDH packets (in SD-SDI mode) or CRC and LN words (in HD-SDI and 3G-SDI modes), and then scrambles the data and outputs a single 20-bit data stream to the `gtp_interface_tx` module.

Table 7: **I/O Ports of the `triple_sdi_tx_output_20b` Module**

Port Name	I/O	Width	Description
clk	In	1	This clock input must be driven by the pipe_clk. It must have a frequency of 74.25 MHz or 74.25/1.001 MHz for HD-SDI, 148.5 MHz or 148.5/1.001 MHz for 3G-SDI, and 148.5 MHz for SD-SDI modes.
ce	In	2	The clock enable must be asserted at a 27 MHz rate for SD-SDI with a mandatory 5/6/5/6 clock cycle cadence. For other SDI modes, this clock enable must always be High. The module requires two identical copies of this clock enable signal on its ce port. For most applications, both bits of the ce port can be driven with the same clock enable signal. ce[0] controls everything except the EDH processor, and ce[1] controls just the EDH processor. If EDH packet insertion or updating is not required, ce[1] can be tied Low and the EDH processor is optimized out of the design by the synthesis tool.
din_rdy	In	1	For SD-SDI, HD-SDI, and 3G-SDI level A formats, this input must be kept High at all times. For 3G-SDI level B format, this input must be asserted every other clock cycle.
rst	In	1	This asynchronous reset input resets the module when High. The falling edge of this reset signal must meet the reset recovery time of all flip-flops relative to the next rising edge of clk.
mode	In	2	This input port selects the mode of operation of this module. Normally, the out_mode port of the `triple_sdi_vpid_insert` module is connected to this input port. This port is encoded as follows: • `00` = HD-SDI, 3G-SDI level A, and dual-link HD-SDI • `01` = SD-SDI • `10` = 3G-SDI level B • `11` = Invalid

Table 7: **I/O Ports of the `triple_sdi_tx_output_20b` Module *(Cont'd)***

Port Name	I/O	Width	Description
ds1a	In	10	This input is driven by the ds1a_out port of the `triple_sdi_vpid_insert` module. The data on this input port depends on the SDI mode: • SD-SDI: Multiplexed Y/C data stream for SD-SDI • HD-SDI: Dual-link HD-SDI and 3G-SDI level A: Y data stream • 3G-SDI level B: Y data stream of link A or Y data stream of HD-SDI signal 1
ds2a	In	10	This input is driven by the ds2a_out port of the `triple_sdi_vpid_insert` module. The data on this input port depends on the SDI mode: • SD-SDI: Not used • HD-SDI: Dual-link HD-SDI and 3G-SDI level A: C data stream • 3G-SDI level B: C data stream of link A or C data stream of HD-SDI signal 1
ds1b	In	10	This input is driven by the ds1b_out port of the `triple_sdi_vpid_insert` module. It carries the Y data stream of link B for 3G-SDI level B or of HD-SDI signal 2. It is unused in all other modes.
ds2b	In	10	This input is driven by the ds2b_out port of the `triple_sdi_vpid_insert` module. It carries the C data stream of link B for 3G-SDI level B or of HD-SDI signal 2. It is unused in all other modes.
insert_crc	In	1	When this input is High, CRC values are calculated and inserted into all data streams when running in HD-SDI and 3G-SDI modes. This input is ignored in SD-SDI mode.
insert_ln	In	1	When this input is High, LN (line number) words are inserted after the EAVs in all data streams when running in HD-SDI and 3G-SDI modes. This input is ignored in SD-SDI mode.
insert_edh	In	1	When this input is High, EDH packets are generated and inserted into the SD-SDI data stream. This input is ignored when running in HD-SDI and 3G-SDI modes.
ln_a	In	1	This is the primary line number input port. A value is required on this port for HD-SDI and 3G-SDI modes when the insert_ln port is High. The line number input must be stable by the XYZ word of the EAV, and must remain stable for at least two sample times. For HD-SDI, dual-link HD-SDI, and level A 3G-SDI modes, this is the only line number input port used, and the value on this port is inserted into the ds1a and ds2a streams. For level B 3G-SDI mode, a second line number port, ln_b, is also used. The ln_a input port is not used for SD-SDI mode. For those video formats where the transport line number is not the same as the picture line number, the value on this port must be the transport line number.
ln_b	In	11	When transmitting in level B 3G-SDI format, and when the insert_ln signal is High, the line number on this port is inserted into the ds1b and ds2b data streams. This allows the two independent HD-SDI signals carried by level B to be vertically unsynchronized. This input port is not used for SD-SDI, HD-SDI, or 3G-SDI level A formats. This input has the same timing and other requirements as described for ln_a.
eav	In	1	This input must be High when the XYZ word of each EAV enters the module on the data stream inputs. This input is not required for SD-SDI mode. This port is typically driven by the eav_out port of the `triple_sdi_vpid_insert` module.
sav	In	1	This input must be High when the XYZ word of each SAV enters the module on the data stream inputs. This input is not required for SD-SDI mode. This port is typically driven by the sav_out port of the `triple_sdi_vpid_insert` module.
txdata	Out	20	The scrambled SDI data stream is output on this port. The data should be directly connected to the pipe_data port of the `gtp_interface_pll` module. For SD-SDI mode, this data is 5.5X-oversampled data running at 74.25 MHz. For HD-SDI mode, it runs at the native data rate of 74.25 MHz. For 3G-SDI mode, the native data rate is 148.5 MHz.
ce_align_err	Out	1	The SD-SDI bit replication logic requires that the ce input signal has an exact 5/6/5/6 clock cycle cadence. If the ce signal varies from this cadence, the bit replication logic underflows or overflows, and the ce_align_err output is asserted High until the cadence returns to normal.

GTP TX Interface Module

A block diagram of the `gtp_interface_pll` module is shown in Figure 9. This module abstracts the clock creation and buffering so that alternate clock creation and distribution schemes can be employed by the user, if desired. The 20-bit data is routed through this module to facilitate synchronization between clock domains, if required for these alternate clocking schemes.

The module creates the two synchronized user clocks required for interfacing to the GTP transceiver: usrclk, and usrclk2. The pipe_clk signal is identical to usrclk2. The input clock is the TX output clock from the GTP transceiver, which is based on the reference clock.

Table 8 describes the inputs and outputs of the `gtp_interface_pll` module. Figure 10 shows the `gtp_interface_pll` module in the context of a triple-rate SDI transmitter. This module is also used for the GTP receiver, but it requires a separate instance, and the usage and some clock frequencies are different (see GTP RX Interface Module, page 9).

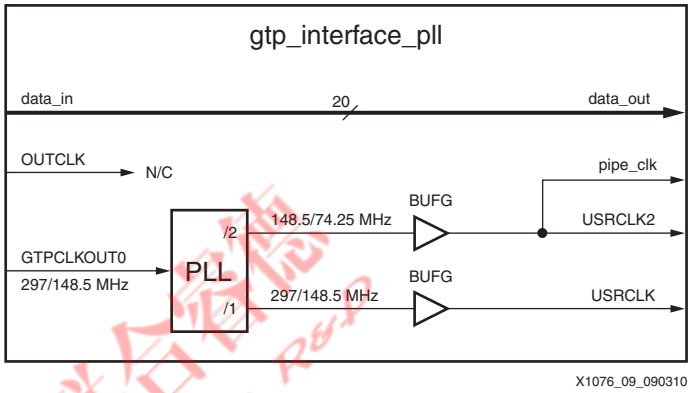


Figure 9: **Block Diagram of the `gtp_interface_pll` Module**

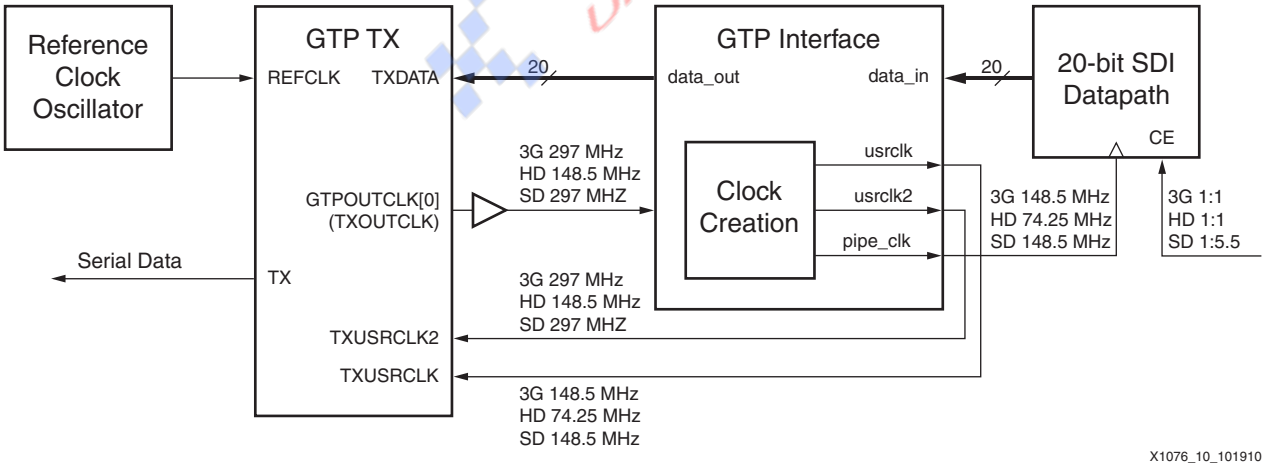


Figure 10: **GTP Interface Module in TX Context**

Table 8: **I/O Ports of the gtp_interface_pll Module**

Port Name	I/O	Width	Description
outclk	In	1	Not used. Output clocks using dedicated clock routing come through gtpclkout.
gtpoutclk	In	1	PLL input clock on dedicated clock routing. Nominally, 297 MHz for 3G-SDI and SD-SDI, and 148.5 MHz for HD-SDI. This should be connected to the GTPOUTCLK[1] port of the GTP transceiver through a BUFIO2 buffer.

Table 8: **I/O Ports of the gtp_interface_pll Module *(Cont'd)***

Port Name	I/O	Width	Description
pll_reset_in	In	1	Asynchronous PLL reset. 1 = Reset PLL 0 = Do not reset PLL
data_in	In	20	20-bit input data. This should be connected to the txdata port of the triple_sdi_tx_output_20b module.
usrclk	Out	1	Output user clock at 1X the input clock rate. Nominally, 297 MHz for 3G-SDI and SD-SDI, and 148.5 MHz for HD-SDI. This is used to drive the USRCLK input of the GTP transceiver.
usrclk2	Out	1	Output user clock at 1/2x the input clock rate. Nominally, 148.5 MHz for 3G-SDI and SD-SDI, and 74.25 MHz for HD-SDI. This is used to drive the USRCLK2 input of the GTP transceiver.
pipe_clk	Out	1	Pipeline clock for downstream pixel processing. This must be at the same frequency as usrclk2. In the reference design it is the same clock.
data_out	Out	20	20-bit output data. This should be connected to the TXDATA port of the GTP transceiver.
pll_locked_out	Out	1	Locked flag from the PLL. 1 = Locked 0 = Not locked

Operation of the Triple-Rate SDI Transmitter in the Various SDI Modes

The SDI mode (SD-SDI, HD-SDI, 3G-SDI level A or 3G-SDI level B) of the triple-rate SDI transmitter is determined by the mode and level inputs of the transmitter datapath modules. The clock frequency requirements and the number of data streams expected by the triple-rate SDI transmitter datapath depend on the SDI mode.

SD-SDI Transmitter Operation

When operating in SD-SDI mode, the pipe_clk from gtp_interface_tx module has a frequency of 148.5 MHz. The interleaved Y/C data stream is connected to the a_y_in port of the triple_sdi_vpid_insert module. The clock enable inputs of both modules (triple_sdi_vpid_insert and triple_sdi_tx_output_20b) must be asserted at a 27 MHz rate, as shown in Figure 11. Thus they must be asserted with a 5/6/5/6 clock cycle cadence. Any other cadence of the ce signals causes the SD-SDI bit replication logic to underflow or overflow. The din_rdy ports of both modules must always be held High in SD-SDI mode.

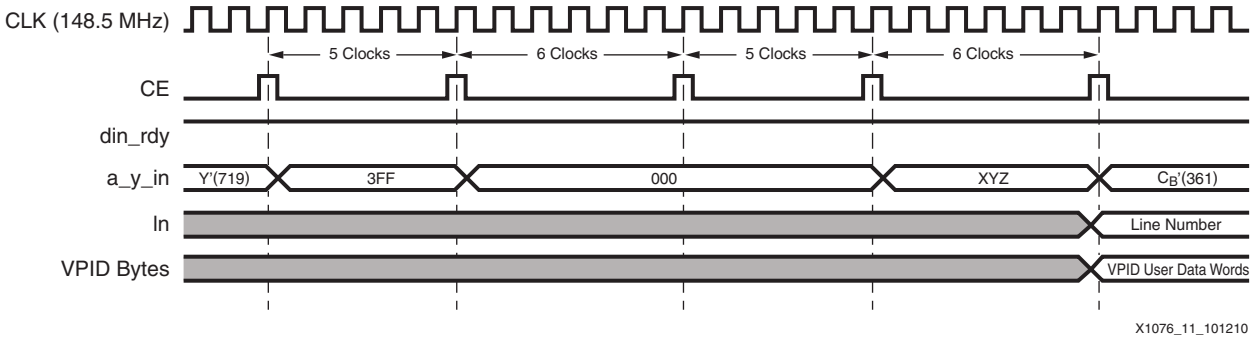


Figure 11: **SD-SDI TX Timing**

Notes relevant to Figure 11:

1. clk, ce, din_rdy, and ln are connected to the corresponding ports of both the triple_sdi_vpid_insert and triple_sdi_tx_output_20b modules. All inputs except ce have an entire sample time in which to become stable. Data is only sampled on

the inputs on the rising edge of clk when ce is High. The sample time is the time between rising edges of clk when ce is High. As shown in Figure 11, the ce signal must always have a 5/6/5/6 clock cycle cadence. Any other cadence underflows or overflows the SDI bit replication logic. Although Figure 11 shows that the first word of the EAV is a 5-clock cycle sample, there is no such relationship required. The first word of the EAV could instead be a 6-clock cycle sample.

2. In SD-SDI mode, the line number value on the ln input port is only used by the `triple_sdi_vpid_insert` module, must be stable starting with the XYZ word of the EAV, and must remain valid through the entire HANC interval.
3. The VPID input bytes on the input of the `triple_sdi_vpid_insert` module must be stable beginning with the XYZ word of the EAV, and must remain stable during the entire HANC interval on lines where SMPTE 352 packets are inserted.

The `triple_sdi_vpid_insert` module optionally inserts SMPTE 352 VPID packets and outputs the modified data stream to the triple_`sdi_tx_output_20b` module on the ds1a connection.

The `triple_sdi_tx_output_20b` module inserts EDH packets, if insert_edh is High, then the module scrambles the data and replicates each scrambled bit 11 times. The scrambled and replicated data is output on the txdata port as 20-bit words at 148.5 MHz. This data goes through the `gtp_interface_pll` module to the TXDATA port. The GTP transmitter serializes the data for transmissions over the serial interface.

Line number values input on the ln_a port of the `triple_sdi_vpid_insert` module are only required if SMPTE 352 packets are inserted in SD-SDI mode. Line number values are not required to be input on the ln_a port of the `triple_sdi_tx_output_20b` module in SD-SDI mode.

HD-SDI Transmitter Operation

When operating in HD-SDI mode, the pipe_clk output from the `gtp_interface_tx` module has a frequency of 74.25 MHz or 74.25/1.001 MHz. The ce and din_rdy inputs to the datapath modules must always be High. Figure 12 shows the timing of the input signals for HD-SDI mode.

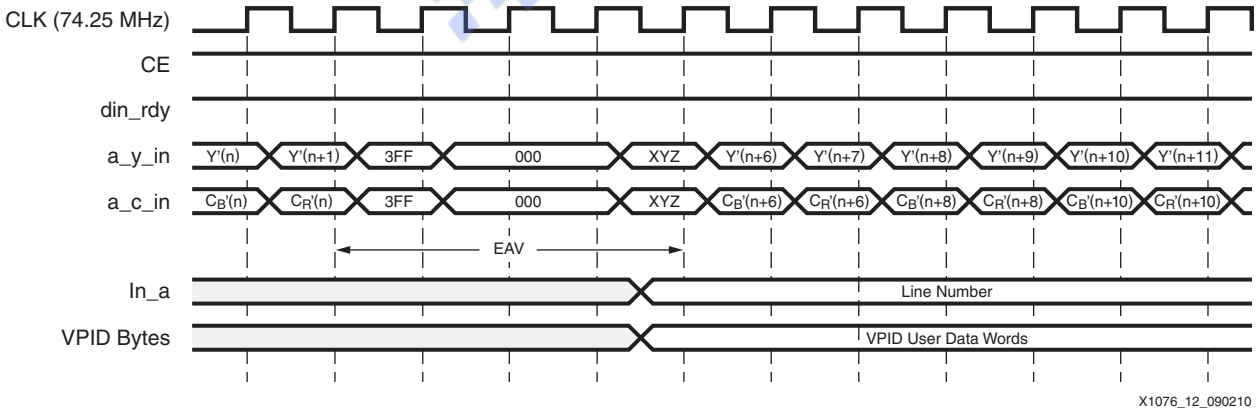


Figure 12: **HD-SDI TX Timing**

Notes relevant to Figure 12:

1. In HD-SDI mode, the line number value on the ln_a input port is used by both the `triple_sdi_vpid_insert` and the `triple_sdi_tx_output_20b` modules and must be stable starting with the XYZ word of the EAV, and must remain valid through the entire HANC interval.

2. The VPID input bytes on the input of the `triple_sdi_vpid_insert` module must be stable beginning with the XYZ word of the EAV, and must remain stable during the entire HANC interval on lines where SMPTE 352 packets are inserted.

Data enters the `triple_sdi_vpid_insert` module as two 10-bit data streams with the Y data stream on the a_y_in port and the C data stream on the a_c_in port. The input data rate is 74.25 MHz or 74.25/1.001 MHz. The `triple_sdi_vpid_insert` module optionally inserts SMPTE 352 VPID packets into the Y data stream before outputting the two data streams on the ds1a and ds2a ports to the `triple_sdi_tx_output_20b` module.

The `triple_sdi_tx_output_20b` module inserts line numbers into both data streams immediately after the EAV, if insert_ln is High. It calculates and inserts CRC values immediately after the line numbers if insert_crc is High. It scrambles the data streams and outputs one 20-bit data stream running at 74.25 MHz or 74.25/1.001 MHz on the txdata output port. This data passes through the `gtp_interface_pll` module and through to the GTP TXDATA port. The GTP transmitter serializes the data for transmissions over the serial interface.

Line numbers must be supplied on the ln_a port of the `triple_sdi_vpid_insert` module if SMPTE 352 packet insertion is enabled in HD-SDI mode. Line numbers must be supplied on the ln_a port of the `triple_sdi_tx_output_20b` module if line number insertion is enabled.

3G-SDI Transmitter Operation

When operating in 3G-SDI mode, the datapath modules can either run in level A mode or level B mode as selected by the `triple_sdi_vpid_insert` module's level input (level A = Low, level B = High). The `triple_sdi_vpid_insert` module informs the `triple_sdi_tx_output_20b` module which level is selected through the out_mode port.

Level A Mode

In 3G-SDI level A mode, the `triple_sdi_vpid_insert` module requires two 10-bit data streams on the a_y_in (data stream 1) and a_c_in (data stream 2) input ports. The clock frequency is 148.5 MHz or 148.5/1.001 MHz. The clock enable and din_rdy inputs to the datapath modules must be High. For 1080p 50 Hz, 59.94 Hz, and 60 Hz, the Y component data stream is input on the a_y_in port and the C component data stream is input on the a_y_in port. For all other 3G-SDI video formats, the video must be mapped to the two SDI data streams before they go into the `triple_sdi_vpid_insert` module.

Figure 13 shows the timing of the inputs signals for 3G-SDI level A mode.

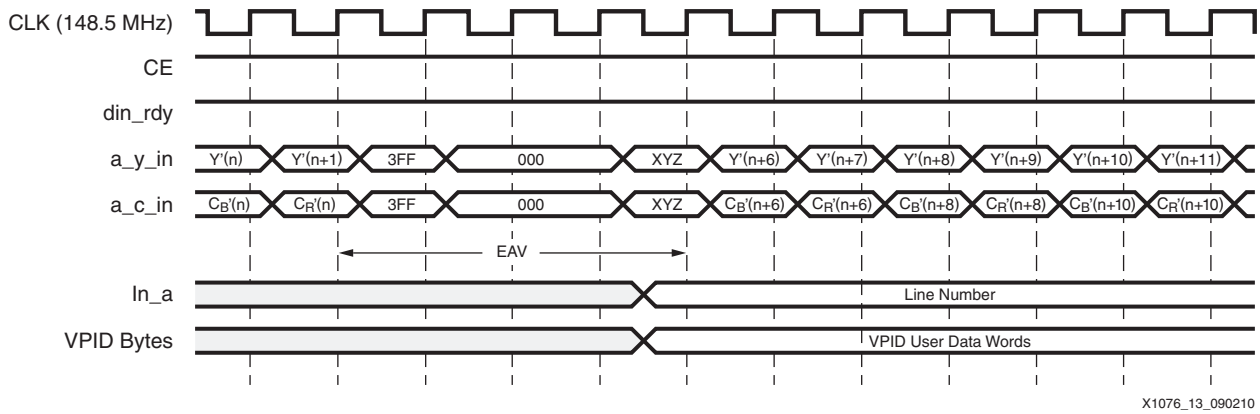


Figure 13: **3G-SDI Level A TX Timing**

Notes relevant to Figure 13:

1. In 3G-SDI level A mode, the line number value on the ln input port is used by both the `triple_sdi_vpid_insert` and the `triple_sdi_tx_output_20b` modules. It must

be stable starting with the XYZ word of the EAV, and must remain valid through the entire HANC interval.

2. The VPID input bytes on the input of the `triple_sdi_vpid_insert` module must be stable beginning with the XYZ word of the EAV, and must remain stable during the entire HANC interval on lines where SMPTE 352 packets are inserted.

The `triple_sdi_vpid_insert` module inserts SMPTE 352 packets into both data streams, before outputting the data streams on the ds1a and ds2a output ports to the `triple_sdi_tx_output_20b` module. The `triple_sdi_tx_output_20b` module inserts line numbers into both data streams immediately after the EAV, if insert_ln is High. It calculates and inserts CRC values immediately after the line numbers if insert_crc is High. It scrambles the data streams and outputs one 20-bit data stream running at 148.5 MHz on the txdata output port. This data goes through the `gtp_interface_pll` module to the TXDATA port of the GTP transmitter. The GTP transmitter serializes the data for transmissions over the serial interface.

Line numbers are required on the ln_a port of the `triple_sdi_vpid_insert` module if SMPTE 352 packet insertion is enabled. However, SMPTE 352 packets are required in 3G-SDI mode. Line numbers are required on the ln_a port of the `triple_sdi_tx_output_20b` module if line number insertion is enabled.

Level B Mode

When running in SG-SDI level B mode, the `triple_sdi_vpid_insert` module requires four 10-bit data streams on its a_y_in (Y channel of link A), a_c_in (C channel of link A), b_y_in (Y channel of link B), and b_c_in (C channel of link B) ports. These four data streams can be SMPTE 372 dual-link HD-SDI data streams, or they can be two independent HD-SDI streams that are to be combined onto a single 3G-SDI level B link. The clock frequency is 148.5 MHz or 148.5/1.001 MHz, therefore, the datapath modules must be clocked every other clock cycle in 3G-SDI level B mode, as shown in Figure 14. For the `triple_sdi_vpid_insert` module, either ce or din_rdy can be used to enable this module every other clock cycle. However, for the `triple_sdi_output_20b` module, ce cannot be used and din_rdy must be used. Thus, in 3G-SDI B mode, the correct way to control these modules is to keep ce High and toggle the din_rdy signal every other clock cycle, as shown in Figure 14.

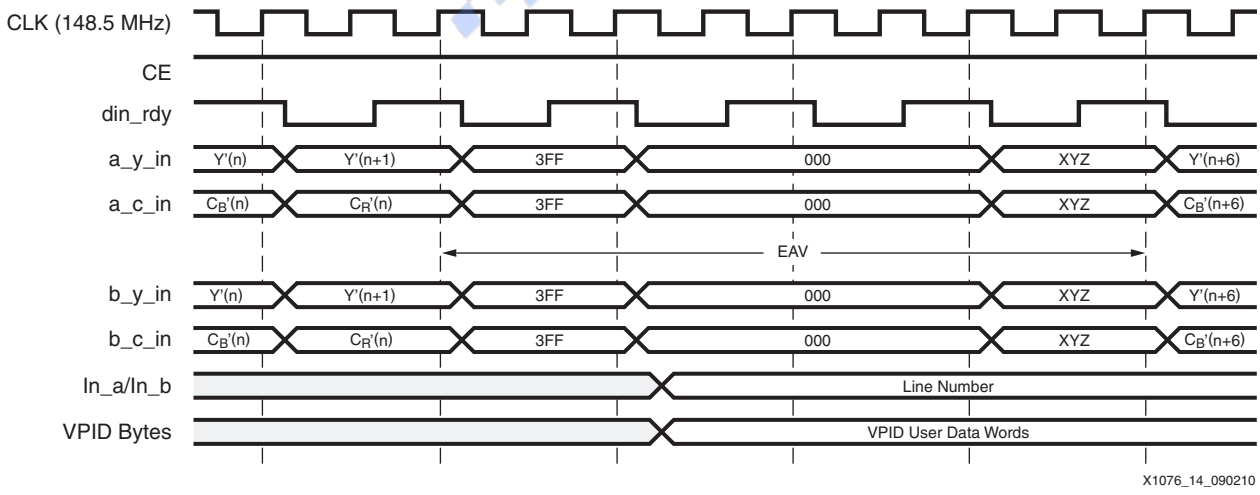


Figure 14: **3G-SDI Level B TX Timing**

Notes relevant to Figure 14:

1. In 3G-SDI mode, the line number value on the ln_a and ln_b input ports is used by both the `triple_sdi_vpid_insert` and the `triple_sdi_tx_output_20b` modules. It must be stable starting with the XYZ word of the EAV, and must remain valid through the entire HANC interval.

2. The VPID input bytes on the input of the `triple_sdi_vpid_insert` module must be stable beginning with the XYZ word of the EAV, and must remain stable during the entire HANC interval on lines where SMPTE 352 packets are inserted.

The `triple_sdi_vpid_insert` module inserts SMPTE 352 VPID packets in the Y data streams of both link A and link B. The four data streams are output to the `triple_sdi_tx_output_20b` module on the ds1a, ds2a, ds1b, and ds2b ports.

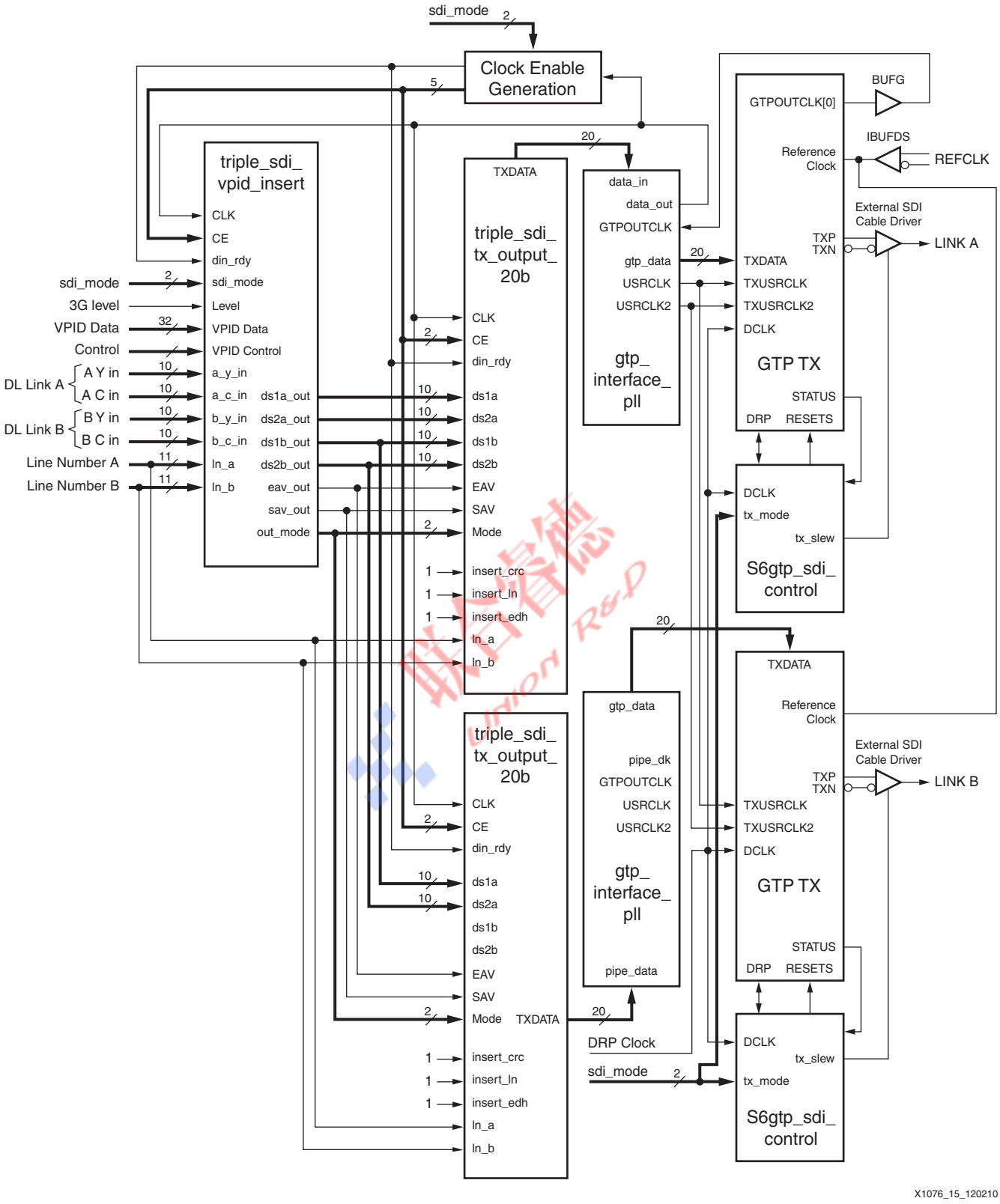
If insert_ln is High, the `triple_sdi_tx_output_20b` module inserts line numbers into all four data streams. If insert_crc is High, it calculates and inserts CRC values immediately after the line numbers in all four data streams. The data streams are then interleaved to produce a single 20-bit data stream running at 148.5 MHz or 148.5/1.001 MHz on the txdata output port. The GTP transmitter serializes the data for transmissions over the serial interface.

If SMPTE 352 packets are to be inserted, line numbers are required on both the ln_a (for link A) and ln_b (for link B) input ports of the `triple_sdi_vpid_insert` module. SMPTE 352 packets are mandatory for 3G-SDI. If line number insertion is enabled, line numbers are also required on the ln_a and ln_b ports of the `triple_sdi_tx_output_20b` module. Two different line number ports are provided for the case where two independent HD-SDI signals are carried by 3G-SDI level B. When dual-link HD-SDI is carried by 3G-SDI level B, ln_a and ln_b must be driven with identical line numbers. And, for video formats where the picture line number and the transport line number are not the same, the line numbers must always be transport line numbers.

Dual-Link HD-SDI Transmitter Operation

The triple-rate SDI transmitter does not contain the formatting logic to map the various video formats into SMPTE 372 dual-link HD-SDI streams. However, formatted SMPTE 372 dual-link HD-SDI streams can be transmitted by a pair of triple-rate SDI transmitters. There are several ways to build a dual-link HD-SDI transmitter, capable of transmitting preformatted dual-link HD-SDI streams, using the modules that make up the Spartan-6 FPGA triple-rate SDI transmitter.

The two transmitters are always paired as a dual-link HD-SDI pair. In this method, shown in Figure 15, the Link A transmitter can be used in any SDI mode, but the Link B transmitter is only used in dual-link HD-SDI mode. A single `triple_sdi_vpid_insert` module inserts SMPTE 352 packets into the Y stream of both HD-SDI links of the dual-link pair. These four data streams out of the VPID inserter are connected to two `triple_sdi_tx_output_20b` modules with the link A data streams going to one output module and the link B data streams going to the other, as shown in Figure 15. The link B data streams are also routed to the Link A output module because they are required for 3G-SDI level B operation.



X1076_15_120210

Figure 15: **Example Dual-Link HD-SDI TX (Transmitters Always Paired)**

The input timing is identical to normal HD-SDI mode as shown in Figure 10. The ln_b line number port of the `triple_sdi_vpid_insert` module(s) is ignored in dual-link HD-SDI mode. The ln_b port is only used in 3G-SDI level B mode.

As required by the SMPTE 372 standard, the two HD-SDI links must be uniquely identified using bit 6 of byte 4 of the SMPTE 352 packets. This bit must be `0` in link A and `1` in link B. If a single VPID insert module is used as shown in Figure 15, the `triple_sdi_vpid_insert` module has separate input ports for byte 4 for the two links just for this purpose.

Summary of Triple-Rate TX Modes

The input data rates and connections for all SDI modes are summarized in Table 9.

Table 9: **Triple-Rate SDI TX Modes**

SDI Mode	Level	TXOUTCLK	ce	din_rdy	Input Data Rate	a_y_in	a_c_in	b_y_in	b_c_in
HD-SDI and Dual-Link HD-SDI	X	74.25 MHz or 74.25/1.001 MHz	High	High	74.25 MHz or 74.25/1.001 MHz	Y	C		
SD-SDI	X	148.5 MHz	5/6/5/6 Cadence	High	27 MHz	Y/C			
3G-SDI A	`0`	148.5 MHz or 148.5/1.001 MHz	High	High	148.5 MHz or 148.5/1.001 MHz	Data Stream 1	Data Stream 2		
3G-SDI B	`1`	148.5 MHz or 148.5/1.001 MHz	High	Asserted every other clock cycle	74.25 MHz or 74.25/1.001 MHz	Link A Y	Link A C	Link B Y	Link B C

Triple-Rate TX Details

This section provides some additional details about the operation of the Spartan-6 FPGA triple-rate SDI transmitter reference design.

SMPTE 352 Packet Insertion

The `triple_sdi_vpid_insert` module can insert SMPTE 352 video payload ID packets into SD-SDI, HD-SDI, dual-link HD-SDI, and 3G-SDI (both level A and level B) streams. The dual-link HD-SDI and 3G-SDI standards require SMPTE 352 packets in the data streams. The packets are optional in SD-SDI and HD-SDI data streams.

The `triple_sdi_vpid_insert` module is a wrapper around a pair of `SMPTE352_vpid_insert` modules. The behavior of this module is the same as the behavior of the underlying `SMPTE352_vpid_insert` module. Refer to Chapter 26 of *Audio/Video Connectivity Solutions for Virtex-5 FPGAs* [Ref 3] to review the behavior of this module. This chapter also describes the required values for the user data words of the SMPTE 352 packets.

SMPTE 352 packets are inserted on one designated video line in each frame if the transport is progressive, and one designated video line in each field if the transport is interlaced. The designated video lines that carry SMPTE 352 packets vary depending on the SDI mode and the video format. They are detailed in Chapter 26 of *Audio/Video Connectivity Solutions for Virtex-5 FPGAs* [Ref 3]. The line_f1 input port determines the line in the progressive frame or in the first interlaced field where the module inserts a SMPTE 352 packet. The line_f2 input port determines the line in the second interlaced field where the module inserts the SMPTE 352 packet. The line_f2_en input determines whether packets are inserted in the line specified by line_f2. The line_f2_en input must be Low for a progressive transport and High for an interlaced transport.

The line_f2_en input to the `triple_sdi_vpid_insert` module must be controlled correctly. This input must be High for interlaced video and Low for progressive video. However, this input must be controlled based on whether the transport is interlaced or progressive, not the picture. The 1080p 50 Hz and 60 Hz 4:2:2 10-bit video formats, when carried by dual-link HD-SDI or 3G-SDI level B (but not 3G-SDI level A) are transported in an interlaced manner, even though the picture is progressive. The same is true for progressive segmented frame transport. It is

essential that the SMPTE 352 packets are inserted into both fields of the transport data streams for these formats. Some receiving equipment fail to lock properly if the SMPTE 352 packets are only present in one field rather than in both fields of interlaced transports.

Dual-link HD-SDI data streams coming from a dual-link SDI receiver can already have SMPTE 352 packets in the Y data streams of each link, because SMPTE 372 mandates these packets. If the packets are present, the first user data word should be hex 87 indicating that the data streams are carried on a dual-link HD-SDI interface. If these dual-link data streams are to be retransmitted on a 3G-SDI level B interface, the first user data word of the packet must be replaced with a value of hex 8A, indicating a SMPTE 372 signal carried on a 3G-SDI level B interface. Thus the SMPTE 352 packets must be modified when sending the dual-link HD-SDI streams on a 3G-SDI level B interface. Only the first byte of the VPID data must be modified. The values of the other bytes do not need to change. However, they must be captured first and applied to the VPID byte inputs of the triple_sdi_vpid_insert module so that they are re-inserted when the packet is overwritten by the triple_sdi_vpid_insert module (as part of the process of updating the packet). This is because the inserter overwrites the entire packet, not just the first user data word.

In 3G-SDI level B mode, the two independent HD-SDI signals can carried by the 3G-SDI level B interface (called dual stream mode). These two HD-SDI signals can be vertically unsynchronized (not frame locked). If this is the case, then SMPTE 352 packets are inserted independently on the two HD-SDI signals carried by the 3G-SDI level B interface. The second line number input port on the triple_sdi_vpid_insert module allows two separate line numbers to be provided for level B, one for each HD-SDI signal. As there is only one set of VPID data inputs to the insertion module, the SMPTE 352 packets, with the exception of byte 4, are identical. Thus the two HD-SDI streams must be carrying identical video formats. This is normally the case because of the restrictions in 3G-SDI level B mode requiring the two HD-SDI streams have the same format. If an application requires insertion of different VPID packets into the two HD-SDI signals carried by a 3G-SDI level B signal, the triple_sdi_vpid_insert module must be modified to provide two completely independent sets of VPID input data ports.

Line Number Insertion

3G-SDI and HD-SDI both require line numbers in the two words that follow each EAV. If the insert_ln input is High, the triple_sdi_tx_output_20b module inserts those line numbers appropriately for HD-SDI and both levels of 3G-SDI (inserting them into all four data streams for level B 3G-SDI). The line numbers to be inserted are provided to the triple_sdi_tx_output_20b module on the ln_a and ln_b inputs. In some cases, it might not be necessary or desirable to overwrite line numbers that are already present in the data streams. In that case, the insert_ln input can be driven Low, and no line numbers are inserted. If the insert_ln input is hardwired Low in the source code, the line number insertion logic is optimized out of the design by the synthesis tool.

For dual-link HD-SDI data streams carried either by dual-link HD-SDI or 3G-SDI level B, if the video format is 1080p 50 Hz, 59.94 Hz, or 60 Hz, the line numbers are required to be transport line numbers, not video line numbers.

For HD-SDI, dual-link HD-SDI, and 3G-SDI level A formats, only the line number on ln_a is used. For 3G-SDI level B, the line number on ln_a is inserted into the Y and C data streams of link A and the line number on ln_b is inserted into the Y and C data streams of link B (again, only if insert_ln is asserted). For dual-link HD-SDI, two triple_sdi_tx_output_20b modules are used, one for each link. Each triple_sdi_tx_output_20b module processes its HD-SDI data streams through the ds1a and ds2a input ports, just as for normal HD-SDI mode. The ln_b input port is not used in dual-link HD-SDI mode.

CRC Generation and Insertion

Both 3G-SDI and HD-SDI modes require CRC values in the two words that follow the line numbers after the EAV. If the insert_crc input is High, the triple_sdi_tx_output_20b module calculates and inserts CRC values for each line for HD-SDI and both levels of 3G-SDI

(inserting them into all four data streams for level B 3G-SDI). In some cases, it might not be necessary or desirable to overwrite the CRC values already present in the data stream. In that case, the insert_crc input can be driven Low, and no CRC values are inserted. If the insert_crc input is hardwired Low, the CRC generation and insertion logic is optimized out of the design by the synthesis tool.

EDH Generation and Insertion

EDH packets are optional, but usually present, in SD-SDI. They are never used for HD-SDI and 3G-SDI. The EDH packets contain CRC values that can be used to detect errors in the SD-SDI data stream. When running in SD-SDI mode, the `triple_sdi_tx_output_20b` module generates and inserts EDH packets when insert_edh is High. If insert_edh is hardwired Low, the synthesis tool optimizes the EDH generator out of the design. Xilinx has two different EDH processor designs. These are documented in Chapters 6 and 7 of *Audio/Video Connectivity Solutions for Virtex-II Pro and Virtex-4 FPGAs* [Ref 1]. Although originally written for Virtex-II Pro FPGAs, both of these EDH processors can be used in the Spartan-6 FPGA triple-rate SDI reference design without any modifications. The EDH processor described in Chapter 6 is used, by default, in the `triple_sdi_tx_output_20b` module, but can be replaced with the EDH processor described in Chapter 7 by modifying the `triple_sdi_tx_output_20b` module.

Ancillary Data Insertion

Ancillary data packets can be inserted into the data streams prior to the `triple_sdi_vpid_insert` module or between the `triple_sdi_vpid_insert` module and the `triple_sdi_tx_output_20b` module. Because the SMPTE 352 packets inserted by the `triple_sdi_vpid_insert` module must go, according to the SMPTE 352 standard, at the beginning of the HANC space (immediately after the CRC words following the EAV), and ancillary data packets can get inserted on the same lines as the SMPTE 352 packets, it is better to insert the ancillary data packets after the `triple_sdi_vpid_insert` module, unless space is reserved for the SMPTE 352 packets.

The `triple_sdi_tx_output_20b` module inserts EDH packets in SD-SDI mode, if enabled. Any ancillary data packets that conflict with the standard EDH packet locations are partially or completely overwritten when the EDH packets are inserted.

Reference Clocks and Reference Clock Switching

The Spartan-6 FPGA GTP transmitters require two different reference clock frequencies to transmit all of the SDI bit rates. These reference clock frequencies are typically 74.25 MHz and 74.25/1.001 MHz (or 2X or 4X times these two frequencies). In SD-SDI mode, it is important to always use 74.25 MHz (or a multiple) and never 74.25/1.001 MHz (or multiples).

There are two ways to switch between these two reference clock frequencies. The first is to provide a single external reference clock to the Spartan-6 FPGA GTP transmitter and change the frequency of the reference clock externally. A more flexible approach is to provide both reference clock frequencies to the Spartan-6 FPGA GTP transmitter and use the PLL reference clock multiplexer to dynamically switch between these two reference clock frequencies. The TXPLLREFSELDY input port on the GTP transceiver wrapper dynamically controls the reference clock multiplexer.

The GTP transmitter must always be reset after the reference clock frequency is changed. This ensures that the PMA PLL locks correctly to the new reference clock frequency. Regardless of whether the reference clock is changed externally or by changing the reference clock multiplexer, the GTPRESET input of the GTP transmitter must be briefly asserted High to reset the GTP transmitter after the reference clock frequency has changed. Unlike many other resets associated with the GTP transmitter, this reset is not automatically handled by the `s6gtp_sdi_control` module because it does not know when the reference clock frequency changes. The `s6gtp_sdi_control` module has a gtpreset port that must be connected to the GTPRESET port of the GTP transceiver wrapper, but this signal does not reset the GTP

transmitter when the reference clock frequency changes. However, a signal indicating a reference clock frequency change can be connected to the gtp_resetn_in port of the `s6gtp_sdi_control` module. The `s6gtp_sdi_control` module ORs this input with its other internal GTP transmitter reset conditions to drive the gtpreset port.

Use of the GCLK reference clocks are not recommended for SDI transmitter applications because of jitter concerns. If necessary, GCLK reference clocks can be used to supply a reference clock to SDI receivers because the additional jitter on the reference clock has less impact on the performance of the CDR than it does on the transmitter output jitter.

Electrical Interface

The GTP TX outputs are not directly compatible with the SDI electrical requirements. An SDI cable driver is required to interface the GTPTX output to the SDI connector. AC coupling is usually required between the GTP TX outputs and the SDI cable driver inputs. The value of these AC coupling capacitors is typically 4.7 µF.

SDI GTP Transceiver Control Module

The `s6gtp_sdi_control` module provides a number of functions primarily associated with control and monitoring of the Spartan-6 FPGA GTP transceiver. These control and monitoring functions are essential to reliable operation of the GTP transceiver when implementing SDI interfaces. Table 10 describes the ports of the `s6gtp_sdi_control` module, and, Table 11 lists the parameters of the `s6gtp_sdi_control` module.

One `s6gtp_sdi_control` module is required for each GTP transceiver dual tile. The module contains separate ports for each receiver and transmitter in each lane of the tile. If only the receiver section of the GTP transceiver is used, the transmitter ports do not need to be connected to the GTP transceiver. If only the transmitter section of the GTP transceiver is used, the receiver ports do not need to be connected to the GTP transceiver. However, the DRP ports must always be connected to the GTP transceiver because the DRP is used for both receive and transmit control functions.

Table 10: **I/O Ports of the `s6gtp_sdi_control` Module**

Port Name	I/O	Width	Description
DRP and Reset Ports			
rate_refclk	In	1	This must be driven by a stable, constant frequency reference clock. It is used as a comparison frequency by the rate detector. The minimum frequency is 27 MHz. Any higher frequency can be used, up to the point where timing cannot be met. This reference clock frequency does not have to be, in any way, related to the video clock frequency. It must, however, be a constant frequency, never changing even when the SDI mode changes. The rate of this clock is specified with the RATE_REFCLK_FREQ parameter.
rx0_usrclk	In	1	The user connects this port to the global or regional clock that is also connected to the USRCLK2 input of RX0 of the GTP transceiver tile.
rx1_usrclk	In	1	The user connects this port to the global or regional clock that is also connected to the USRCLK2 input of RX1 of the GTP transceiver tile.
tx0_usrclk	In	1	The user connects this port to the global or regional clock that is also connected to the USRCLK 2 input of TX0 of the GTP transceiver tile.
tx1_usrclk	In	1	The user connects this port to the global or regional clock that is also connected to the USRCLK2 input of TX1 of the GTP transceiver tile.

Table 10: **I/O Ports of the `s6gtp_sdi_control` Module *(Cont'd)***

Port Name	I/O	Width	Description
Mode Selection and Reference Clock Selection & Routing			
rx0_mode	In	2	Selects the operating mode for RX0: • `00`: HD-SDI • `01`: SD-SDI • `10`: 3G-SDI • `11`: Invalid
rx1_mode	In	2	Selects the operating mode for RX1: • `00`: HD-SDI • `01`: SD-SDI • `10`: 3G-SDI • `11`: Invalid
tx0_mode	In	2	Selects the operating mode for TX0: • `00`: HD-SDI • `01`: SD-SDI • `10`: 3G-SDI • `11`: Invalid
tx1_mode	In	2	Selects the operating mode for TX1: • `00`: HD-SDI • `01`: SD-SDI • `10`: 3G-SDI • `11`: Invalid
Cable Driver Slew Rate Control			
tx0_slew	Out	1	Slew rate control for the cable driver for TX0. This output is High for SD-SDI and Low for HD-SDI and 3G-SDI.
tx1_slew	Out	1	Slew rate control for the cable driver for TX1. This output is High for SD-SDI and Low for HD-SDI and 3G-SDI.
Receiver Rate Detection			
rx0_rate	Out	1	Indicates the bit rate being received by RX0. Refer to Table 8 for details.
rx1_rate	Out	1	Indicates the bit rate being received by RX1. Refer to Table 8 for details.
GTPRESET			
gtp_reset0_in	In	1	When pulsed High, this input initiates a long duration GTPRESET0 pulse to the GTP transceiver tile. The duration of the pulse is controlled by the GTP transceiver reset counter. If unused, this input must be Low.
gtp_reset1_in	In	1	When pulsed High, this input initiates a long duration GTPRESET1 pulse to the GTP transceiver tile. The duration of the pulse is controlled by the GTP transceiver reset counter. If unused, this input must be Low.
clocks_stable	In	1	This input can be used to generate a GTPRESET if the reference clock is not stable. The duration of the GTPRESET pulse to the GTP transceiver tile is equal to the duration that clocks_stable is Low. If unused, this input must be High.
gtpreset0	Out	1	The user connects this output port to the GTPRESET0 input port of the GTP transceiver tile.
gtpreset1	Out	1	The user connects this output port to the GTPRESET0 input port of the GTP transceiver tile.
RX0 Reset Signals			
rx0_pcs_reset	In	1	The application can drive this input High to initiate a reset of the PCS section of RX0.

Table 10: **I/O Ports of the `s6gtp_sdi_control` Module *(Cont'd)***

Port Name	I/O	Width	Description
rx0_cdr_reset	In	1	The application can drive this input High to initiate a reset of the CDR and PCS sections of RX0.
resetdone0	In	1	This input must be driven by the RESETDONE0 output of the GTP transceiver tile.
rxbufstatus0_b2	In	1	This input must be driven by bit 2 of the RXBUFSTATUS0 port of the GTP transceiver tile.
rx0_fabric_reset	Out	1	This output can be used to reset the FPGA logic associated with RX0.
rxreset0	Out	1	The user connects this output port to the RXRESET0 input port of the GTP transceiver tile.
rxbufreset0	Out	1	The user connects this output port to the RXBUFRESET0 input port of the GTP transceiver tile.
rxcdrreset0	Out	1	The user connects this output port to the RXCDRRESET0 input port of the GTP transceiver tile.
RX1 Reset Signals			
rx1_pcs_reset	In	1	The application can drive this input High to initiate a reset of the PCS section of RX1.
rx1_cdr_reset	In	1	The application can drive this input High to initiate a reset of the CDR and PCS sections of RX1.
resetdone1	In	1	This input must be driven by the RESETDONE1 output of the GTP transceiver tile.
rxbufstatus1_b2	In	1	This input must be driven by bit 2 of the RXBUFSTATUS1 port of the GTP transceiver tile.
rx1_fabric_reset	Out	1	This output can be used to reset the FPGA logic associated with RX1.
rxreset1	Out	1	The user connects this output port to the RXRESET1 input port of the GTP transceiver tile.
rxbufreset1	Out	1	The user connects this output port to the RXBUFRESET1 input port of the GTP transceiver tile.
rxcdrreset1	Out	1	The user connects this output port to the RXCDRRESET1 input port of the GTP transceiver tile.
TX0 Reset Signals			
tx0_reset	In	1	The application can drive this input High to initiate a PCS reset of the SDI transmitter TX0.
txbufstatus0_b1	In	1	The user connects this input port to bit 1 of the TXBUFSTATUS0 output port of the GTP transceiver tile.
tx0_fabric_reset	Out	1	This output can be used to reset the datapath of the SDI transmitter TX0.
txreset0	Out	1	This output must be connected to the TXRESET0 input port of the GTP transceiver tile.
TX1 Reset Signals			
tx1_reset	In	1	The application can drive this input High to initiate a PCS reset of the TX1.
txbufstatus1_b1	In	1	The user connects this input port to bit 1 of the TXBUFSTATUS1 output port of the GTP transceiver tile.
tx1_fabric_reset	Out	1	This output port can be used to reset the datapath of the SDI transmitter TX1.
txreset1	Out	1	This output port must be connected to the TXRESET1 input port of the GTP transceiver tile.

Table 10: **I/O Ports of the `s6gtp_sdi_control` Module *(Cont'd)***

Port Name	I/O	Width	Description
DRP Signals			
dclk	In	1	This input port must be connected to a free-running clock. The same clock signal must be connected to the DCLK input of the GTP transceiver tile. This clock input also clocks much of the reset logic, so it must be driven with a valid clock signal even if the DRP controller is not used. It is usually possible for dclk and rate_refclk to be sourced by the same clock signal.
drst	In	1	This is an asynchronous reset for the DRP controller. If used, it should also be connected to the DRST input of the GTP transceiver tile. A High pulse on this input causes the DRP controller to reinitialize the GTP transceiver attributes that it controls dynamically.
drdy	In	1	The user connects this port to the DRDY output port of the GTP transceiver tile.
drpo	In	16	The user connects this port to the DRPO output port of the GTP transceiver tile.
daddr	Out	7	The user connects this port to the DADDR input port of the GTP transceiver tile.
den	Out	1	The user connects this port to the DEN input port of the GTP transceiver tile.
di	Out	16	The user connects this port to the DI input port of the GTP transceiver tile.
dwe	Out	1	The user connects this port to the DWE input port of the GTP transceiver tile.

Table 11: **Parameters for the `s6gtp_sdi_control` Module**

Parameter	Type	Description
GTP_RESET_CNTR_SIZE	Integer	This parameter specifies the size (number of bits) of the GTP transceiver reset delay counter. This counter is reset to `0` when gtp_reset_in is `1`. When gtp_reset_in goes Low, the counter starts counting and the gtpreset output signal remains at `1` until the terminal count is reached. The terminal count occurs when the MSB of the counter becomes `1`. The reset counter is clocked by the dclk clock input, therefore the delay count is dclk frequency / (2^GTP_RESET_CNTR_SIZE).
CBL_DISCONNECT_RST_CNTR_SIZE	Integer	This parameter is obsolete because the cable disconnect reset is no longer needed. It is included for completeness. It used to specify the size of the cable disconnect reset duration counter in terms of DCLK cycles.
RATE_REFLK_FREQ	Integer	This parameter specifies the frequency of the dclk in Hz. This parameter is required for the receiver rate detection logic function to work correctly. The rate detection function is an essential part of the GTP receiver reset logic. If an SDI receiver is implemented, this parameter is required to be accurate even if the rx_m rate detection output of the module is not used.
PMA_RX_CFG_HD	25-bit vector	This value is used by the DRP controller to set the PMA_RX_CFG attribute of the GTP receiver when the receiver is placed in HD-SDI mode. The default value is normally used.
PMA_RX_CFG_SD	25-bit vector	This value is used by the DRP controller to set the PMA_RX_CFG attribute of the GTP receiver when the receiver is placed in SD-SDI mode. The default value is normally used.
PMA_RX_CFG_3G	25-bit vector	This value is used by the DRP controller to set the PMA_RX_CFG attribute of the GTP receiver when the receiver is placed in 3G-SDI mode. The default value is normally used.

Table 11: **Parameters for the `s6gtp_sdi_control` Module *(Cont'd)***

Parameter	Type	Description
PLL_RXDIVSEL_OUT_HD	2-bit vector	This value is used by the DRP controller to set the PLL output divider for the GTP receiver when the receiver is in HD-SDI mode: • `00`: divide by 1 • `01`: divide by 2 • `10`: divide by 4
PLL_RXDIVSEL_OUT_SD	2-bit vector	This value is used by the DRP controller to set the PLL output divider for the GTP receiver when the receiver is in SD-SDI mode: • `00`: divide by 1 • `01`: divide by 2 • `10`: divide by 4
PLL_RXDIVSEL_OUT _3G	2-bit vector	This value is used by the DRP controller to set the PLL output divider for the GTP receiver when the receiver is in 3G-SDI mode: • `00`: divide by 1 • `01`: divide by 2 • `10`: divide by 4
PLL_TXDIVSEL_OUT_HD	2-bit vector	This value is used by the DRP controller to set the PLL output divider for the GTP transmitter when the transmitter is in HD-SDI mode: • `00`: divide by 1 • `01`: divide by 2 • `10`: divide by 4
PLL_TXDIVSEL_OUT_SD	2-bit vector	This value is used by the DRP controller to set the PLL output divider for the GTP transmitter when the transmitter is in SD-SDI mode: • `00`: divide by 1 • `01`: divide by 2 • `10`: divide by 4
PLL_TXDIVSEL_OUT _3G	2-bit vector	This value is used by the DRP controller to set the PLL output divider for the GTP transmitter when the transmitter is in 3G-SDI mode: • `00`: divide by 1 • `01`: divide by 2 • `10`: divide by 4

DRP Port

The `s6gtp_sdi_control` module must always be connected to the DRP of the GTP transceiver when a receiver is used. It dynamically changes transceiver attributes though the DRP, for the receiver, when the SDI mode changes.

The dclk input port serves as the master clock for the module. It synchronizes the module with the DRP of the GTP transceiver, and clocks the internal logic of the module. The rate_refclk port is used as part of the receiver bit rate detection logic that generates the rx_rate output. Because of this, the module needs to know the frequency of rate_refclk. The module has a parameter/generic called RATE_REFCLK_FREQ that specifies the frequency of rate_refclk in Hz. This parameter/generic must be correctly specified when the `s6gtp_sdi_control` module is instantiated, otherwise the receiver rate detection function does not work correctly.

The other parameters of the `s6gtp_sdi_control` module are used by the module's DRP controller to dynamically switch the GTP transceiver between SDI modes. These parameters must all be correctly specified for the GTP transceiver to work correctly, although the three PMA_RX_CFG_XX parameters are best left in the default settings, which have been optimized for best receiver jitter tolerance. Similarly, the PLL_DIVSEL_OUT_XX parameters have been chosen for the particular mode and should not be changed.

Transmitter Functions

The `s6gtp_sdi_control` module provides several functions required for SDI transmitter operation. Many of the signals between the module and the GTP transceiver must be synchronous with the TX pipeclk, therefore this clock must be provided to the `s6gtp_sdi_control` module on the txusrclk port.

The SDI mode of the GTP transmitter is selected by the tx_mode input port. A change in the TX mode can be used to trigger a dynamic reconfiguration of the GTP transmitter in the reference design. The PLL post divider in the TX is changed based on tx_mode.

In certain circumstances, the GTP transmitter must be reset to ensure proper operation. The module asserts the txreset_out port to reset the GTP transmitter in these circumstances:

- When the txreset_in port is asserted High, the txreset_out port is driven High. The txreset_in port is typically driven by a signal that is pulsed High whenever the transmitter reference clock frequency is changed. The GTP transmitter must be reset when the frequency of the reference clock changes, either by changing the frequency of the external reference clock signal, or by changing reference clocks using the GTP transceiver PLL reference clock multiplexer. If the reference clock inputs are not stable as the FPGA emerges from configuration, the txreset_in port should be asserted briefly when the reference clocks become stable.
- If a GTP transmitter buffer error occurs, as indicated by the TXBUFSTATUS[1] bit, the GTP transmitter is reset.

The `s6gtp_sdi_control` module provides two outputs for convenience that are not required for proper operation of the SDI transmitter: tx_rate_change_done and tx_slew. The tx_rate_change_done output is asserted High upon completion of the dynamic TX rate change algorithm. When this output is Low, the GTP transmitter is not operating normally. The module also provides a tx_slew output that can be used to control the slew rate input of the external SDI cable driver.

Receiver Functions

The `s6gtp_sdi_control` module provides several functions required for SDI receiver operation. Some of the signals between the module and the GTP transceiver must be synchronous with the RX pipeclk, therefore this clock must be provided to the `s6gtp_sdi_control` module on the rxusrclk port.

The SDI mode of the GTP receiver is selected by the rx_mode input port. Changing this input port causes the module to dynamically configure the GTP receiver for the selected SDI mode. The `triple_sdi_rx_20b` module automatically searches for the correct SDI mode whenever it is not locked to the input SDI signal. To implement this search function, the mode output of the `triple_sdi_rx_20b` module must drive the rx_mode input of the `s6gtp_sdi_control` module. When the SDI mode changes, the `triple_sdi_rx_20b` module changes the line rate of the GTP receiver by changing the value on the RXRATE port of the GTP transceiver. it also changes GTP CDR attributes through the DRP port.

The `triple_sdi_rx_20b` module issues a RXCDRRESET to the GTP transceiver under two conditions:

- RXCDRRESET is asserted when the receiver rate detection module detects that the input SDI bit rate has changed between the two HD-SDI bit rates or between the two 3G-SDI bit rates. This is essential to allow the CDR to lock correctly to the new bit rate.
- The `triple_sdi_rx_20b` module resets the GTP receive buffer by asserting its rxbufreset output, which must be connected to the RXBUFRESET input port of the GTP transceiver whenever there is a receive buffer error. Receive buffer errors are indicated by assertion of bit 2 of the RXBUFSTATUS port.

The `triple_sdi_rx_20b` module contains a bit rate detector. The bit rate detector compares the frequency of the recovered clock to the frequency of the dclk to determine the input SDI bit rate. The rx_rate output is Low if the bit rate is 1.485 Gb/s or 2.97 Gb/s, and High if it is 1.485/1.001 Gb/s or 2.97/1.001 Gb/s. The RATE_REFCLK_FREQ parameter/generic must be used to specify the frequency of the rate_refclk for the rate detection function to operate correctly. Even if the application does not use the rx_rate output port of the `triple_sdi_rx_20b` module, the rate detection function is still required. This is because the rate detection function is an essential part of the RXCDRRESET generator.

Creating the GTP Transceiver Wrapper

GTP transceiver wrapper modules must be created to support SDI. This section describes the steps necessary to create GTP transceiver wrappers compatible with SDI interfaces.

Running the GTP Transceiver Wizard

The Spartan-6 FPGA GTP Transceiver Wizard is used to create the GTP transceiver wrapper modules that must be instantiated into the application. The wizard can be used to create a GTP transceiver wrapper module for a single GTP transceiver tile which can then be instantiated multiple times in the design.

The wizard does not support SDI as a protocol template. Therefore the wizard must be configured manually to create the GTP transceiver wrapper for SDI. The GTP transceiver wrapper created by the wizard must also be manually edited after it is created to implement an SDI interface.

These steps describe how to create and edit the GTP transceiver wrapper for an SDI application using version 1.4 and later of the wizard.

1. The easiest way to run the wizard is to use the New Source Wizard in the ISE® software project manager, invoked by the New Source command. When the New Source Wizard starts, set the type as **IP (CORE Generator & Architecture Wizard)** and provide a file name. The file name chosen is not only used as the GTP transceiver wrapper file name, but also as the name of the GTP transceiver wrapper module. The first page of the New Source Wizard GUI is shown in Figure 16.

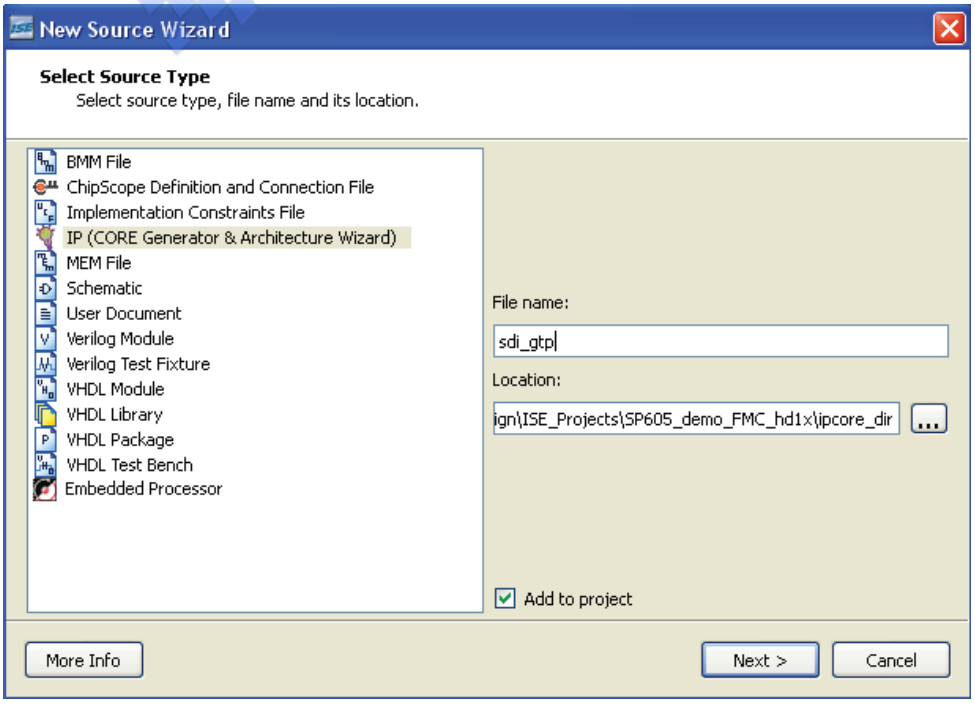


Figure 16: **New Source Wizard, Page 1**

Checking **Add to project** adds the GTP transceiver wrapper source to the list of source files in the ISE project. This step is optional. If this is option is selected, the GTP transceiver wrapper is added to the user's project source code. If this is option not selected, the two GTP transceiver wrapper source code, `filename.vhd` (or `filename.v`) and `filename_tile.vhd` (or `filename_tile.v`), files must be added to the project manually.

2. Click **Next >** to move to the second page of the New Source Wizard. Select the **Spartan-6 FPGA GTP Transceiver Wizard**, as shown in Figure 17, then click **Next >** to move to the third page of the New Source Wizard.

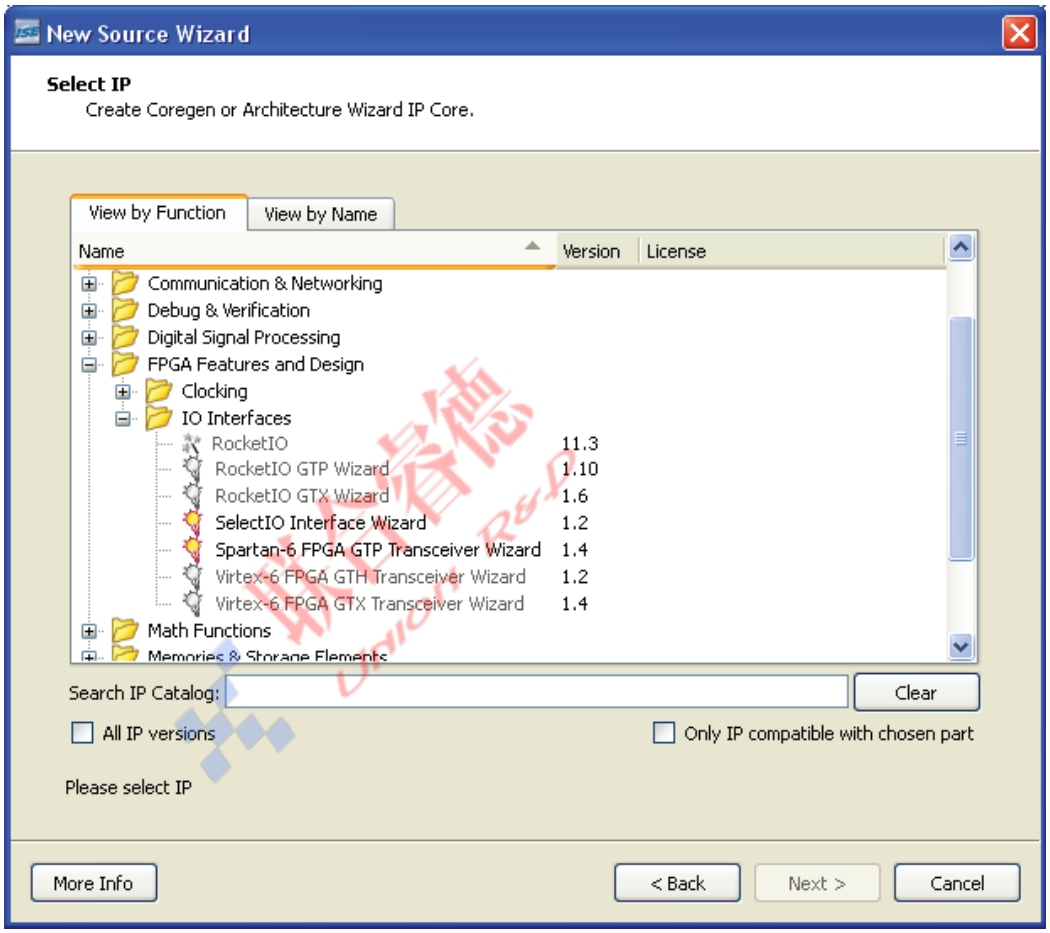


Figure 17: **New Source Wizard, Page 2**

3. On the third page of the New Source Wizard, click **Finish** to launch the GTP Transceiver Wizard. The first page of the Spartan-6 FPGA GTP Transceiver Wizard is shown in Figure 18.

Figure 18: **Spartan-6 FPGA GTP Transceiver Wizard, Page 1**

Table 12 lists all wizard settings pertinent to SDI and their recommended values.

Table 12: **Spartan-6 FPGA GTP Transceiver Wizard Settings**

Wizard Page	Parameter	Recommended Setting (SDI)	Description
1	Tile Location	Application Specific	Selects the GTP1A_DUAL tile to be used. Location origin (X0_Y0) is at upper left of die.
1	GTP0 REFCLK	Application Specific	Selects the reference clock to be used by both the transmitter and receiver. Can select clocks from adjacent tiles. *It is important that a reference clock input is selected for the TX clock source. The RX can use this same clock, which does not have to be synchronous to the incoming data.*
1	GTP1 REFCLK	Application Specific	Selects the reference clock to be used by both the transmitter and receiver. Can select clocks from adjacent tiles. *It is important that a reference clock input is selected for the TX clock source. The RX can use this same clock, which does not have to be synchronous to the incoming data.*
2	Use Dynamic Reconfiguration Port	(Checked)	This box must be checked to use triple-rate RX. This allows parameters to be changed dynamically, for example, from SD-SDI to HD-SDI.

Table 12: **Spartan-6 FPGA GTP Transceiver Wizard Settings *(Cont'd)***

Wizard Page	Parameter	Recommended Setting (SDI)	Description
2	Protocol Template	"sdi" if available or "Start from scratch"	This sets defaults for predefined standards such as PCI Express®. In wizard version 1.5 and later, there is an "sdi" menu option that can be used.
2	Target Line Rate	1.485 Gb/s	Sets the line rate desired for the target design. This is changed dynamically through the DRP controller.
2	Reference Clock	148.5 MHz	Selects the optimal reference clock frequency provided by the application.
2	TX Line Rate	1.485 Gb/s	Selects the target line rate. This can be independent of the receive line rate and is changed dynamically through the DRP controller.
2	TX Encoding	None	Disables TX 8B/10B encoding.
2	TX Data Path Width	20	Sets the datapath interface width for the transmitter.
2	RX Line Rate	2.97 Gb/s	Selects the target line rate. This can be independent of the transmit line rate.
2	RX Decoding	None	Disables RX 8B/10B decoding.
2	RX Data Path Width	20	Sets the datapath interface width for the receiver.
3	8B/10B Optional Ports	N/A	None of the ports are available because 8B/10B encoding/decoding is not enabled.
4	Enable TX Buffer	(Checked)	Enables the TX buffer between the parallel clock domain of the PCS to the serial clock domain of the PMA. This buffer can be disabled to minimize latency, but this generates a TX synchronizer module that requires a synchronization procedure at start-up.
4	TXUSRCLK Source	TXOUTCLK	The setting is based on whether or not the TX buffer is enabled. If the TX buffer is enabled, TXOUTCLK must be selected. If it is not enabled, REFCLKOUT must be selected.
4	Enable RX Buffer	(Checked)	Enables the RX buffer between the serial clock domain of the PMA and the parallel clock domain of the PCS. This buffer can be disabled to minimize latency, but this generates an RX synchronizer module that requires a synchronization procedure at start-up.
4	RXUSRCLK Source	RXRECCLK	The user clock from the receiver must be the recovered from the RX port. This clock is output on both the RXRECCLK port and on GTPOUTCLK[1].
4	Optional Clock Ports	• RXRESET • RXRECCLK • RXBUFSTATUS • RXBUFRESET • TXOUTCLK • TXRESET • TXBUFSTATUS	These ports are required to interface to the `s6gtp_sdi_control` module.
5	Comma Detection	(Not checked)	Comma detection does not apply to SDI.
5	Optional Comma Ports	N/A	Comma detection does not apply to SDI.
6	Preemphasis	0	This value is not critical when the SDI cable driver is located close to the FPGA. The value can be modified to meet the application requirements.
6	Differential Swing	0	This value is not critical when the SDI cable driver is located close to the FPGA. The value can be modified to meet the application requirements.

Table 12: **Spartan-6 FPGA GTP Transceiver Wizard Settings *(Cont'd)***

Wizard Page	Parameter	Recommended Setting (SDI)	Description
6	RX Equalization	10	This value is not critical when the SDI cable driver is located close to the FPGA. The value can be modified to meet the application requirements.
6	RX Termination	• AC coupling disabled • VTTRX for termination voltage	The internal AC coupling capacitors are not large enough to handle the long run lengths of the SDI protocols. External AC coupling capacitors are used between the cable equalizer and the GTP receiver inputs.
6	Optional Termination Ports	RXCDRRESET	This port is required to interface to the `s6gtp_sdi_control` module.
7 to 10	(All other settings)	N/A	These do not apply to SDI.

Manually Editing the GTP Transceiver Wrapper Files

The Spartan-6 FPGA GTP Transceiver wizard produces a family of files supporting the configuration. The `filename_tile.v` file contains the actual instantiation of the GTP transceiver tile. This is where the GTP transceiver parameters are specified and the ports are connected. In some cases, the reference clock selects, REFSELDYPLL0 and REFSELDYPLL1, do not reflect the reference clock specified in the GTP0 REFCLK and GTP1 REFCLK parameters of the wizard. Typically, REFSELDYPLL0 and REFSELDYPLL1 are set to `0` when they should have another value. The user must verify the connections of these two inputs and change the settings if necessary. Refer to Chapter 2 (Shared Transceiver Features) of *Spartan-6 FPGA GTP Transceivers User Guide* [Ref 4] for details on clock selection.

Reference Design

The reference design files can be downloaded at https://secure.xilinx.com/webreg/clickthrough.do?cid=154516.

Table 13 lists the reference design matrix, which provides additional information about the design.

Table 13: **Reference Design Matrix**

Parameter	Description
General	
Developer Name	Xilinx
Target Device	Spartan-6 FPGA
Source Code provided	Yes
Source Code format	VHDL, Verilog
IP used	Yes (Design uses code/IP from XAPP875 [Ref 2] and PicoBlaze™ processor)
Simulation	
Functional simulation performed	No
Timing simulation performed	No
Testbench used for functional and timing simulations provided	No
Testbench format	N/A

Table 13: **Reference Design Matrix *(Cont'd)***

Parameter	Description
Simulator Software / version	N/A
SPICE / IBIS simulations	No
Implementation	
Synthesis tool / version	XST version in ISE software v12.2
Implementation tool / version	ISE software v12.2
Static Timing Analysis performed	Yes
Hardware Verification	
Hardware verified	Yes
Hardware platform used for verification	SP605 demonstration board with CTXIL671

FPGA Resource Usage

Table 14 shows the FPGA resources required for each of the triple-rate SDI interface reference designs described in this document. The resource usage includes all the modules required to implement the interface, including the `s6gtp_sdi_control` module.

Table 14: **Triple-Rate SDI Interface FPGA Resource Usage**

Reference Design	FFs	LUTs	PLLs	BUFGs
Triple-Rate RX (no EDH processor)	485	416	1	2
Triple-Rate RX (with EDH processor)	860	917	1	2
Triple-Rate TX (with EDH processor)	2533	2883	1	2
Triple-Rate TX (EDH processor optimized out of the design)	2143	2403	1	2

The triple-rate SDI receiver does not include an EDH processor for SD-SDI, but an EDH processor can be connected to the outputs of the SDI receiver. Table 14 shows the size of the receiver both with and without an EDH processor. The SDI TX includes an EDH processor; however, it can be optimized out of the design by connecting the insert_edh input port of the `triple_sdi_tx_output_20b` module Low. Table 14 shows the size of the transmitter both with the EDH processor and with it optimized out of the design. In both the receiver and transmitter cases, the EDH processor used is described in Chapter 6 of *Audio/Video Connectivity Solutions for Virtex-II Pro and Virtex-4 FPGAs* [Ref 1].

The results shown were achieved using ISE software v12.2 with design strategy set to **area reduction strategy1**. The XST "Safe Implementation" property was set to **Yes**.

The SDI receiver and transmitter interface designs do not require any block RAMs or DSP48A1 slices.

Timing

The maximum datapath clock frequency used in these designs is 148.5 MHz. This timing can be met in -3 and above speed grades in Spartan-6 devices. In the -2 speed grade, the GTP transceiver parallel interface rate exceeds the component switching limit.

The RXRECCLK and TXOUTCLK (GTPOUTCLK[1:0]) signals generated by the GTP transceiver can, under some conditions, have erratic timing when the input SDI bitstream stops and restarts, or when the GTP transceiver is switched between SDI modes. These periods of erratic timing can cause problems for sequential control logic, such as finite state machines. Most synthesis tools, by default, optimize away illegal state recovery for finite state machines.

The erratic timing of the GTP transceiver clocks can cause the finite state machine to go into illegal states. Therefore it is highly recommended that the synthesis tool be forced to generate “safe” implementations of finite state machines that include illegal state recovery.

Example UCF constraint files are supplied with the reference designs and can be used as examples of the timing and placement constraints required for SDI interfaces.

SD-SDI DRU

The DRU used for recovering the SD-SDI data is not supplied in source-code form. It is supplied as a precompiled NGC file. For Verilog implementations, a wrapper file called `dru.v` defines the ports of the DRU. This wrapper file must be included in the project. The `dru.ngc` file must be placed in the project directory where the tools can find it and include it into the design. For VHDL implementations, the DRU component must be defined in the entity. See [Ref 2] for a description of the DRU.

Conclusion

This document describes the triple-rate SDI receiver and transmitter reference designs for the GTP transceivers in Spartan-6 devices. The Spartan-6 FPGA GTP transceivers are well suited for implementing SDI interfaces, providing a high degree of performance and reliability in a low-cost device.

References

This section provides links to documents referenced in this document:

1. XAPP514, *Audio/Video Connectivity Solutions for Virtex-II Pro and Virtex-4 FPGAs*
2. XAPP875, *Dynamically Programmable DRU for High-Speed Serial I/O*
3. XAPP1014, *Audio/Video Connectivity Solutions for Virtex-5 FPGAs*
4. UG386, *Spartan-6 FPGA GTP Transceivers User Guide*
5. UG753, *FPGA Broadcast Mezzanine Card User Guide*

Appendix A: Triple-Rate SDI Demonstration

The demonstration design described in this appendix incorporates both the transmitter and receiver designs for triple-rate SDI on Spartan-6 devices, and demonstrates the basic functionalities of both. The demonstration runs on a Xilinx® SP605 evaluation board connected to an FPGA broadcast mezzanine card (FMC card). Refer to [Ref 5] for a description of the FMC card.

Setup

Figure 19 shows the basic elements of the hardware. The FMC card is connected to the SP605 board via the low pin count (LPC) FMC connector, with the FMC card on top. The FMC card receives all of its power and control from the SP605 board.

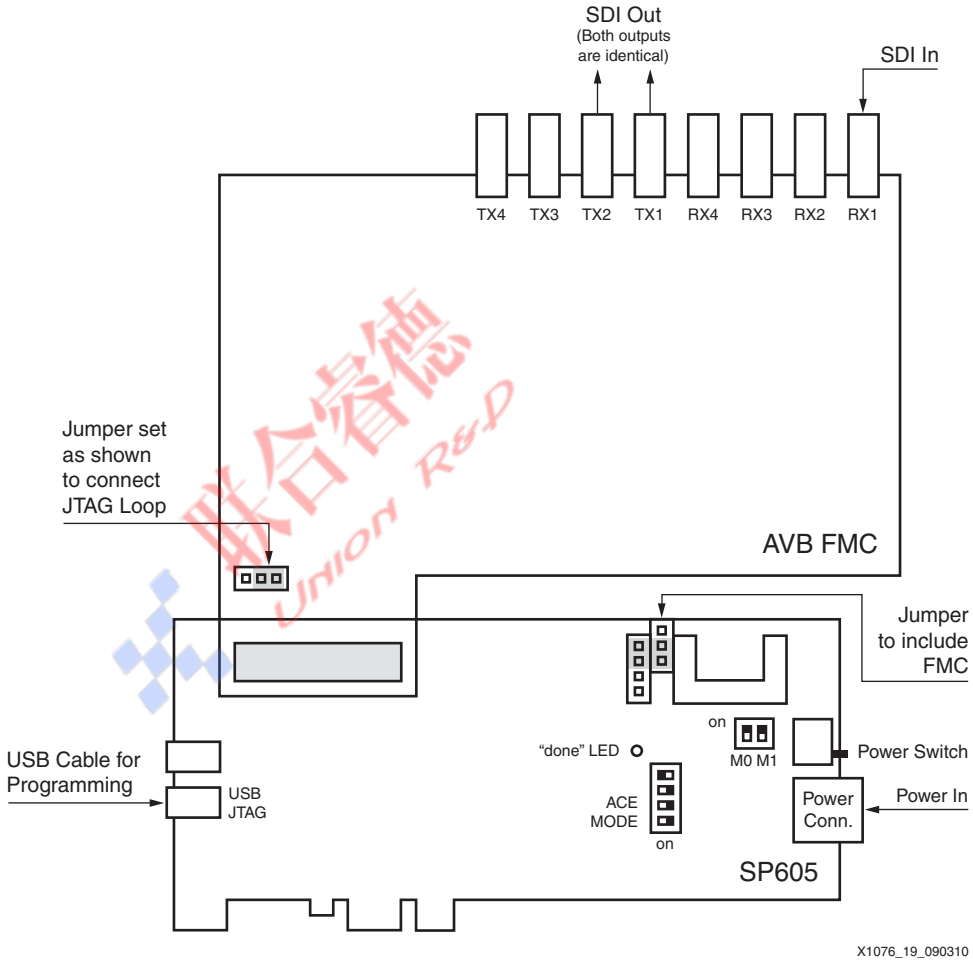


Figure 19: **Triple-Rate SDI Demonstration Setup**

- Two jumpers must be set up properly to include the FMC card in the JTAG shift loop:
  - FMC JTAG jumper (J19) on the SP605 board must be set to include the FMC connector in the loop.
  - JTAG CONFIG jumper (J21) on the FMC card must be set to exclude the expansion FMC connector from the loop.
- Two sets of DIP switches affect loading the Spartan-6 FPGA onto the SP605 board:
  - Mode switches (SW1) for M0 and M1 should both be set to ON.
  - The ACE MODE switch (S1) and program switches pertain to programming the FPGA from a CompactFlash card. Normally, switches 1, 2, and 3 are OFF and switch 4 is ON.

- The power cable must be plugged into the power connector, and a USB cable is plugged into the USB JTAG port to program the FPGA via JTAG.
- Video inputs and outputs are at the top of the FMC card. There are two identical transmitter outputs and one receiver input.
- The Spartan-6 FPGA on the FMC card must be preprogrammed for audio and video demonstrations. Refer to *FPGA Broadcast Mezzanine Card User Guide* [Ref 5] for more information on the FMC card.
- The Spartan-6 FPGA on the SP605 board can be programmed with the demonstration bit file via the USB JTAG port using the ChipScope™ tool. The done bit LED indicates programming status. The demonstration bit file enables both the receiver and transmitter demonstrations.

Triple-Rate Transmitter Demonstration

The transmitter demonstrates the capability to transmit in SD-SDI, HD-SDI, and 3G-SDI modes. There are controls to set the mode select to various line standards and bit rates. Multiple test patterns can be used in the demonstration.

SD-SDI, HD-SDI, and 3G-SDI modes can be transmitted from the SDI TX1 and SDI TX2 connectors on the FMC card. Both outputs are identical. The demonstration is controlled via DIP switches and push buttons on the SP605 board. The basic elements of the hardware for the triple-rate SDI transmitter demonstration is shown in Figure 20.

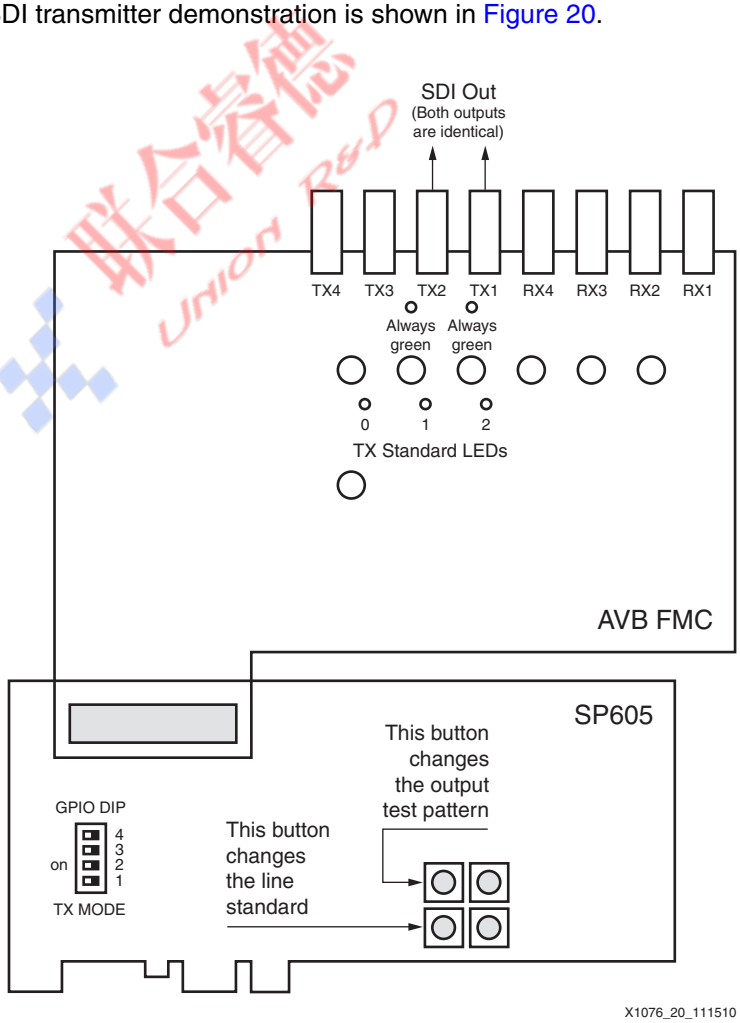


Figure 20: **Triple-Rate SDI Transmitter Demonstration**

The TX mode is controlled by the GPIO DIP switches on the SP605 board. Switches 1 and 2 determine the mode, as identified in Table 15. Switch 3 controls the reference clock frequency, and therefore the bit rate. Not all line standards have a valid /M mode, so switch 3 is ignored for line standards that have only one valid bit rate. Switch 4 should always be OFF.

Table 15: **Transmitter Modes**

Switch 1	Switch 2	Mode
OFF	OFF	HD-SDI
OFF	ON	SD-SDI
ON	OFF	3G-SDI
ON	ON	Invalid

The line standard is indicated by the three TX standard LEDs that illuminate orange. The bit 0 LED is near the AESTX1 connector, the bit 1 LED is near SYNC IN, and the bit 2 LED is near the AES3 RX2 connector. The line standard can be incremented using the line standard button (SW5). The standards are identified in Table 16.

Table 16: **Transmitter Line Standards**

Mode	Standard Value	LED 2	LED 1	LED 0	Switch 3 (Bit Rate)	Line Standard
SD-SDI	0	OFF	OFF	OFF	x	NTSC
	1	OFF	OFF	ON	x	PAL
HD-SDI	0	OFF	OFF	OFF	x	720p 50 Hz
	1	OFF	OFF	ON	OFF	1080sF 24 Hz
		OFF	OFF	ON	ON	1080sF 23.98 Hz
	2	OFF	ON	OFF	OFF	1080i 30 Hz
		OFF	ON	OFF	ON	1080i 29.97 Hz
	3	OFF	ON	ON	x	1080i 25 Hz
	4	ON	OFF	OFF	OFF	1080p 30 Hz
		ON	OFF	OFF	ON	1080p 29.97 Hz
	5	ON	OFF	ON	x	1080p 25 Hz
	6	ON	ON	OFF	OFF	1080p 24 Hz
		ON	ON	OFF	ON	1080p 23.98 Hz
	7	ON	ON	ON	OFF	720p 60 Hz
		ON	ON	ON	ON	720p 59.94 Hz
3G-SDI	4	ON	OFF	OFF	OFF	1080p 60 Hz
		ON	OFF	OFF	ON	1080p 59.94 Hz
	5	ON	OFF	ON	x	1080p 50 Hz

There are two or three test patterns available depending on mode. In each case, one of these is the pathological pattern. The test pattern can be changed via the test pattern button (SW4).

Triple-Rate Receiver Demonstration

The receiver demonstrates the capability to receive in SD-SDI, HD-SDI, and 3G-SDI modes, determine the mode, and decode and lock onto the correct mode and line standard. For 3G-SDI mode, only Level A format is supported in this demonstration. For SD-SDI mode, EDH

packets are analyzed and errors are indicated. For HD-SDI and 3G-SDI modes, CRC is checked, and errors are indicated.

The basic elements of the hardware for the triple-rate SDI transmitter demonstration are shown in Figure 21. SDI is input at the SDI RX1 port on the FMC card. The adjacent LED illuminates green if the carrier detect from associated cable equalizer device on the card is active. If no carrier is detected, the LED is red.

The GPIO LEDs at the bottom of the SP605 board serve as the TX status LEDs (Figure 21). Their functions are described in Table 17.

Table 17: **Transmitter Status LEDs**

LED	Name	Description
0 (DS3)	Rate	Illuminates when the received line standard is /1.001 rate.
1 and 2 (DS4 and DS5)	Mode	SDI Mode. Refer to Table 18 for mode selection.
3 (DS6)	rx error	Receiver error bit. For SD-SDI mode, this is an EDH error. For HD-SDI and 3G-SDI, it is a CRC error. This LED remains illuminated until cleared with the error clear pushbutton SW7 (see Figure 21 for location of SW7).
4 (DS7)	Locked	Illuminates when the SDI mode and rate are successfully detected and locked.

Table 18: **Receiver Modes**

LED 2	LED 1	Mode
OFF	OFF	HD-SDI
OFF	ON	SD-SDI
ON	OFF	3G-SDI
ON	ON	Invalid

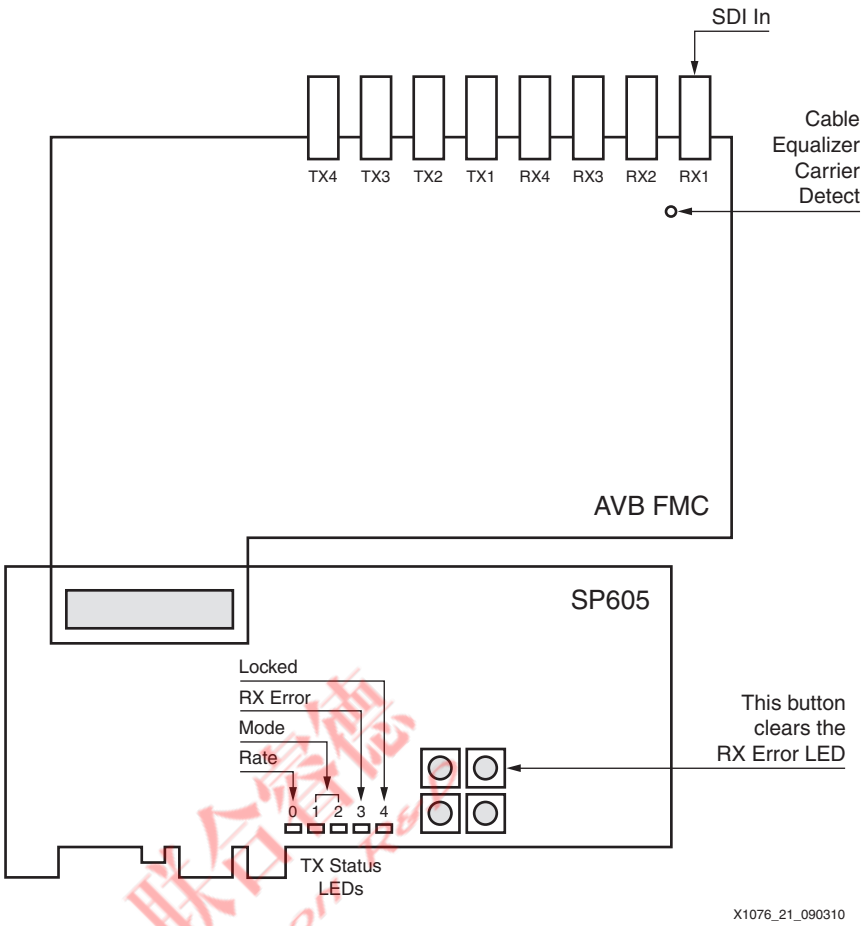


Figure 21: **Triple-Rate SDI Receiver Demonstration**

Much of the data and status information from the receiver is only observable using the ChipScope tool. A project for the ChipScope Pro software v12.2 is included with the reference design. It contains a VIO module and an ILA module. The VIO module provides status information about the receiver and the video format being received. The ILA module shows the video and timing information output by the receiver.

Appendix B: Triple-Rate SDI Genlock Pass-Through Demonstration

The SP605 board in combination with the FPGA broadcast mezzanine card (FMC card) and a clock module can be used to demonstrate the triple-rate RX and TX capabilities of Spartan-6 devices in a pass-through configuration and in a genlock mode.

Only a genlocked demonstration is possible for pass-through on the SP605 board with the FMC card. A non-genlocked pass-through demonstration requires a free-running clock for the SDI receiver, and also a recovered clock for the SDI transmitter (see Figure 26, page 59). However, it is not possible to use two clocks on the SP605 board through the FMC because the FMC connector of the SP605 board is wired to a single GTP transceiver on the Spartan-6 FPGA. Each transceiver has a single clock input (See Figure 22).

Therefore, although non-genlocked pass-through SDI is supported by the Spartan-6 FPGA, the fact that it requires separate transceivers for the RX and TX precludes it from being demonstrated on the SP605/FMC card demonstration platform. Thus, a genlocked demonstration is included with the reference design.

The demonstration design described in this appendix incorporates both the transmitter and receiver designs for triple-rate SDI on Spartan-6 devices connected in a pass-through configuration and demonstrates the basic functionality of both. The demonstration runs on an SP605 card connected to an FMC card with a clock module card (Clock Module L) mounted on the FMC card.

The FMC connector on the SP605 board contains connections to a single GTP transceiver on the Spartan-6 FPGA. To use this single transceiver with a single shared clock to both transmitter and receiver, the demonstration utilizes a genlock configuration in which a reference sync is used to generate the reference clock to the GTP transceiver.

Hardware Configuration

To use the single transceiver connected to the FMC connector, this demonstration uses a single reference clock that is synchronized to the incoming video via genlock. Thus, this clock is present even when the SDI source is disconnected. This prevents lock-up that would otherwise be possible if one attempted to use the incoming SDI as the timing source to recover a clock from itself.

Figure 22 is a simplified diagram of the pass-through demonstration. The video source and the SP605 board are both genlocked to a single sync source. The SDI stream is received by the GTP transceiver in the FPGA, converted to parallel data and decoded by the RX logic. The parallel data is passed to the TX logic where it is re-encoded for transmission and sent out through the TX portion of the GTP transceiver. The video out of the SP605 board is identical to the input stream but delayed.

This demonstration uses the genlock control HDL module to control genlock operation using the FMC card and the clock module mounted on it. This module is described in Genlock Control Module, page 53.

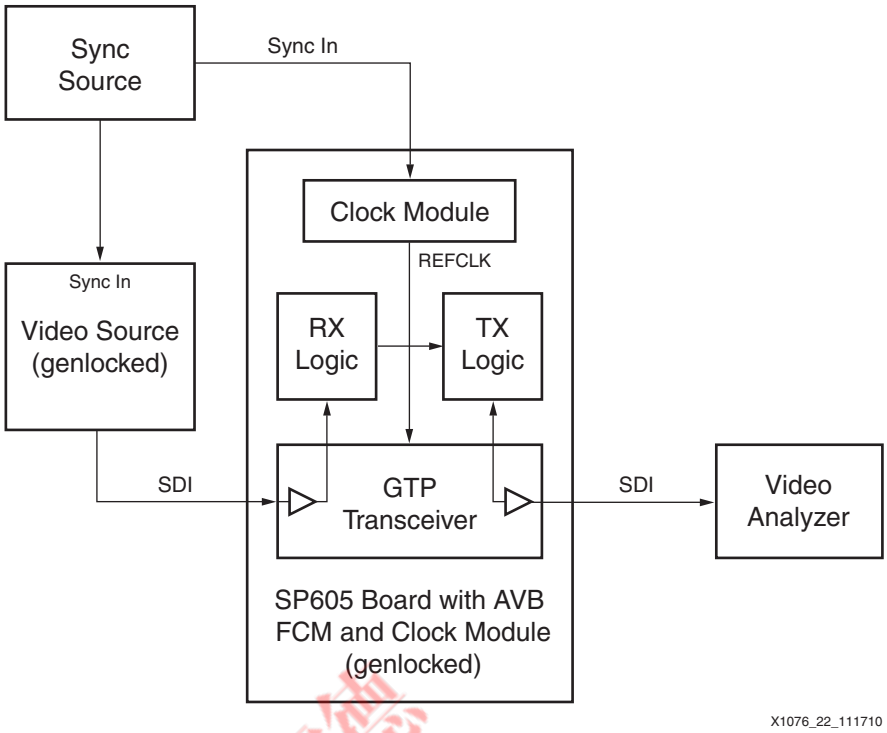


Figure 22: **SP605 Configuration for Genlocked Pass-Through Demonstration**

Figure 23 shows how the `s6_sdi_rx_light_20b` module in the RX section is connected to the `triple_sdi_vpid_insert` module in the TX section. Mode and line standard information is extracted by the RX section to produce the correct clocks and to operate the TX section in the proper manner.

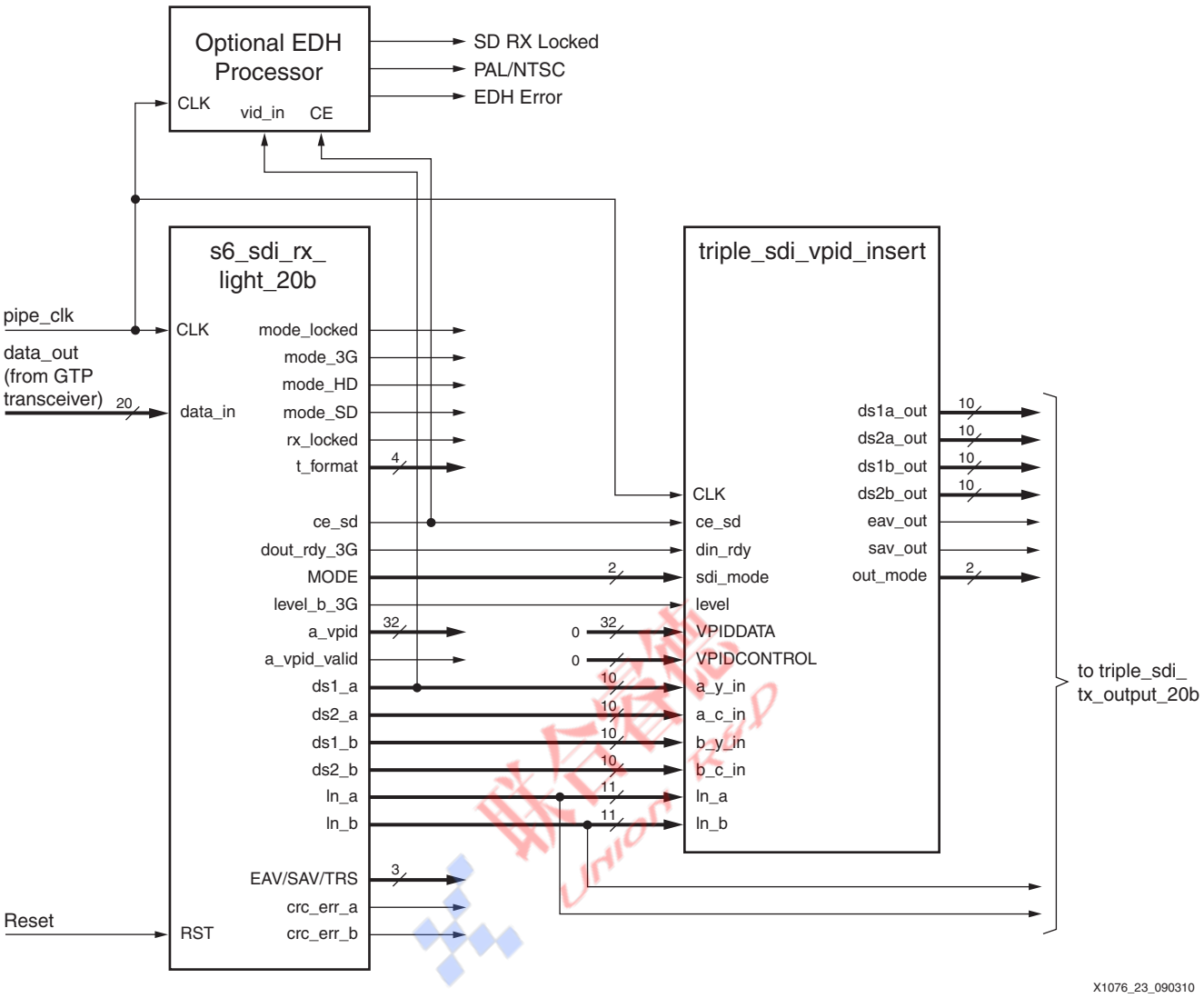


Figure 23: **RX to TX Connections**

In HD-SDI mode and 3G-SDI mode, the pipeline logic in the RX and TX sections operates at the same pipeline clock rate with an operation occurring at every clock. In contrast, in SD-SDI mode, the RX pipeline clock runs at half the rate of the TX clock, and a clock enable is used to slow the timing of the pipeline relative to the clock, so that operations happen at the 270 MHz SD-SDI rate. The lower RX clock rate is necessary to accommodate the data recovery unit, and the higher TX clock rate is required because the data rate of the GTP output must be an integer multiple (11X) of the SD-SDI data rate (see SD-SDI Transmitter Operation, page 23 for details). Also, the recovered clock enable from the RX section follows a 3/3/3/2 on a 74.25 MHz clock, whereas the TX modules require a clock enable cadence of 5/6/5/6 on a 148.5 MHz clock.

Figure 24 shows how data is passed from the RX domain to the TX domain, and shows the relationship of the various timing signals. A small state machine is used to create the TX clock enable with 5/6/5/6 cadence, and a shortened pulse width, but locked to the RX clock enable. Figure 24 shows the states of this state machine and the sequence in the RX clock enable that triggers a reset of this state machine.

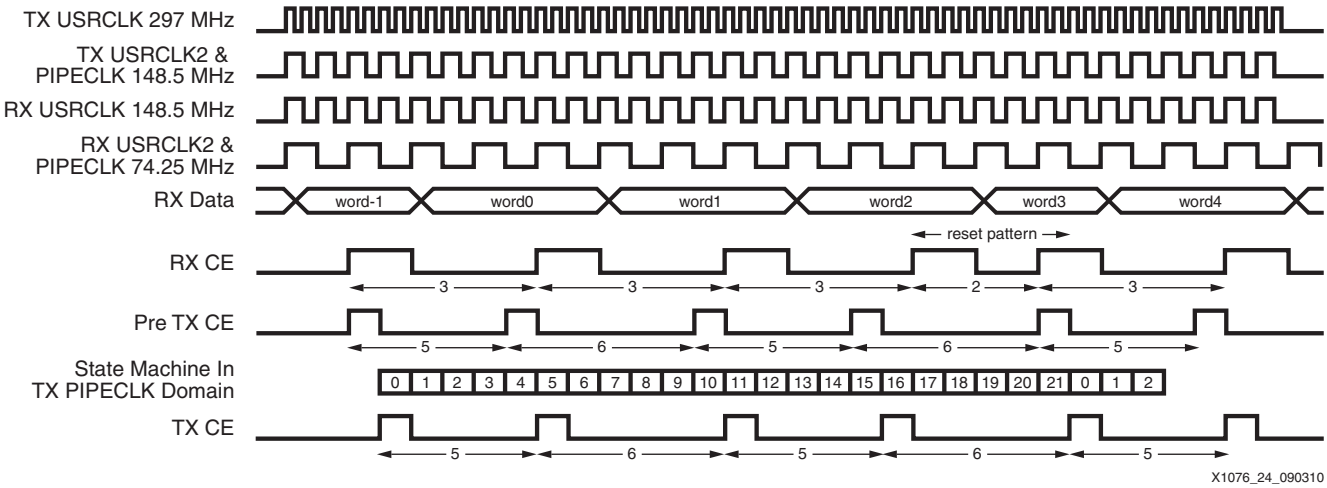


Figure 24: **RX to TX Timing of Data and Clock Enables**

Genlock Control Module

The genlock control module, `genlock_control`, controls multiple Si5324 digital phase-locked loops (PLLs) to implement a genlocked system. The module handles switching the system in and out of genlock mode. When not in genlock mode, the system runs from the local 27 MHz crystal oscillator on the broadcast FMC mezzanine card.

The module is designed to work with either two or three Si5324 devices. The main Si5324 device, generally the Si5324 that is on the FMC mezzanine card itself, as opposed to the ones on the clock modules, is used to create a 27 MHz clock from the external horizontal sync signal when running in genlocked mode. In non-genlocked mode, this Si5324 passes the 27 MHz signal from the local crystal oscillator straight through, running in a 27 MHz in / 27 MHz out mode. The 27 MHz signal created by the main Si5324 is used as the reference clock to the other Si5324 devices that synthesize the GTP transceiver reference clocks from the 27 MHz input.

Si5324 Control

The main Si5324 device must have both its input clock source and its input frequency select selection changed as the system is switched between genlock and non-genlock modes. The secondary Si5324 devices are, essentially, statically programmed to always take in 27 MHz and output the proper GTP transceiver reference clock frequencies.

The secondary Si5324 devices, those fed from the main Si5324 device, can have very long lock times (on the order of about one minute), if they are not controlled properly. When the 27 MHz signal from the main Si5324 device changes as the system is switched between genlock and non-genlock modes, the secondary Si5324 devices must first be placed into digital holdover mode (DHOLD) before the main Si5324 device is reprogrammed, otherwise they will often exhibit this long lock time behavior.

Also, the secondary Si5324 devices must be controlled carefully immediately after FPGA configuration. They must be controlled in such a way that they perform an internal calibration (ICAL) cycle after the main Si5324 has locked and is outputting a valid and stable 27 MHz signal. On the broadcast FMC card, the only way to force a Si5324 ICAL cycle is to change the frequency synthesis scheme. The application must be written so that the Si5324 devices are held in a dummy mode by programming their frequency selection to some mode other than the actual intended operating mode until the main Si5324 device has locked. After the main Si5324 device locks, the application must switch the secondary Si5324 devices into their intended frequency synthesis mode, converting 27 MHz to the GTP transceiver reference clock frequencies. The `genlock_control` module provides a signal that indicates when the main

Si5324 device has reached lock for the very first time after FPGA configuration specifically for this purpose.

The initial startup after FPGA configuration is carefully controlled by the `genlock_control` module. The module initially sets the main Si5324 device to lock to the 27 MHz local crystal oscillator, even if genlock mode is selected. After the main Si5324 device is locked to the 27 MHz reference, the secondary Si5324 devices are programmed. These devices do an internal calibration, locking to the 27 MHz clock from the main Si5324 devices. After all Si5324 devices have locked, the module allows the system to switch to genlock mode if genlock mode has been selected. This ensures that all devices are initially locked and provides a clean transition to genlock mode.

There is a case that is not completely handled by the `genlock_control` module. If the system is operating in genlock mode and the frequency of the external horizontal sync changes, the control module cannot place the secondary Si5324 devices into digital hold mode before the main Si5324 device loses lock. Therefore, this case often experiences long lock times for the secondary Si5324 devices. The secondary Si5324 devices do recover and the system returns to normal operation, but it can take up to a minute for this to occur.

Inputs and Outputs

The ports of the `genlock_control` module are described in Table 19.

Table 19: **Ports of the `genlock_control` Module**

Port Name	I/O	Width	Description
clk	In	1	This clock input must be driven by the same clock that controls the clk inputs of the `main_avb_control` and `cm_avb_control` modules. This is typically the 27 MHz clock signal from the crystal oscillator on the broadcast FMC card (which is not the same signal as the 27 MHz output of the main Si5324).
hsync_in	In	1	This must be driven by the horizontal signal from the FMC mezzanine card that comes from the LMH1981 sync separator. The horizontal sync signal enters the carrier board on the LA30_P signal (H34).
genlock_enable	In	1	This input port determines whether the system runs in genlock mode (1) or non-genlock mode (0). It is treated as an asynchronous input and synchronized to clk before being used internally.
default_code	In	5	This port supplies the non-genlock input frequency select value for the main Si5324 device. When running in non-genlock mode, the in_freq_sel output port of this module is set equal to the default_code value. Typically, this is set to `5'b10111` to select a 27 MHz input frequency.
default_bw_sel	In	4	This port supplies the value that is written into the bandwidth select registers of the main Si5324 device when the system is running in non-genlock mode. In genlock mode, the bandwidth is specified by the GENLOCK_SI5324_BW_SEL parameter.
sync_video_fmt	In	11	This input port must be connected to the sync_video_fmt output port of the `main_avb_control` module.
sync_frame_rate	In	3	This input port must be connected to the sync_frame_rate output port of the `main_avb_control` module.
Si5324_LOL_0	In	1	This input port should be connected to the loss of lock (LOL) output of the main Si5324 device. This is usually the Si5324_LOL output port of the `main_avb_control` module.

Table 19: **Ports of the `genlock_control` Module *(Cont'd)***

Port Name	I/O	Width	Description
Si5324_LOL_1	In	1	This input port should be connected to the LOL output of one of the secondary Si5324 devices. If there is only one secondary Si5324 device, its LOL signal should be connected here. This input is normally connected to one of the Si53234_x_LOL outputs of the `cm_avb_control` module.
Si5324_LOL_2	In	1	This input port should be connected to the LOL output of one of the secondary Si5324 devices. If there is only one secondary Si5324 device used by the genlock application, this input should be wired Low. This input is normally connected to one of the Si53234_x_LOL outputs of the `cm_avb_control` module.
Si5324_clk_sel	Out	1	This port must control the input clock selection of the main Si5324 device. When this input is Low, the 27 MHz local crystal oscillator must be selected as the main Si5324 input clock. When this input is High, the horizontal sync signal from the sync separator must be selected. For most applications, both bits of the Si5324_clkin_sel port of the `main_avb_control` module should be driven Low when this output is Low and both driven High when this output is High.
in_freq_sel	Out	5	This port must control the input frequency selection of the main Si5324 device. Normally, this port is connected to the Si5324_in_fsel input port of the `main_avb_control` module.
Si5324_bw_sel	Out	4	This port must control the bandwidth selection of the main Si5324 device. Normally, this port is connected to the Si5324_bw_sel input port of the `main_avb_control` module.
Si5324_locked	Out	1	This output is Low after FPGA configuration and remains Low until the main Si5324 has locked for the very first time. This output is used to control the input frequency selects of the secondary Si5324 devices so that they switch to their final programming mode only after the main Si5324 device is locked for the first time. Thus, when this output is Low, the application must apply some valid input frequency select value, other than the actual intended input frequency select value, to the input frequency select input ports of the secondary Si5324 devices. When this output switches to High, the final input frequency select code (usually `5'b10111` for 27 MHz) must be applied to the input frequency select ports (usually the Si5324_x_in_fsel ports of the `cm_avb_control` module).
sync_missing	Out	1	This output is High if the horizontal sync input signal stops.
sync_invalid	Out	1	This output is High if the sync_video_fmt and sync_frame_rate inputs do not match a supported video standard.
local_lock	Out	1	This output is High when the system is locked to the local 27 MHz crystal oscillator. This output is only asserted after all Si5324 devices have achieved lock.

Table 19: **Ports of the `genlock_control` Module *(Cont'd)***

Port Name	I/O	Width	Description
genlock	Out	1	This output is High when the system is locked to the external sync signal. This output is only asserted after all Si5324 devices have achieved lock.
Si5324_DHOLD	Out	1	This output controls the digital holdover mode of the secondary Si5324 devices. This is usually wired to the Si5324_x_DHOLD inputs of the `cm_avb_control` for all Si5324 devices that are driven by the 27 MHz clock from the main Si5324 device.

Usage Description

The Spartan-6 FPGA genlock pass-through demonstration uses three Si5324 devices – the main Si5324 on the broadcast FMC mezzanine card and two Si5324 devices (the B and C devices) on the "L" clock module. The Si5324 devices are configured as shown in Figure 25. The main Si5324 device gets either the 27 MHz local crystal oscillator or the horizontal sync signal. Both of these inputs are multiplexed into the CKIN2 input of the Si5324device by the Spartan-3A FPGA under the control of the Si5324_clkin_sel port of the `main_avb_control` module.

The 27 MHz output of the main Si5324 device is connected to the CKIN1 port of Si5324 C by routing it through clock crossbar 1. This 27 MHz signal is also routed to CKIN2 port of Si5324 B by wrapping the 27 MHz signal through the ML605 carrier board and back out to the clock module on the broadcast FMC mezzanine card.

Si5324 B on the clock module is programmed to generate a 148.5 MHz reference clock. Si5324 C on the clock module is programmed to generate a 148.35 MHz reference clock. These two clocks go into the crossbar switch, XBAR1. Based on the line standard of the incoming video, one of these two clocks is selected to be the GTP reference clock, GBTCLK0_M2C, on the SP605 board. Because the reference clock to the GTP transceiver changes due to the received SDI data, the entire GTP transceiver must be reset each time the reference clock selection changes.

The output frequency select ports of the `cm_avb_control` module for these two Si5324 devices (on clock module L) are hard wired to generate 148.5 MHz and 148.35 MHz output frequencies from a 27 MHz input. However, the input frequency select for these two Si5324 devices is multiplexed. This multiplexer is controlled by the Si5324_locked port of the `genlock_control` module. When Si5324_locked is Low as the device comes out of configuration, the multiplexor drives a value of `5'b00000` into the input frequency select ports for the two Si5324 devices. After the main Si5324 device is locked and Si5324_locked goes High, a value of `5'b10111` (selecting 27 MHz as the input frequency) is driven to the input frequency select ports of the two Si5324 devices. This switch forces these two Si5324 devices to run an ICAL cycle and calibrate themselves to the 27 MHz input clock from the main Si5324 device. This is essential to ensure that these two Si5324 devices lock.

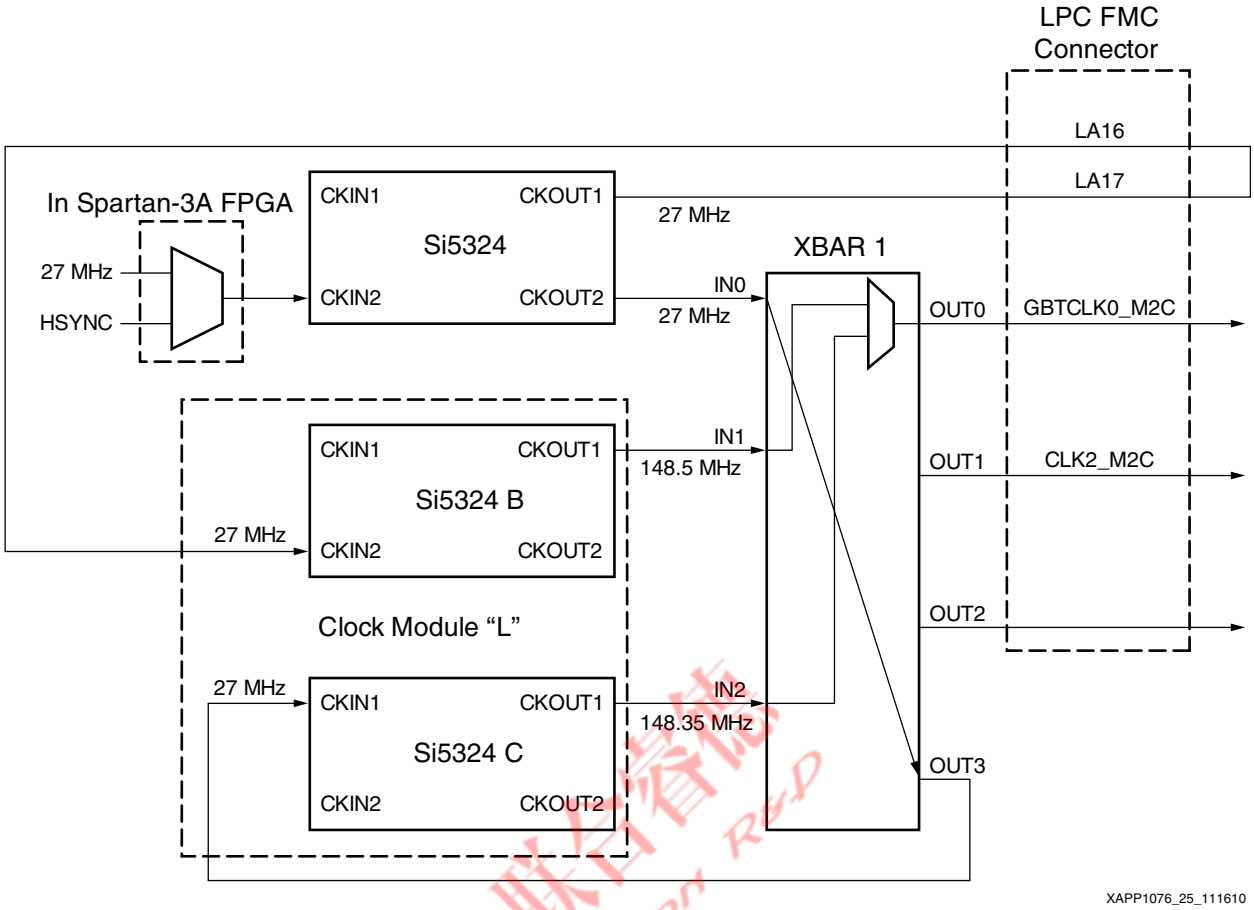


Figure 25: **Clocking for Spartan-6 FPGA Genlock Pass-Through Demonstration**

The LOL signals from the secondary Si5324 devices are also connected to the RX and TX reset inputs of the GTP transceiver. GTPRXRESET and GTPTXRESET reset the entire GTP RX and TX, respectively, including the PMA PLLs. The PMA PLLs must be reset whenever the input reference clocks change or are unstable. Thus, using the LOL signals from the Si5324 devices ensures that the PMA PLLs stay in reset until after the reference clocks have stabilized. The GTPRXRESET inputs are only connected to the LOL of Si5324 B because the receivers only use the 148.5 MHz reference clock generated by this Si5324 device. The transmitters, on the other hand, can be driven by either Si5324 device. Thus, the GTPTXRESET inputs are connected to the OR of LOL signals from Si5324 B and Si5324 C.

The three Si5324 devices on the clock module have asserted Low reset inputs that are controlled by pins on the FMC connector. Because these resets are asserted Low, the application must drive these reset inputs High using the FPGA on the carrier board. The three Si5324 resets for clock module "L" are shown in Table 20.

Table 20: **Clock Module L Si5324 Reset Signals**

Si5324 Device	Reset FMC Pin
Si53234 A	D17 (LA13_P)
Si5324 B	D26 (LA26_P)
Si5324 C	G36 (LA33_P)

It is also possible to build a genlock demonstration with two Si5324 devices, rather than three. The main Si5324 device still takes in either 27 MHz for local lock or HSYNC for genlock and generates 27 MHz. This 27 MHz output is fed into one other Si5324 device that is programmed

to generate either 148.5 MHz or 148.35 MHz, as required. To dynamically switch the TX bit rate, this secondary Si5324 device must be reprogrammed. This makes the process somewhat slower than when using three Si5324 devices.

State Diagram

Figure 26 shows the state diagram of the FSM that controls the `genlock_control` module's operation.

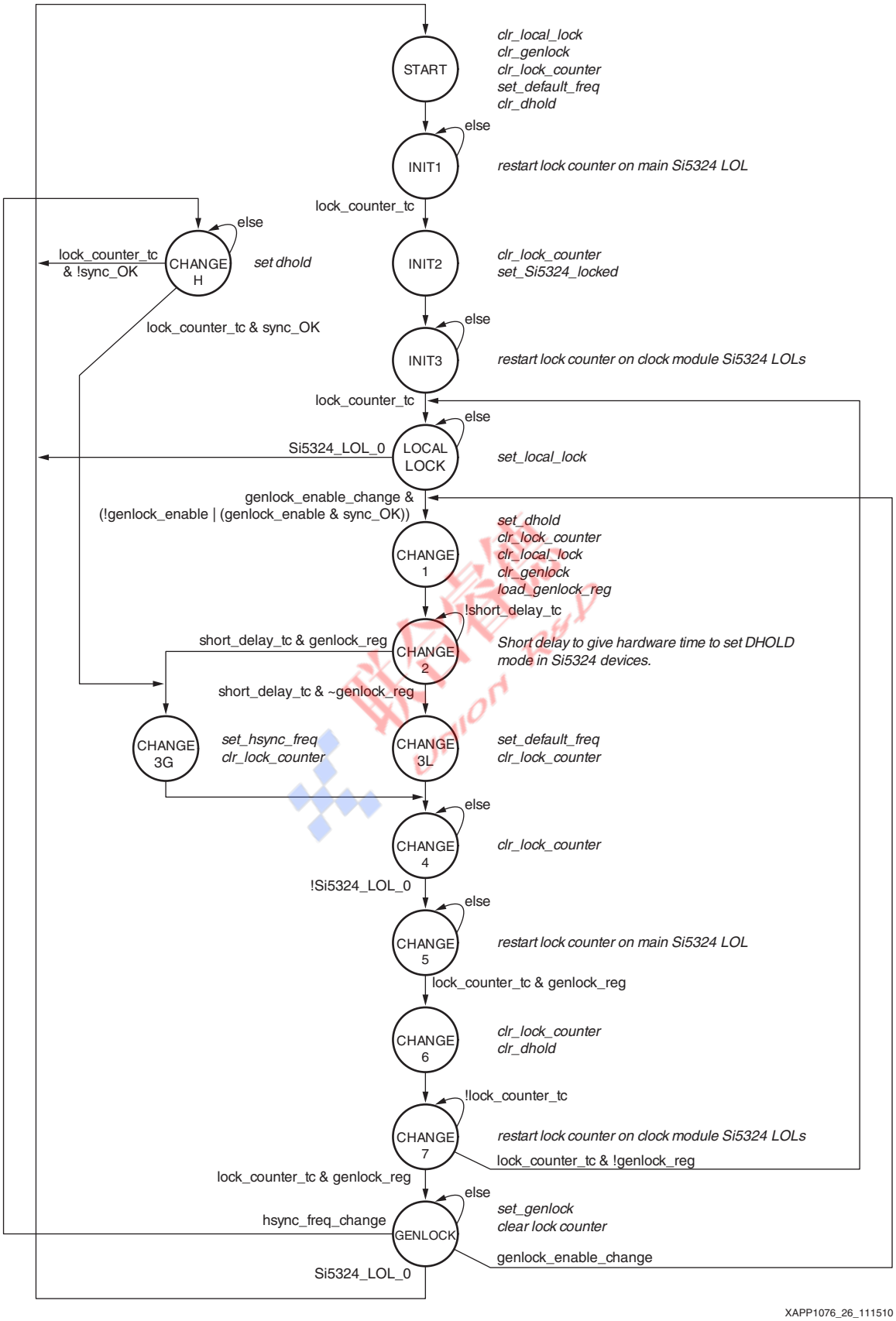


Figure 26: **Genlock Control State Diagram**

Recommended Configurations

The single transceiver used in this demonstration (due to the limitation of a single GTP transceiver on the SP605 FMC connector) is not recommended for a pass-through configuration. A preferred configuration is to use separate transceivers for RX and TX so that the RX can have an independent free-running clock, as shown in Figure 27. The TX clock should be derived from the RX recovered clock to ensure that the TX output is synchronized with the incoming video stream.

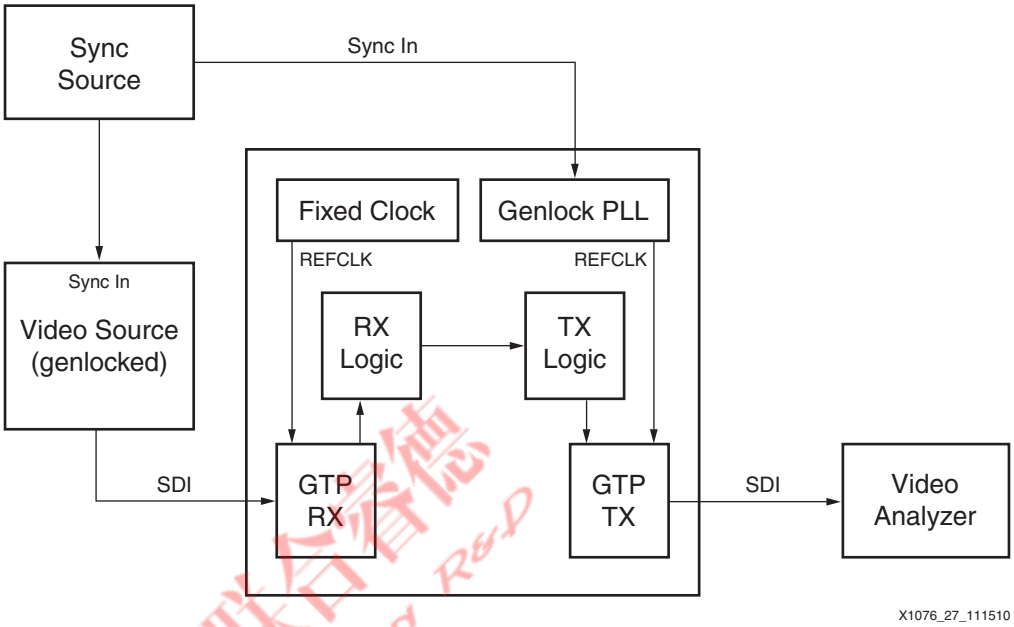


Figure 27: **Recommended Configuration for Genlocked Pass-Through Operation**

In the general case, pass-through operation does not require genlock. The GTP RX can use a fixed rate reference clock to recover a clock based on the incoming data. This clock can be passed through a clock cleaner chip external to the FPGA to reduce the jitter and provide a reference clock suitable for the GTP TX. In this way, the TX is locked to the incoming SDI data, but the incoming data need not be genlocked to any external timing source. This is shown in Figure 28.

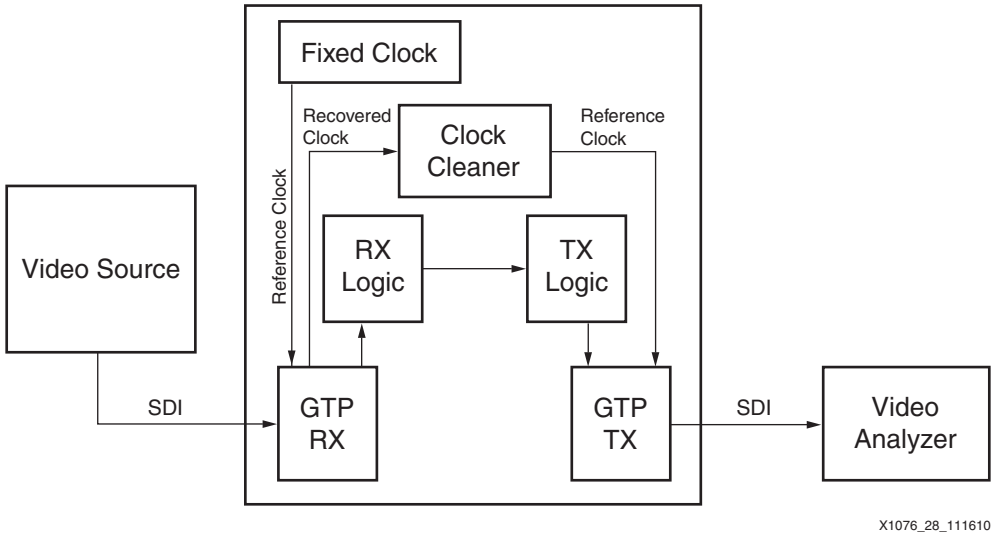


Figure 28: **Recommended Configuration for Non-Genlocked, Pass-Through Operation**

Demonstration Setup

Figure 29 shows the basic elements of the SP605 hardware. The FMC card is connected to the SP605 board via the LPC FMC connector with the FMC card on top, as shown. The FMC card receives all of its power and control from the SP605 board. For this demonstration, a clock module must be installed in the “Clock Module L” location.

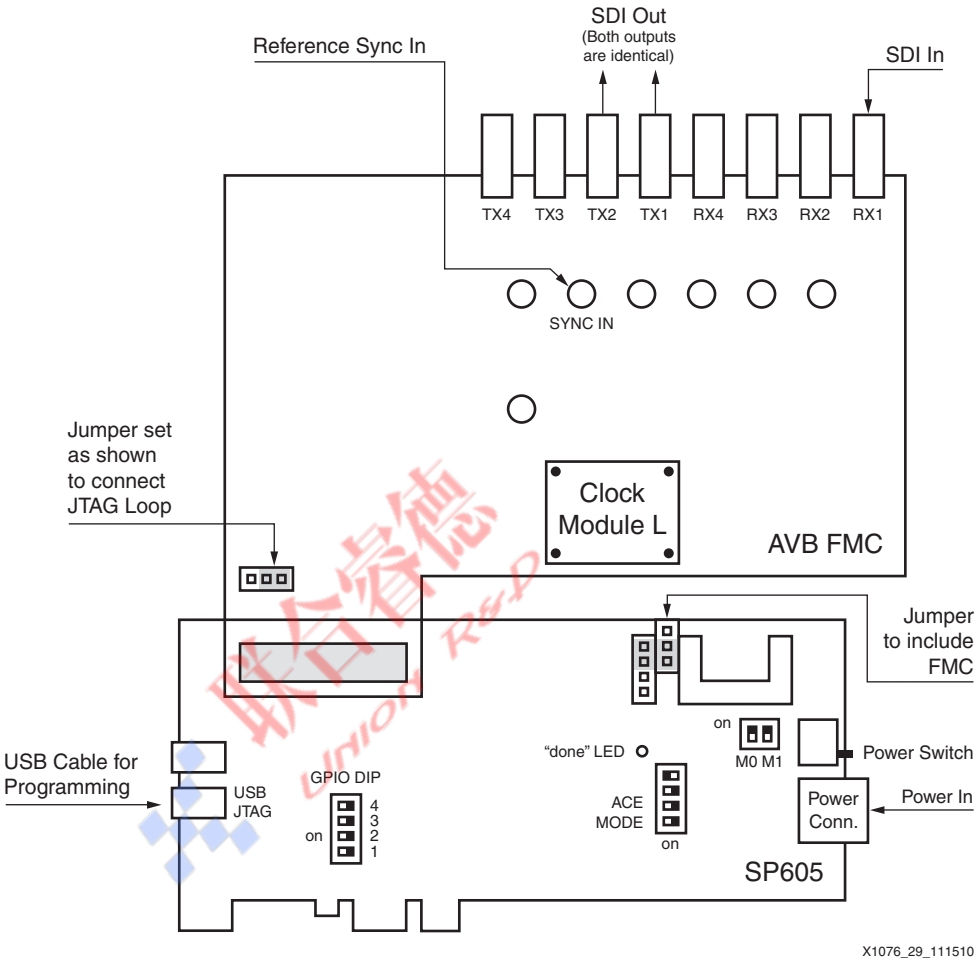


Figure 29: **Triple-Rate SDI Pass-Through Demonstration Setup**

Two jumpers must be set up properly to include the FMC card in the JTAG shift loop. One is on the FMC JTAG jumper (J19) on the SP605 card, and is set to include the FMC connector in the loop. The other is the JTAG CONFIG jumper (J21) on the FMC card and is set to exclude the expansion FMC connector from the loop.

There are two sets of DIP switches pertain to loading the Spartan-6 FPGA on the SP605 board:

1. Mode switches (SW1) for M0 and M1. These should both be set to ON.
2. The ACE MODE (S1) and program switches. These pertain to programming the FPGA from a CompactFlash card. Normally, switches 1, 2, and 3 are OFF, and switch 4 is ON.

A third set of DIP switches, the GPIO DIP switches, are not used in this demonstration.

The power cable must be plugged into the power connector and a USB cable is plugged into the USB JTAG port to program the FPGA via JTAG.

The reference clock is received through the top-mounted BNC labeled “SYNC IN”.

Video inputs and outputs are at the top of the FMC card. There are two identical transmitter outputs and one receiver input.

The Spartan-3 FPGA on the FMC card must be preprogrammed for audio and video demonstrations. Refer to *FPGA Broadcast Mezzanine Card User Guide* [Ref 5] for more information on the FMC card.

The Spartan-6 FPGA on the SP605 board can be programmed with the demo bit file via the USB JTAG port using the ChipScope software. The “done” bit LED indicates programming status. The demo bit file enables both the receiver and transmitter demonstrations.

Demonstration Operation

The pass-through demonstration shows the ability to receive SD-SDI, HD-SDI, and 3G-SDI, determine the mode, decode a lock on to the correct mode and line standard, and retransmit the data as it was received. For 3G, Level A and Level B are supported in this demonstration. As data is received, errors are checked. For SD-SDI, EDH packets are analyzed and errors indicated. For HD-SDI and 3G-SDI, CRC is checked, and errors are indicated.

Figure 30 shows elements of the configuration pertinent to the triple-rate pass-through demonstration. The reference sync used for clock creation is input to the SYNC_IN port. SDI is input at the SDI RX1 port on the FMC card. The adjacent LED lights up green if the carrier detect from associated cable equalizer chip on the card is active. If no carrier is detected, the LED is red.

The reference sync is input to the SYNC IN connector, which has an adjacent LED that is normally green, but turns red if a problem with the sync is detected by the Si5324 device to which it is connected. The reference sync must be synchronous with the SDI input video source.

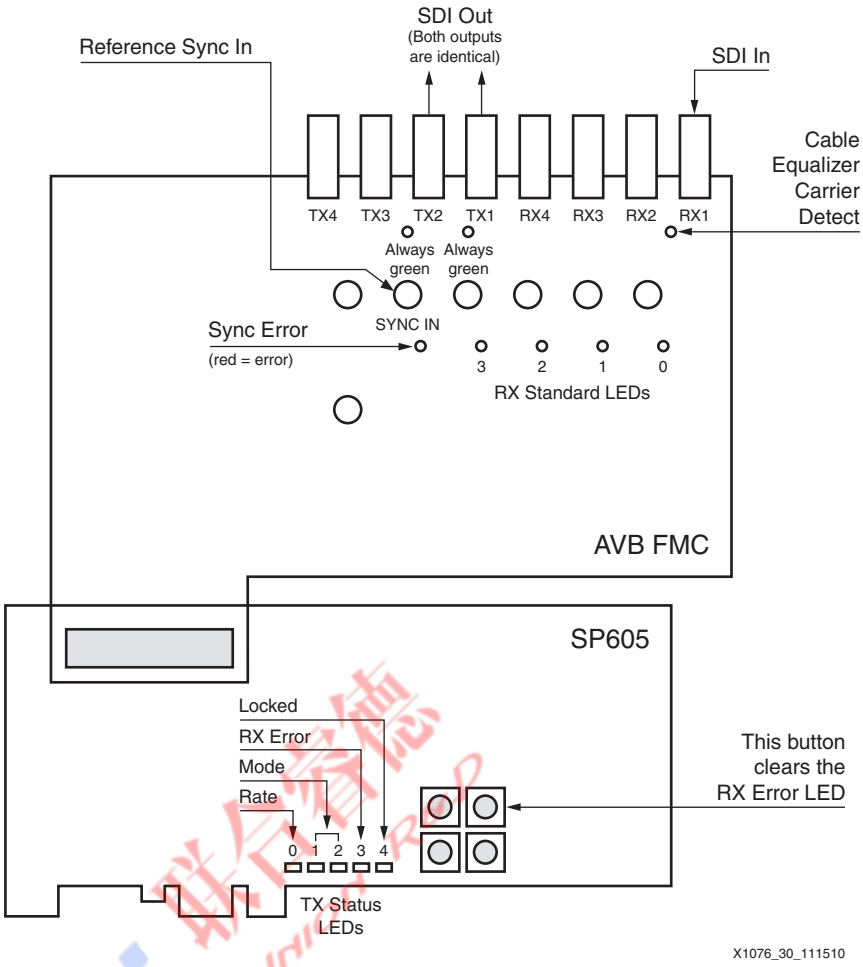


Figure 30: **Triple-Rate SDI Genlock Pass-Through Demonstration**

The GPIO LEDs at the bottom of the SP605 board serve as the status LEDs as shown in Figure 30 and described in Table 21 through Table 23. Table 21 lists the function of the status LEDs on the SP605 board. Table 22 lists the coding of the mode LEDs. Table 23 shows the coding of the line standard LEDs for line standards supported by this demonstration. Refer to Figure 29 for the location of these LEDs. The RX error LED is persistent and stays lit even after the error has gone away. The pushbutton indicated (SW7) clears the LED until the next error is detected.

There are four LEDs called the *RX Standard LEDs* near the topside BNC connectors that identify the line standard recognized by the RX section when it is locked. There are four LEDs called the *Line Standard LEDs* that identify the line standard recognized by the RX section when it is locked.

Table 21: **TX Status LEDs**

LED	Name	Meaning
0 (DS3)	rate	ON: Received line standard is /1.001 rate OFF: Received line standard is /1 rate
1 and 2 (DS4 and DS5)	mode	SDI Mode (see Table 22).

Table 21: **TX Status LEDs *(Cont'd)***

LED	Name	Meaning
3 (DS6)	rx error	Receiver error bit. For SD-SDI mode, this is an EDH error. For HD-SDI and 3G-SDI modes, it is a CRC error. This LED is persistent, and can be cleared with the error clear button (see Figure 30).
4 (DS7)	locked	Indicates that the line standard detector is locked to the incoming video. ON: Locked OFF: Not locked

Table 22: **Receiver Mode LEDs**

LED2	LED1	Mode
OFF	OFF	HD-SDI
OFF	ON	SD-SDI
ON	OFF	3G-SDI
ON	ON	Invalid

Four LEDs called the *Line Standard LED* that identify the line standard recognized by the RX section when it is locked. Table 23 shows the meaning of these LEDs in HD-SDI and 3G-SDI modes. These LEDs are always OFF in SD-SDI mode.

Table 23: **Line Standards**

Mode	STD #	LED 3	LED 2	LED 1	LED 0	LED DS3 (Bit Rate)	Line Standard
HD-SDI	0	OFF	OFF	OFF	OFF	OFF	1035i 30 Hz
		OFF	OFF	OFF	OFF	ON	1035i 29.97 Hz
	2	OFF	OFF	ON	OFF	OFF	1080i 30 Hz or 1080sF 30 Hz
		OFF	OFF	ON	OFF	ON	1080i 29.97 Hz or 1080sF 29.97 Hz
	3	OFF	OFF	ON	ON	OFF	1080i 25 Hz or 1080sF 25 Hz
	4	OFF	ON	OFF	OFF	OFF	1080p 30 Hz
		OFF	ON	OFF	OFF	ON	1080p 29.97 Hz
	5	OFF	ON	OFF	ON	OFF	1080p 25 Hz
	7	OFF	ON	ON	ON	OFF	720p 60 Hz
		OFF	ON	ON	ON	ON	720p 59.94
	9	ON	OFF	OFF	ON	OFF	720p 50 Hz
	10	ON	OFF	ON	OFF	OFF	720p 30 Hz
		ON	OFF	ON	OFF	ON	720p 29.97 Hz
	11	ON	OFF	ON	ON	OFF	720p 25 Hz
3G-SDI Level A	13	ON	ON	OFF	ON	OFF	1080p 60 Hz
		ON	ON	OFF	ON	ON	1080p 59.97 Hz
	14	ON	ON	ON	OFF	OFF	1080p 50 Hz
3G SDI							

Table 23: **Line Standards *(Cont'd)***

Mode	STD #	LED 3	LED 2	LED 1	LED 0	LED DS3 (Bit Rate)	Line Standard
Level B	2	OFF	OFF	ON	OFF	OFF	1080p 60 Hz
		OFF	OFF	ON	OFF	ON	1080p 59.97 Hz
	3	OFF	OFF	ON	ON	OFF	1080p 50 Hz

The output video is transmitted with the same data and timing as the received video. This demonstrates the capability to transmit SD-SDI, HD-SDI, and 3G-SDI. Both SDI TX1 and SDI TX1 connectors on the FMC card output identical SDI signals.

Conclusion

The combination of the FPGA broadcast mezzanine card with the SP605 board is useful for demonstrating the capabilities of the triple-rate SDI reference designs. Using inputs and outputs provided by this hardware platform, the basic functionality of both the receiver and transmitter can be demonstrated using the demonstrations described in this document.

Revision History

The following table shows the revision history for this document.

Date	Version	Description of Revisions
12/15/10	1.0	Initial Xilinx release.

Notice of Disclaimer