

# JESD204 PHY v3.3

## *LogiCORE IP Product Guide*

**Vivado Design Suite**

**PG198 April 5, 2017**

# Table of Contents

## IP Facts

### Chapter 1: Overview

Applications .....	6
Licensing and Ordering Information .....	8

### Chapter 2: Product Specification

Performance .....	9
Resource Utilization .....	9
Port Descriptions .....	9
AXI4-Lite Register Space .....	22
Line Rate Switching .....	37

### Chapter 3: Designing with the Core

JESD204 PHY Configuration Options .....	50
General Design Guidelines .....	51
Clocking .....	52
Resets .....	54
Protocol Description .....	55

### Chapter 4: Design Flow Steps

Customizing and Generating the Core .....	56
Constraining the Core .....	61
Simulation .....	62
Synthesis and Implementation .....	62

### Chapter 5: Example Design

### Chapter 6: Test Bench

AXI Interface .....	70
---------------------	----

### Appendix A: Verification, Compliance, and Interoperability

Simulation .....	71
Hardware Testing .....	71

**Appendix B: Debugging**

Finding Help on Xilinx.com ..... 72  
Debug Tools ..... 73  
Simulation Debug ..... 75  
Hardware Debug ..... 76

**Appendix C: Linux Drivers**

Downloading Linux Drivers ..... 77

**Appendix D: Additional Resources and Legal Notices**

Xilinx Resources ..... 78  
References ..... 78  
Revision History ..... 79  
Please Read: Important Legal Notices ..... 81

## Introduction

The Xilinx® LogiCORE™ IP JESD204 PHY core implements a JESD204B physical interface to simplify sharing serial transceiver channels between transmit and receive cores. This core is not intended to be used standalone and should only be used only in conjunction with the JESD204 core.

## Features

- Designed to JEDEC® JESD204B [Ref 1]
- Supports 1 to 12 lane configurations
- Supports Subclass 0, 1, and 2
- Physical Layer functions provided
- Supports transceiver sharing between TX and RX cores
- Optional AXI interface with AXI to DRP bridges for QPLL and Transceiver access<sup>(1)</sup>
- AXI interface enables line rate switching
- AXI register allows control of selected transceiver signals

LogiCORE IP Facts Table	
<b>Core Specifics</b>	
Supported Device Family <sup>(1)</sup>	UltraScale+™, UltraScale™ Zynq®-7000 All Programmable SoC, 7 Series
Supported User Interfaces	N/A
Resources	<a href="#">Performance and Resource Utilization web page</a>
<b>Provided with Core</b>	
Design Files	RTL
Example Design	Verilog
Test Bench	Verilog
Constraints File	XDC
Simulation Model	Verilog
Supported S/W Driver	N/A
<b>Tested Design Flows<sup>(2)</sup></b>	
Design Entry	Vivado® Design Suite
Simulation	For supported simulators, see the <a href="#">Xilinx Design Tools: Release Notes Guide</a> .
Synthesis	Vivado Synthesis
<b>Support</b>	
Provided by Xilinx at the <a href="#">Xilinx Support web page</a>	

### Notes:

1. For a complete list of supported devices, see the Vivado IP catalog.
2. For the supported versions of the tools, see the [Xilinx Design Tools: Release Notes Guide](#).

1. Not available for GTP transceiver devices.

## Overview

The LogiCORE™ IP JESD204 PHY core implements a JESD204B Physical interface supporting line rates between 1.0 and 12.5 Gb/s on 1 to 12 lanes. Refer to the relevant User Guide for specific line rates supported by each device ([Table 1-1](#)):

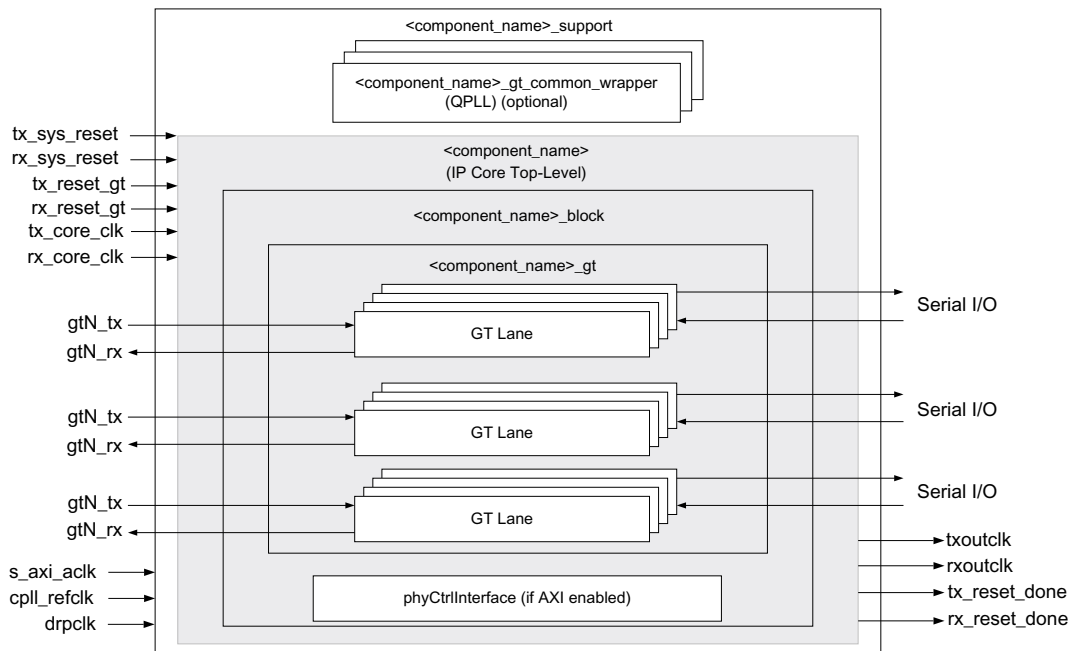
**Table 1-1: Transceiver User Guides**

7-Series	UltraScale	UltraScale+
GTPE2 <a href="#">[Ref 12]</a>	GTHE3 <a href="#">[Ref 10]</a>	GTHE4 <a href="#">[Ref 10]</a> <sup>(1)</sup>
GTXE2 <a href="#">[Ref 11]</a>	GTYE3 <a href="#">[Ref 18]</a>	GTYE4 <a href="#">[Ref 18]</a> <sup>(1)</sup>
GTHE2 <a href="#">[Ref 11]</a>		
GTYE2 – Not Supported		

- For certain UltraScale+ device speed grades, the line rate may also be limited by the maximum frequency specified for TXUSRCLK/RXUSERCLK (core clock) with 40-bit *Interconnect Logic* Data width. The maximum line rate in these devices is TX/RXUSERCLK \* 40. Refer to the relevant device data sheet.

The JESD204 PHY core can be configured with independent transmit and receive line rates.

[Figure 1-1](#) shows the JESD204 PHY core with shared logic in the example design.



**Figure 1-1: JESD204 PHY Block Diagram – Shared Logic in Example Design**

Figure 1-2 shows a block diagram of the JESD204 PHY core with shared logic in the core.

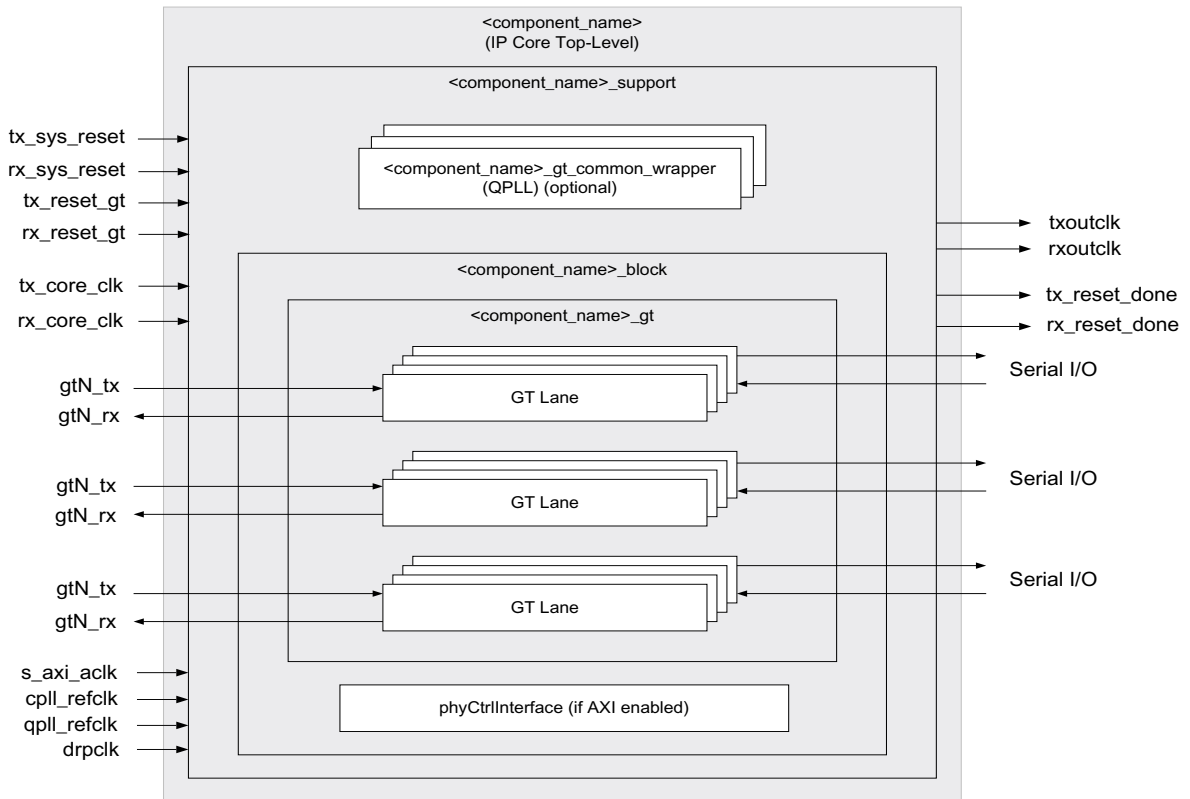


Figure 1-2: JESD204 PHY Block Diagram – Shared Logic in Core

When used in conjunction with the JESD204 core, the JESD204 PHY core is a fully-verified solution design delivered by using the Xilinx® Vivado® Design Suite. In addition, an example design is provided in Verilog. For more information, see the *JESD204 LogiCORE IP Product Guide* (PG066) [Ref 2].

## Applications

The JESD204 PHY core is a sub-core of the JESD204 core. For application information, see the *JESD204 LogiCORE IP Product Guide* (PG066) [Ref 2]. Figure 1-3 shows a JESD204 PHY used in the JESD204 design.

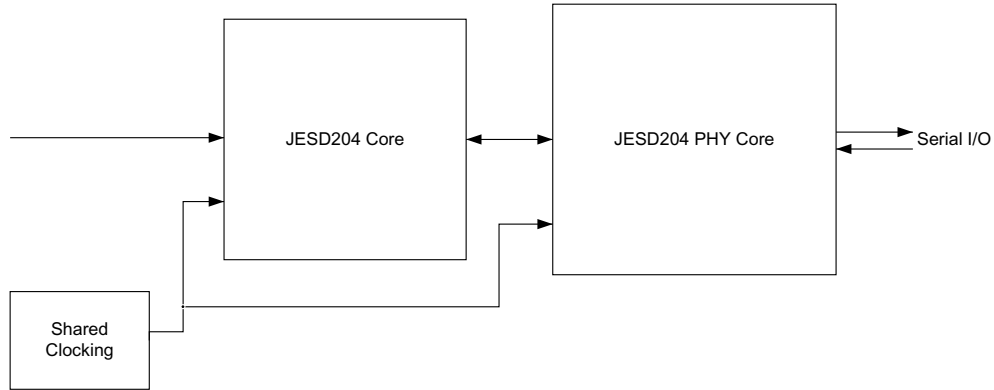


Figure 1-3: JESD204 PHY Used in JESD204 Solution

The JESD204 PHY allows complex transceiver and JESD204 core sharing to be accomplished. Figure 1-4, taken from Vivado IP integrator, shows three JESD204 PHY IPs sharing transceivers between four JESD204 IP cores.

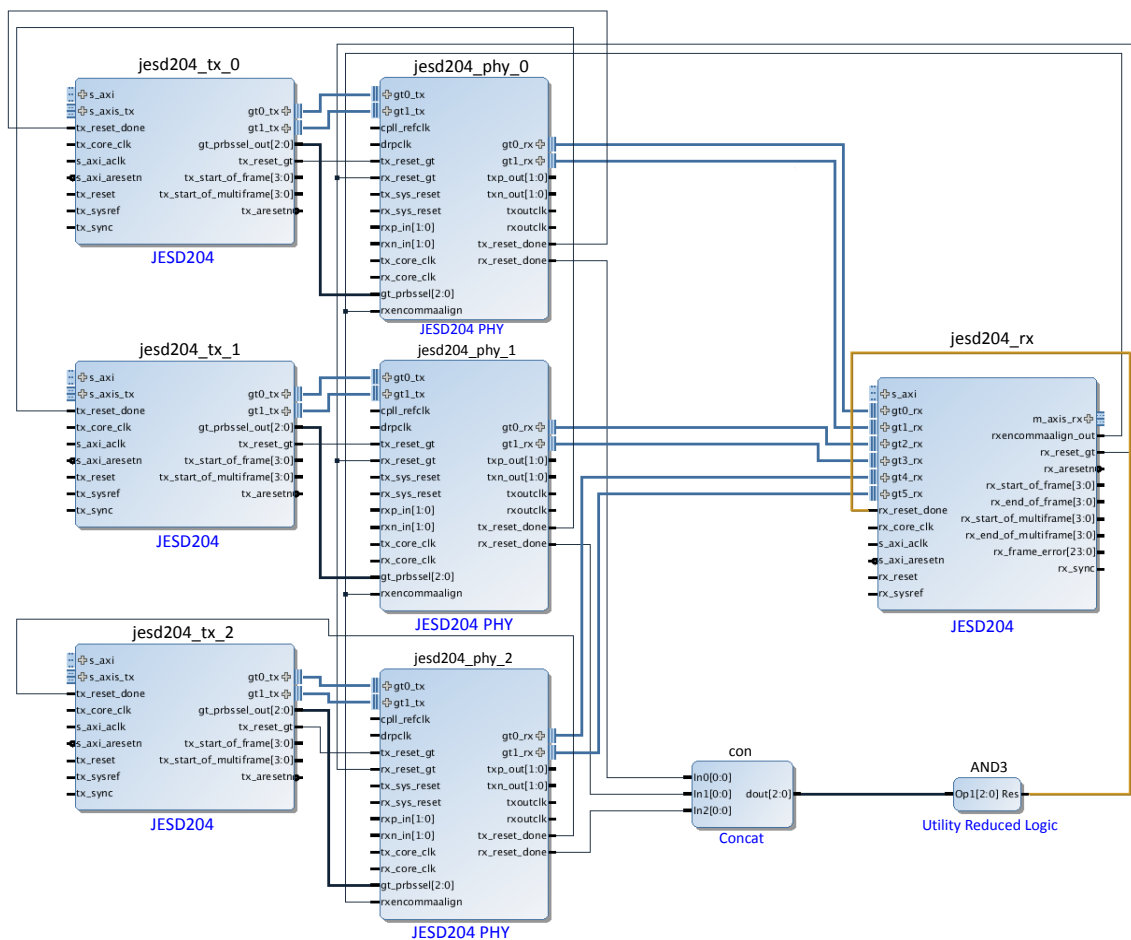


Figure 1-4: IP Integrator Example Design 32-bit

The JESD204 PHY supports many use cases from the most common single JESD204, single JESD204 PHY configuration to the extremely complex multi-JESD204 interleaved JESD204 PHY configurations. See the *JESD204 LogiCORE IP Product Guide* (PG066) [\[Ref 2\]](#).

---

## Licensing and Ordering Information

This Xilinx LogiCORE™ IP module is provided at no additional cost with the Xilinx Vivado Design Suite under the terms of the [Xilinx End User License](#). Information about this and other Xilinx LogiCORE IP modules is available at the [Xilinx Intellectual Property](#) page. For information about pricing and availability of other Xilinx LogiCORE IP modules and tools, contact your [local Xilinx sales representative](#).



# Product Specification

This chapter details the resource utilization and ports for the JESD204 PHY core.

---

## Performance

For details about performance, visit the [Performance and Resource Utilization web page](#).

The core meets the performance specification of JESD204B. The maximum serial line rate is limited by the maximum GTX/GTP/GTH/GTY line rate for the chosen device. See the appropriate device data sheet listed in [References in Appendix D](#).

---

## Resource Utilization

For details about resource utilization, visit the [Performance and Resource Utilization web page](#).

---

## Port Descriptions

This section contains details about the JESD204 PHY ports.

### Clock and Reset Ports

The clock and reset ports available on the delivered core component depend on the Shared Logic selection when customizing the core. [Table 2-1](#) lists the Common Clock and Reset Ports for 32-bit Configurations (Shared Logic in Core).

Table 2-1: Common Clock and Reset Ports - 32-bit Configurations (Shared logic in core)

Signal Name	Direction	Description
<b>Clocks</b>		
txoutclk	Out	Output clock from transceiver. This clock is sourced from the Tx master channel, which is selected from the GUI in the JESD PHY core.
rxoutclk	Out	Output clock from transceiver. This clock is sourced from the Rx master channel, which is selected from the GUI in the JESD PHY core.
tx_core_clk	In	Core clock used to drive txusrclk2 of transceiver. In UltraScale and UltraScale Plus devices txoutclk can be used to drive this port.
rx_core_clk	In	Core clock used to drive rxusrclk2 of transceiver. In UltraScale and UltraScale Plus devices rxoutclk can be used to drive this port.
drp_clk	In	Dynamic Reconfiguration Port (DRP) clock.
cpll_refclk	In	Reference clock for the Transceiver Channel PLL.
qpll_refclk	In	Reference clock for the Quad Common PLL in 7 Series.
qpll0/1_refclk	In	Reference clock for the Quad Common PLL(s) in UltraScale and UltraScale+ devices.
commonM_qpll_clk_out	Out	Clock output from the QPLL (Quad M). Only present when QPLL enabled.
commonM_qpll_refclk_out	Out	Reference clock output from the QPLL (Quad M). Only present when QPLL enabled.
commonM_qpll_lock_out	Out	Lock output from the QPLL (Quad M). Only present when QPLL is enabled.
commonM_pll0_clk_out	Out	Clock output from the PLL (Quad M). Only present when PLL0 is enabled.
commonM_pll0_refclk_out	Out	Reference clock input from the PLL (Quad M). Only present when PLL0 is enabled.
commonM_pll0_lock_out	Out	Lock output from the PLL0 (Quad M). Only present when PLL0 is enabled.
commonM_pll1_clk_out	Out	Clock output from the PLL (Quad M). Only present when PLL0 is enabled.
commonM_pll1_refclk_out	Out	Reference clock output from the PLL (Quad M). Only present when PLL1 is enabled.
commonM_pll1_lock_out	Out	Lock output from the PLL1 (Quad M). Only present when PLL1 is enabled.
commonM_qpll0_clk_out	Out	Clock output from the QPLL (Quad M). Only present when QPLL0 is enabled.
commonM_qpll0_refclk_out	Out	Reference clock output from the QPLL (Quad M). Only present when QPLL0 is enabled.
commonM_qpll0_lock_out	Out	Lock output from the QPLL0 (Quad M). Only present when QPLL0 is enabled.

**Table 2-1: Common Clock and Reset Ports - 32-bit Configurations (Shared logic in core) (Cont'd)**

Signal Name	Direction	Description
commonM_qpll1_clk_out	Out	Clock output from the QPLL (Quad M). Only present when QPLL1 is enabled.
commonM_qpll1_refclk_out	Out	Reference clock output from the QPLL (Quad M). Only present when QPLL1 is enabled.
commonM_qpll1_lock_out	Out	Lock output from the QPLL1 (Quad M). Only present when QPLL1 enabled.
mmcm_locked	Out	Output indicating that the internal MMCM has locked. Signal can be shared with other IP using the same clocks. Only present for GTP devices.
<b>Resets</b>		
tx_reset_gt	In	Tx channel datapath asynchronous logic reset.
rx_reset_gt	In	Rx channel datapath asynchronous logic reset.
tx_sys_reset	In	Tx channel datapath and PLL asynchronous logic reset.
rx_sys_reset	In	Rx channel datapath and PLL asynchronous logic reset.

Table 2-2 lists the Common Clock and Reset Ports for 64-bit Configurations (Shared Logic in Core).

**Table 2-2: Common Clock and Reset Ports - 64-bit Configurations (Shared logic in core)**

Signal Name <sup>(1)</sup>	Direction	Description
<b>Clocks</b>		
tx_core_clk	In	Core clock used to drive txusrclk and txuserclk2 of transceiver. Frequency = serial line rate/66
rx_core_clk	In	Core clock used to drive rxusrclk and rxuserclk2 of transceiver. Frequency = serial line rate/66
drp_clk	In	Dynamic Reconfiguration Port (DRP) clock.
cpll_refclk	In	Reference clock for the Transceiver Channel PLL.
qpll0/1_refclk	In	Reference clock for the Quad Common PLL(s).
commonM_qpll0_clk_out	Out	Clock output from the QPLL (Quad M). Only present when QPLL0 is enabled.
commonM_qpll0_refclk_out	Out	Reference clock output from the QPLL (Quad M). Only present when QPLL0 is enabled.
commonM_qpll0_lock_out	Out	Lock output from the QPLL0 (Quad M). Only present when QPLL0 is enabled.
commonM_qpll1_clk_out	Out	Clock output from the QPLL (Quad M). Only present when QPLL1 is enabled.
commonM_qpll1_refclk_out	Out	Reference clock output from the QPLL (Quad M). Only present when QPLL1 is enabled.
commonM_qpll1_lock_out	Out	Lock output from the QPLL1 (Quad M). Only present when QPLL1 enabled.

Table 2-2: Common Clock and Reset Ports - 64-bit Configurations (Shared logic in core) (Cont'd)

Signal Name <sup>(1)</sup>	Direction	Description
<b>Resets</b>		
tx_reset_gt	In	Tx channel datapath asynchronous logic reset.
rx_reset_gt	In	Rx channel datapath asynchronous logic reset.
tx_sys_reset	In	Tx channel datapath and PLL asynchronous logic reset.
rx_sys_reset	In	Rx channel datapath and PLL asynchronous logic reset.

**Notes:**

1. M = Number of QUADs – 1

Table 2-3 lists the Common Clock and Reset Ports for 32-bit Configurations (Shared Logic in Example Design).

Table 2-3: Common Clock and Reset Ports for 32-bit Configurations (Shared Logic in Example Design)

Signal Name	Direction	Description
<b>Clocks</b>		
tx_core_clk	In	Core clock used to drive txusrclk2 of transceiver. In UltraScale and UltraScale Plus devices txoutclk can be used to drive this port Frequency = serial line rate/40.
rx_core_clk	In	Core clock used to drive rxusrclk2 of transceiver. In UltraScale and UltraScale Plus devices rxoutclk can be used to drive this port Frequency = serial line rate/40.
drp_clk	In	Dynamic Reconfiguration Port (DRP) clock.
tx_usrclk	In	TXUSRCLK input clock to transceiver. Present only on GTP devices.
rx_usrclk	In	RXUSRCLK Input Clock to transceiver. Present only on GTP devices.
commonM_qpll_clk_in	In	Clock input for the QPLL (Quad M). Always present. <b>Note:</b> This port is only applicable to 7-Series devices.
commonM_qpll_refclk_in	In	Reference clock input for the QPLL (Quad M). Always present. <b>Note:</b> This port is only applicable to 7-Series devices.
commonM_pll0_clk_in	In	Reference clock input for the PLL (Quad M). Only present when PLL0 is selected.
commonM_pll0_refclk_in	In	Clock input for the PLL (Quad M). Only present when PLL0 is selected.
commonM_pll1_clk_in	In	Clock input for the PLL (Quad M). Only present when PLL0 is selected.
commonM_pll1_refclk_in	In	Reference clock input for the PLL (Quad M). Only present when PLL1 is selected.

**Table 2-3: Common Clock and Reset Ports for 32-bit Configurations (Shared Logic in Example Design)**

Signal Name	Direction	Description
commonM_qpll0_clk_in	In	Clock input for the QPLL (Quad M). Only present when QPLL0 is selected.
commonM_qpll0_refclk_in	In	Reference clock input for the QPLL (Quad M). Only present when QPLL0 is selected.
commonM_qpll1_clk_in	In	Clock input for the QPLL (Quad M). Only present when QPLL1 is selected.
commonM_qpll1_refclk_in	In	Reference clock input for the QPLL (Quad M). Only present when QPLL1 is selected.
<b>Resets</b>		
tx_reset_gt	In	Tx channel datapath asynchronous logic reset.
rx_reset_gt	In	Rx channel datapath asynchronous logic reset.
tx_sys_reset	In	Tx channel datapath and PLL asynchronous logic reset.
rx_sys_reset	In	Rx channel datapath and PLL asynchronous logic reset.
qpll_reset_out	Out	Reset output from transceiver logic to reset Common Block. Present only when QPLL selected.
qpll0_reset_out	Out	Reset output from transceiver logic to reset QPLL0 Common Block when QPLL0 is selected or the AXI-4 Lite Management Interface is enabled and at least one channel is using either QPLL0 or QLL1. <b>Note:</b> This port is only applicable to UltraScale and UltraScale+ devices.
qpll1_reset_out	Out	Reset output from transceiver logic to reset QPLL1 Common Block when QPLL1 is selected or the AXI-4 Lite Management Interface is enabled and at least one channel is using either QPLL0 or QLL1. <b>Note:</b> This port is only applicable to UltraScale and UltraScale+ devices.
commonM_pll0_reset_out	Out	Reset output from transceiver logic used to reset Common Block. Only present when PLL0 selected.
commonM_pll1_reset_out	Out	Reset output from transceiver logic used to reset Common Block. Only present when PLL1 selected.
mmcm_reset	Out	Reset output from transceiver to reset MMCM. Only present for GTP devices.
mmcm_lock	In	Input from MMCM indicating that the clocks have locked. Only present for GTP devices.

Table 2-4 lists the Common Clock and Reset Ports for 64-bit Configurations (Shared Logic in Example Design).

**Table 2-4: Common Clock and Reset Ports for 64-bit Configurations (Shared Logic in Example Design)**

Signal Name	Direction	Description
<b>Clocks</b>		
drp_clk	In	Dynamic Reconfiguration Port (DRP) clock.
cpll_refclk	In	Reference clock for the Transceiver Channel PLL.
qpll0/1_refclk	In	Reference clock for the Quad Common PLL(s) in UltraScale and UltraScale+ devices.
tx_core_clk	In	Core clock used to drive txusrclk2 of transceiver. Frequency = serial line rate/66.
rx_core_clk	In	Core clock used to drive rxusrclk2 of transceiver. Frequency = serial line rate/66.
commonM_qpll0_clk_in	In	Clock input for the QPLL (Quad M). Only present when QPLL0 is selected.
commonM_qpll0_refclk_in	In	Reference clock input for the QPLL (Quad M). Only present when QPLL0 is selected.
commonM_qpll1_clk_in	In	Clock input for the QPLL (Quad M). Only present when QPLL1 is selected.
commonM_qpll1_refclk_in	In	Reference clock input for the QPLL (Quad M). Only present when QPLL1 is selected.
<b>Resets</b>		
tx_reset_gt	In	Tx channel datapath asynchronous logic reset.
rx_reset_gt	In	Rx channel datapath asynchronous logic reset.
tx_sys_reset	In	Tx channel datapath and PLL asynchronous logic reset.
tx_sys_reset	In	Tx channel datapath and PLL asynchronous logic reset.
qpll0_reset_out	Out	Reset output from transceiver logic to reset QPLL0 Common Block when QPLL0 is selected or the AXI-4 Lite Management Interface is enabled and at least one channel is using either QPLL0 or QLL1.
qpll1_reset_out	Out	Reset output from transceiver logic to reset QPLL1 Common Block when QPLL1 is selected or the AXI-4 Lite Management Interface is enabled and at least one channel is using either QPLL0 or QLL1.

## Transmitter Interface Ports

The transmitter interface ports available on the delivered core component depend on the Datapath Width selection when customizing the core; see [Table 2-5](#) or [Table 2-6](#).

**Table 2-5: TX Parallel Data Interface Ports – Datapath Width = 32-bit configuration**

Signal Name	Direction	Description
gtN_txddata[31:0]	In	Data from TX core. $N = 0 \dots [\text{Lanes} - 1]$
gtN_txcharisk[3:0]	In	Char is K from TX core. $N = 0 \dots [\text{Lanes} - 1]$
gt_prbssel[2:0] <sup>(1)</sup>	In	PRBS select from TX core.

**Notes:**

1. For UltraScale devices, the width of gt\_prbssel is [3:0].

**Table 2-6: TX Parallel Data Interface Ports – Datapath Width = 64-bit configuration**

Signal Name	Direction	Description
gtN_txddata[63:0]	In	Data from TX core. $N = 0 \dots [\text{Lanes} - 1]$
gtN_txheader[1:0]	In	Header flag from TX core

**Table 2-7: TX: Transceiver Serial Interface Ports**

Signal Name	Direction	Description
txp_out[N:0]	Out	Positive differential serial data output $N = (\text{Lanes} - 1)$
txn_out[N:0]	Out	Negative differential serial data output $N = (\text{Lanes} - 1)$

## Receiver Interface Ports

The receiver interface ports available on the delivered core component depend on the Datapath Width selection when customizing the core; see [Table 2-8](#) or [Table 2-9](#).

**Table 2-8: RX Parallel Data Interface Ports – 32-bit configuration**

Signal Name	Direction	Description
gtN_rxddata[31:0]	Out	Data to RX core. $N = 0 \dots [\text{Lanes} - 1]$
gtN_rxcharisk[3:0]	Out	Char is K to RX core. $N = 0 \dots [\text{Lanes} - 1]$
gtN_rxdisperr[3:0]	Out	Disparity error to RX core. $N = 0 \dots [\text{Lanes} - 1]$
gtN_rxnotintable[3:0]	Out	Not In Table to RX core. $N = 0 \dots [\text{Lanes} - 1]$
rxencommalign	IN	Enable comma alignment from RX core.

Table 2-9: RX Parallel Data Interface Ports – 64-bit configuration

Signal Name	Direction	Description
gtN_rxddata[63:0]	Out	Data to RX core. N = 0 ... [Lanes – 1]
gtN_rxheader[1:0]	Out	Header flag to RX core.
gtN_rxblock_sync	Out	Block alignment flag to RX core. N = 0 ... [Lanes – 1]
gtN_rxmismatch	Out	Block mismatched flag to RX core, block had invalid header. N = 0 ... [Lanes – 1]

Table 2-10: RX: Transceiver Serial Data Interface Ports

Signal Name	Direction	Description
rxp_in[N:0]	In	Positive differential serial data input N = (Lanes – 1)
rxn_in[N:0]	In	Negative differential serial data input N = (Lanes – 1)

## Transceiver Debug Interface



**IMPORTANT:** The ports in the Transceiver Control and Status Interface must be driven in accordance with the appropriate GT user guide. Using the input signals listed in [Table 2-11](#) and [Table 2-12](#) might result in unpredictable behavior of the IP core.

The transceiver debug interface (when present) provides access to transceiver control and status pins for debug purposes. See the appropriate transceiver user guide (*UltraScale Architecture GTH Transceivers User Guide* (UG576) [Ref 10], *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) [Ref 11], or *7 Series FPGAs GTP Transceivers User Guide* (UG482) [Ref 12]) for a detailed description of these pins.

Table 2-11: Optional Transceiver Debug Ports (7 Series Devices)

Signal Name <sup>(1)(2)</sup>	Direction	Clock Domain	Description
gtN_drpaddr[8:0]	In	drp_clk	DRP Address Bus
gtN_drpdi[15:0]	In	drp_clk	Data bus for writing configuration data from the FPGA logic resources to the transceiver.
gtN_drpen	In	drp_clk	DRP Enable Signal 0 = No read or write operation performed 1 = Enables a read or write operation
gtN_drpwe	In	drp_clk	DRP Write Enable 0 = Read operation when DEN is 1 1 = Write operation when DEN is 1
gtN_drpdo[15:0]	Out	drp_clk	Data bus for reading configuration data from the GTX/GTH transceiver to the FPGA logic resources.



Table 2-11: Optional Transceiver Debug Ports (7 Series Devices) (Cont'd)

Signal Name <sup>(1)(2)</sup>	Direction	Clock Domain	Description
gtN_drprdy	Out	drp_clk	Indicates operation is complete for write operations and data is valid for read operations.
gtN_loopback[2:0]	In	Async	Transceiver loopback: <ul style="list-style-type: none"> <li>• 000 = No loopback</li> <li>• 001 = Near-end PCS Loopback</li> <li>• 010 = Near-end PMA Loopback</li> <li>• 100 = Far-end PMA Loopback</li> <li>• 110 = Far-end PCS Loopback</li> </ul> <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gtN_txpostcursor[4:0]	In	tx_core_clock	Transmit Differential Driver control. (TX only) <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gtN_txprecursor[4:0]	In	tx_core_clock	Transmit Differential Driver control. (TX only) <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gtN_txdiffctrl[3:0]	In	Async	Transmit Differential Driver control. (TX only) <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gtN_txpolarity	In	tx_core_clock	Transmit polarity control. (TX only) <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gtN_rxpolarity	In	rx_core_clock	Receive polarity control. (RX only) <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gtN_cpplllock_out	Out	Async	Active-High signal indicating that the channel PLL has locked to the input reference clock.
gtN_eyescandataerror_out	Out	Async	Asserted when an EYESCAN error occurs.
gtN_eyescanreset_in	In	Async	This port is pulsed High to initiate the EYESCAN reset process.
gtN_eyescantrigger_in	In	rx_core_clock	A High on this port causes an EYESCAN trigger event.
gtN_rxbufreset_in	In	Async	This port is driven High and then deasserted to start the RX elastic buffer reset process.
gtN_rxbufstatus_out[2:0]	Out	rx_core_clock	RX Elastic Buffer Status
gtN_rxbyteisaligned_out	Out	rx_core_clock	RX Byte Alignment Status
gtN_rxbyterealign_out	Out	rx_core_clock	RX Byte Alignment has changed.
gtN_rxcdrhold_in	In	Async	Hold the CDR control loop frozen.
gtN_rxcommadet_out	Out	rx_core_clock	RX Comma detect out
gtN_rxdfelpmreset_in	In	Async	DFE reset <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.

Table 2-11: Optional Transceiver Debug Ports (7 Series Devices) (Cont'd)

Signal Name <sup>(1)(2)</sup>	Direction	Clock Domain	Description
gtN_rxlpmen_in	In	Async	LPM mode enable <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gtN_rxmonitorout_out	Out	Async	RX Monitor Out
gtN_rxmonitorsel_in	In	Async	RX Monitor Out mode select
gtN_rxpcsreset_in	In	Async	PCS Reset
gtN_rxpdp_in[1:0]	In	Async	RX Power Down <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gtN_rxpmareset_in	In	Async	PMA Reset
gtN_rxprbscreset_in	In	rx_core_clock	RX PRBS Counter Reset
gtN_rxprbserr_out	Out	rx_core_clock	RX PRBS Error Detect
gtN_rxprbssel_in	In	rx_core_clock	RX PRBS Select
gtN_rxresetdone_out	Out	rx_core_clock	RX Reset Done
gtN_rxstatus_out[2:0]	Out	rx_core_clock	Encodes RX status and error codes
gtN_txbufstatus_out[1:0]	Out	tx_core_clock	TX Elastic Buffer Status
gtN_txpcsreset_in	In	Async	TX PCS Reset
gtN_txinhibit	In	tx_core_clock	TX Inhibit
gtN_txpd_in	In	tx_core_clock	TX Power Down <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gtN_txpmareset_in	In	Async	TX PMA Reset
gtN_txprbsforceerr_in	In	tx_core_clock	TX PRBS Force Error
gtN_txresetdone_out	Out	tx_core_clock	TX Reset Done
gtN_rxlpmhfhold_in	In	rx_core_clock	(GTP Only) LPM Mode Control
gtN_rxlpmhfoverden_in	In	rx_core_clock	(GTP Only) LPM Mode Control
gtN_rxlplfhold_in	In	rx_core_clock	(GTP Only) LPM Mode Control

**Notes:**

1. N is the number of the transceiver channels.
2. If you are migrating from a 7 series to an UltraScale architecture-based device, the prefixes of the optional transceiver debug ports for single-lane cores are changed from gt0, gt1 to gt, and the postfix \_in and \_out are dropped. For multi-lane cores, the prefixes of the optional transceiver debug ports gt(n) are aggregated into a single port (see Table 2-12).

Table 2-12: Optional Transceiver Debug Ports (UltraScale Architecture-Based Devices)

Signal Name <sup>(1)</sup>	Direction	Clock Domain	Description
gtN_drpaddr [9:0]	In	drp_clk	DRP Address Bus <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled. <b>Note:</b> GTH=[8:0], GTY=[9:0]
gtN_drpdi [15:0]	In	drp_clk	Data bus for writing configuration data from the FPGA logic resources to the transceiver. <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gtN_drpen	In	drp_clk	DRP Enable Signal 0 = No read or write operation performed 1 = Enables a read or write operation <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gtN_drpwe	In	drp_clk	DRP Write Enable 0 = Read operation when DEN is 1 1 = Write operation when DEN is 1 <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gtN_drpdo [15:0]	Out	drp_clk	Data bus for reading configuration data from the GTX/GTH transceiver to the FPGA logic resources. <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gtN_drprdy	Out	drp_clk	Indicates operation is complete for write operations and data is valid for read operations. <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gt_txpmareset [(LANES-1):0]	In	Async	This port is pulsed High to start the TX PMA reset process.
gt_txpcsreset [(LANES-1):0]	In	Async	This port is pulsed High to start the TX PCS reset process.
gt_txresetdone [(LANES-1):0]	Out	tx_core_clock	A High on this port indicates that the TX reset process has completed.
gt_rxpmareset [(LANES-1):0]	In	Async	This port is pulsed High to start the RX PMA reset process.
gt_rxpcsreset [(LANES-1):0]	In	Async	This port is pulsed High to start the RX PCS reset process.
gt_rxbufreset [(LANES-1):0]	In	Async	This port is driven High and then deasserted to start the RX elastic buffer reset process.
gt_rxpmaresetdone [(LANES-1):0]	Out	Async	A High on this port indicates that the RX PMA reset process has completed.
gt_rxresetdone [(LANES-1):0]	Out	rx_core_clock	A High on this port indicates that the RX reset process has completed.
gt_txbufstatus [(LANES*2)-1:0]	Out	tx_core_clock	Elastic Buffer Status

Table 2-12: Optional Transceiver Debug Ports (UltraScale Architecture-Based Devices) (Cont'd)

Signal Name <sup>(1)</sup>	Direction	Clock Domain	Description
gt_rxbufstatus [(LANES*3)-1:0]	Out	rx_core_clock	RX Elastic Buffer Status
gt_cpplllock [(LANES-1):0]	Out	refclk	Active-High signal indicating that the channel PLL has locked to the input reference clock.
gt_rxrate [(LANES*3)-1:0]	In	rx_core_clock	Link signaling rate control
gt_eyesctrigger [(LANES-1):0]	In	rx_core_clock	A High on this port causes an EYESCAN trigger event.
gt_eyescanreset [(LANES-1):0]	In	Async	This port is pulsed High to initiate the EYESCAN reset process.
gt_eyescanerror [(LANES-1):0]	Out	Async	Asserted when an EYESCAN error occurs.
gt_loopback [(LANES*3)-1:0]	In	Async	Transceiver loopback: <ul style="list-style-type: none"> <li>• 000 = No loopback</li> <li>• 001 = Near-end PCS Loopback</li> <li>• 010 = Near-end PMA Loopback</li> <li>• 100 = Far-end PMA Loopback</li> <li>• 110 = Far-end PCS Loopback</li> </ul> <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gt_rxpolarity [(LANES-1):0]	In	rx_core_clock	Set High to invert the incoming serial data. <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gt_txpolarity [(LANES-1):0]	In	tx_core_clock	Set High to invert the outgoing serial data. <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gt_rxdfelpmreset [(LANES-1):0]	In	Async	Reset for the LPM and DFE datapath. <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gt_rxlpmen [(LANES-1):0]	In	Async	Set to 1 to select the LPM datapath. <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gt_txprecursor [(LANES*5)-1:0]	In	tx_core_clock	Transmitter pre-cursor pre-emphasis control. <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gt_txpostcursor [(LANES*5)-1:0]	In	tx_core_clock	Transmitter post-cursor pre-emphasis control. <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gt_txdiffctrl [(LANES*4)-1:0]	In	Async	Driver swing control. <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gt_txprbsforceerr [(LANES-1):0]	In	tx_core_clock	Set High to drive errors into the PRBS transmitter.

Table 2-12: Optional Transceiver Debug Ports (UltraScale Architecture-Based Devices) (Cont'd)

Signal Name <sup>(1)</sup>	Direction	Clock Domain	Description
gt_txinhibit [LANES-1:0]	In	tx_core_clock	TX Inhibit
gt_pcsrsvdin [(LANES*16)-1:0]	In	Async	16 bits per lane. Bit [2] is DRP reset. Reading read-only registers while the XCLK is not toggling (e.g., during reset or change of reference clocks), causes the DRP to not return a DRPRDY signal and prevent further DRP transactions. In such an event, PCSRSVDIN[2] must be pulsed to reset the DRP interface before initiating further DRP transactions.
gt_rxprbssel [(LANES*4)-1:0]	In	rx_core_clock	Receiver PRBS checker test pattern control.
gt_rxprbserr [(LANES-1):0]	In	rx_core_clock	A High on this port indicates that PRBS errors have occurred.
gt_rxprbscntreset [(LANES-1):0]	In	rx_core_clock	Reset the PRBS error counter
gt_rxcdrhold [(LANES-1):0]	In	Async	Hold the CDR control loop frozen
gt_dmonitorout [(LANES*15-1):0]	Out	Async	Digital Monitor Output Bus
gt_rxdisperr [(LANES*4-1):0]	Out	rx_core_clock	Receiver disparity error indicator
gt_rxnotintable [(LANES*4-1):0]	Out	rx_core_clock	Receiver not in table error indicator
gt_rxcommadet [(LANES-1):0]	Out	rx_core_clock	A High on this port indicates that the comma alignment block has detected a valid comma.
gt_rxpd [(LANES*2-1):0]	In	Async	RX Power Down 00=Normal Operation 11=Lowest power mode <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.
gt_txpd [(LANES*2-1):0]	In	tx_core_clock	TX Power Down 00=Normal Operation 11=Lowest power mode <b>Note:</b> Not present when AXI4-Lite Management Interface is enabled.

**Notes:**

1. N is the number of the transceiver channels.

## AXI4-Lite Register Space

The JESD204 PHY core is configured using an AXI4-Lite Register Interface. The register map is shown in [Table 2-13](#).

**Note:** The AXI register interface is not available for GTP transceiver devices.

**Table 2-13: Register Address Map**

AXI4-Lite Address	Register Name	Access Type
0x000	Version	R
0x004	IP Configuration	R
0x008	Number of Common Interfaces	R
0x00C	Number of Transceiver Interfaces	R
0x014	Timeout Enable	R/W
0x018	Reserved	–
0x01C	Timeout Value	R/W
0x020	Common Interface Selector	R/W
0x024	GT Interface Selector	R/W
0x028 to 0x07F	Reserved	–
0x030	Transceiver Master Channel for RX	R
0x034	Transceiver Master Channel for TX	R
0x038	RX Interface is 64-bit	R
0x03C	TX Interface is 64-bit	R
0x080	PLL Status	R
0x084 to 0x0FF	Reserved	–
0x90	RXLINERATE	R
0x98	RXREFCLK	R
0x9C	RXXMULT	R
0xA0	RXPLL	R
0xB0	TXLINERATE	R
0xB8	TXREFCLK	R
0xBC	TXXMULT	R
0xC0	TXPLL	R
0xD0	SW_CAPABLE	R
0xD4	INS_LOSS	R
0xD8	EQUALISATION	R
0xE0	MIN_RATE	R
0xE4	MAX_RATE	R

Table 2-13: Register Address Map (Cont'd)

AXI4-Lite Address	Register Name	Access Type
<b>Common/Transceiver DRP Control</b>		
0x104/0x204	Common/Transceiver DRP Address	R/W
0x108/0x208	Common/Transceiver DRP Write Data	R/W
0x10C/0x20C	Common/Transceiver DRP Read Data	R
0x110/0x210	Common/Transceiver DRP Reset	R/W
0x114/0x214	Common/Transceiver DRP Access Status	R
0x118/0x218	Reserved	–
0x11C/0x21C	Common/Transceiver DRP Access Complete	R
0x120 to 0x1FF/ 0x220 to 0x2FF	Reserved	–
<b>Common QPLL Control</b>		
0x304	QPLL Power Down (QPLL0 UltraScale)	R/W
0x308	QPLL1 Power Down (UltraScale Only)	R/W
0x30C to 0x3FF	Reserved	–
<b>Transceiver Control – Bank 1</b>		
0x404	RXPD (RX Power Down)	R/W
0x408	CPLLPD (CPLL Power Down)	R/W
0x40C	Transmit PLL Clock Select(1)	R/W
0x410	Receive PLL Clock Select(1)	R/W
0x414	TX Postcursor	R/W
0x418	TX Precursor	R/W
0x41C	Loopback	R/W
0x420	TX System Reset(1)	R/W
0x424	RX System Reset(1)	R/W
0x428 to 0x4FF	Reserved	–
<b>Transceiver Control – Bank 2</b>		
0x504	TXPD	R/W
0x508	TXDIFFCTRL	R/W
0x50C	TXINHIBIT	R/W
0x510	TXPOLARITY	R/W
0x514 to 0x5FF	Reserved	–
<b>Transceiver Control – Bank 3</b>		
0x604	RXPOLARITY	R/W
0x608	RXLPMEN	R/W
0x60C	RXDFELPMRESET	R/W

Table 2-13: Register Address Map (Cont'd)

AXI4-Lite Address	Register Name	Access Type
0x610	<a href="#">RX Invalid SYNC Header Max</a> (RX 64 bit only)	R/W
0x614 to 0xFF	Reserved	–

Table 2-14: Version

Bits	Default Value	Description
31:24	–	Version: Major
23:16	–	Version: Minor
15:8	–	Version: Revision
7:0	–	Reserved (read 0x00)

### Register Address Map

Table 2-15: IP Configuration

Bits	Default Value	Description
31:24	–	FPGA Type: 0 = 7 series 1 = UltraScale 2 = UltraScale+ All other values are reserved.
23:16	–	Speed Grade: 10 = 1 11 = 1L 12 = 1H 13 = 1HV 14 = 1LV 20 = 2 21 = 2L 22 = 2LV 30 = 3 All other values are reserved.



Table 2-15: IP Configuration (Cont'd)

Bits	Default Value	Description
15:8	–	Package: 1 = rf 2 = fl 3 = ff 4 = fb 5 = hc 6 = fh 7 = cs 8 = cp 9 = ft 10 = fg 11 = sb 12 = rb 13 = rs 14 = cl 15 = sf 16 = ba 17 = fa All other values are reserved.
7:0	–	Transceiver Type: 2 = GTXE2 3 = GTHE2 5 = GTHE3 6 = GTYE3 7 = GTHE4 All other values are reserved.

Register Address Map

Table 2-16: Number of Common Interfaces

Bits	Default Value	Description
31:0	–	Reading this register returns the number of GT_COMMON blocks in the core. Normally one common block is included per four transceivers. See the following documents for details: <ul style="list-style-type: none"> <li>◦ <i>UltraScale Architecture GTH Transceivers User Guide</i> (UG576) [Ref 10]</li> <li>◦ <i>7 Series FPGAs GTX/GTH Transceivers User Guide</i> (UG476) [Ref 11]</li> <li>◦ <i>UltraScale Architecture GTY Transceiver User Guide</i> (UG578) [Ref 18]</li> </ul> The number returned here can be used by software to loop round the correct number of times to configure all the QPLLs in the core using the Common DRP control mailbox and the Common PLL control registers. The valid range for the “Common interface select” register <code>cmm_interface_sel</code> (0x020), used to choose which Common PLL is being accessed, is 0 to N-1 where N is the value contained in this register.

Register Address Map

Table 2-17: Number of Transceiver Interfaces

Bits	Default Value	Description
31:0	–	Reading this register returns the number of GT_CHANNEL blocks (same as the number of lanes). The number returned here can be used by software to loop round the correct number of times to configure all the transceivers in the core using the Transceiver DRP control mailbox and the Transceiver control register banks.  The valid range for the “Transceiver interface select” register <code>gt_interface_sel</code> (0x024), used to choose which transceiver is being accessed, is 0 to N-1 where N is the value contained in this register.

[Register Address Map](#)

Table 2-18: Timeout Enable

Bits	Default Value	Description
31:1	–	Reserved
0	1	Enable the AXI4-Lite timeout. This ensures that transactions to the transceiver registers do not lock the AXI4-Lite bus which can happen if the transceiver is in reset or not being clocked when an access is attempted for example.  If a timeout occurs during an AXI transaction, it is indicated on the bresp bus as a SLVERR response, allowing the firmware to act accordingly.

[Register Address Map](#)

Table 2-19: Timeout Value

Bits	Default Value	Description
31:12	–	Reserved
11:0	128	Set the number of AXI clock cycles to wait before terminating the AXI4-Lite access without completing.  If using timeout, the value must be modified according to the relationship between the AXI and DRP clock. The IP handles the DRP access by stretching the AXI interface response until it is completed. If the DRP clock is much slower than the AXI clock, this results in an unintentional timeout of the value is not increased. The timeout counts up so this value should be programmed with 4096-timeout.  The complete timeout length is this value + 2 cycles. Valid timeout value are 0 to 4094. Care must be taken to not set to small a value. Base setting this value on the slowest clock of the set, <code>rx_core_clk</code> , <code>tx_core_clk</code> , and <code>drpclk</code> .

[Register Address Map](#)

Table 2-20: Common Interface Selector

Bits	Default Value	Description
31:2	–	Reserved
1:0	–	Set the number corresponding to the Common DRP control mailbox or the Common PLL control registers to be accessed. The range is 0 to N-1, where N is the value returned from the Number of Common Interfaces register (0x008).

[Register Address Map](#)

Table 2-21: GT Interface Selector

Bits	Default Value	Description
31:4	–	Reserved
3:0	–	Set the number corresponding to the Transceiver DRP control mailbox or the Transceiver control register bank to be accessed. The range is 0 to N-1, where N is the value returned from the Number of Transceiver Interfaces register (0x00C).

[Register Address Map](#)

Table 2-22: Transceiver Master Channel for RX

Bits	Default Value	Description
31:4	–	Reserved
3:0	–	Returns master transceiver channel for RX.

[Register Address Map](#)

Table 2-23: Transceiver Master Channel for TX

Bits	Default Value	Description
31:4	–	Reserved
3:0	–	Returns master transceiver channel for TX.

[Register Address Map](#)

Table 2-24: RX Interface is 64-bit

Bits	Default Value	Description
31:1	–	Reserved
0	–	Returns operating mode of RX 1 = 64 bit mode 0 = 32 bit mode

[Register Address Map](#)

Table 2-25: TX Interface is 64-bit

Bits	Default Value	Description
31:1	–	Reserved
0	–	Returns operating mode of TX 1 = 64 bit mode 0 = 32 bit mode

[Register Address Map](#)

Table 2-26: PLL Status

Bits	Default Value	Description
31:5	–	Reserved
4	–	Returns 1 when a transmit reset is in progress.
3	–	Returns 1 when a receive reset is in progress.
2	–	Returns 0 when all the CPLLs are locked.
1	–	Returns 0 when all the QPLLs (7 series) or QPLL0s (UltraScale) are locked.
0	–	Returns 0 when all the QPLL1s are locked (UltraScale only, always returns 0 for 7 series devices).

[Register Address Map](#)

Table 2-27: RXLINERATE

Bits	Default Value	Description
31:0	–	The default value of RX line rate that the core was generated to use. Value in kHz.

[Register Address Map](#)

Table 2-28: RXREFCLK

Bits	Default Value	Description
31:0	–	The default value of RX refclk frequency that the core was generated to use. Value in kHz.

[Register Address Map](#)

Table 2-29: RXXMULT

Bits	Default Value	Description
31:0	–	The default value of RX xMult, the ratio between RX linerate and RX refclk, that the core was generated to use. Value = linerate/refclk * 1000.

[Register Address Map](#)

Table 2-30: **RXPLL**

Bits	Default Value	Description
1:0	–	The default PLL for the RX path that the core was generated to use. 00 = CPLL 10 = QPLL1 (UltraScale Only) 11 = QPLL (7 series) QPLL0 (UltraScale)

[Register Address Map](#)

Table 2-31: **TXLINERATE**

Bits	Default Value	Description
31:0	–	The default value of TX line rate that the core was generated to use. Value in kHz.

[Register Address Map](#)

Table 2-32: **TXREFCLK**

Bits	Default Value	Description
31:0	–	The default value of TX refclk frequency that the core was generated to use. Value in kHz.

[Register Address Map](#)

Table 2-33: **TXXMULT**

Bits	Default Value	Description
31:0	–	The default TX xMult, the ratio between TX linerate and TX refclk, that the core was generated to use. Value = linerate/refclk * 1000.

[Register Address Map](#)

Table 2-34: **TXPLL**

Bits	Default Value	Description
1:0	–	The default PLL for the TX path that the core was generated to use. 00 = CPLL 10 = QPLL1 (UltraScale Only) 11 = QPLL (7 series) QPLL0 (UltraScale)

[Register Address Map](#)

Table 2-35: SW\_CAPABLE

Bits	Default Value	Description
1:0	–	Line rate switching capability of the generated core 0 = Generated core not capable of line rate switching 1 = Generated core capable of line rate switching

[Register Address Map](#)

Table 2-36: INS\_LOSS

Bits	Default Value	Description
31:0	–	The default RX Insertion loss that the core was generated to support. Value = setting in dB from GUI * 1000

[Register Address Map](#)

Table 2-37: EQUALISATION

Bits	Default Value	Description
1:0	–	The default Equalization mode that the core was generated to use 0: Auto (Equalization mode set based on insertion loss) 1 = Low_Loss (Equalization mode LPM) 2 = High_Loss (Equalisation mode DFE)

[Register Address Map](#)

Table 2-38: MIN\_RATE

Bits	Default Value	Description
31:0	–	The minimum line rate that the core was generated to use. Value in KHz

[Register Address Map](#)

Table 2-39: MAX\_RATE

Bits	Default Value	Description
31:0	–	The minimum line rate that the core was generated to use. Value in KHz

[Register Address Map](#)

## Common/Transceiver DRP Control

The DRP interface provides an “indirect mailbox” mechanism for read/write to multiple DRPs. Bits[31:30] of the DRP Address are used to auto-initiate a read or write of the DRP interface. A Status register is provided to allow you to check the access has completed without error. There are two mailboxes, one at 0x1XX for the COMMON DRPs and one at 0x2XX for the transceiver DRPs.

For accesses to the Common DRP mailboxes, ensure that register `cmm_interface_sel` (0x020) is programmed with index of the Common DRP that is required. The range is 0 to N-1, where N is the value returned in the “Number of Common interfaces” register (0x008).

For accesses to the Transceiver DRP mailboxes, ensure that register `gt_interface_sel` (0x024) is programmed with the index of the Transceiver DRP that is required. The range is 0 to M-1, where M is the value returned in the “Number of Transceiver interfaces” register (0x00C).

**Table 2-40: Common/Transceiver DRP Address**

Bits	Default Value	Description
31	–	Set to 1 to perform a write to the DRP.
30	–	Set to 1 to perform a read from the DRP.
29:0	–	DRP register address (See <i>UltraScale Architecture GTH Transceivers User Guide</i> (UG576) [Ref 10] or <i>7 Series FPGAs GTX/GTH Transceivers User Guide</i> (UG476) [Ref 11] for a complete DRP address map).

**Notes:**

- The lower Bits[29:0] can be read/written without triggering a DRP access allowing the firmware to test the address value if required. If both upper bits are set the access is ignored as DRP cannot be read/written at the same time. This will result in 0x0 being read in these upper two bits.

[Register Address Map](#)

**Table 2-41: Common/Transceiver DRP Write Data**

Bits	Default Value	Description
31:16	–	Reserved, the DRP registers are all 16 bits
15:0	–	Data to be written to the selected DRP register

[Register Address Map](#)

**Table 2-42: Common/Transceiver DRP Read Data**

Bits	Default Value	Description
31:16	–	Reserved, the DRP registers are all 16 bits
15:0	–	Data read back from the selected DRP register

[Register Address Map](#)

Table 2-43: Common/Transceiver DRP Reset

Bits	Default Value	Description
31:1	–	Reserved
0	0	Write a 1 to reset the DRP interface. Self-clearing. UltraScale only.

[Register Address Map](#)

Table 2-44: Common/Transceiver DRP Access Status

Bits	Default Value	Description
31:3	–	Reserved
2	–	Access Type 0 = read 1 = write This register is only set when “DRP Access in Progress” bit is set to allow a read of all zeros check of the register for completion. Sticky on Timeout Error, updated on new DRP access.
1	–	Reserved
0	–	DRP Access in Progress Set on a write of 1 to either of the top two bits of DRP Address register and auto cleared when DRP data is valid.

[Register Address Map](#)

Table 2-45: Common/Transceiver DRP Access Complete

Bits	Default Value	Description
31:1	–	Reserved
0	–	When asset to 1, the AXI4-Lite does not complete the final write to the DRP Address register until the DRP access has completed. When set, there is no need to poll the DRP Access in Progress bit of the DRP Access Status register.

### DRP Access Sequence Example

Both the number of transceivers and common block present are held in the Configuration register bank and can be used as maximum values in the configuration loops, saving firmware regeneration.

As an example, to program each COMMON PLL block:

1. Read how many COMMON blocks exist in the system (`num_com = AXI read 0x8`).
2. Loop over each COMMON `i = 0; i < num_com; i++`.
3. Set the Common Interface number to access (0 to `(num_com - 1)`) (AXI Write `i` to `0x020`). To write to the DRP:
  - a. Program the value to write over the DRP interface (AXI Write `<value> 0x108`).



- b. Program the DRP address value also setting the write flag
- 4. To set the write flag, OR the address with 0x8000\_0000 (AXI Write (<address> | 0x8000\_0000) to 0x104). To read from the DRP,
  - a. Program the DRP address value and also set the read flag.
- 5. To set the read flag, OR the address with 0x4000\_0000 (AXI Write (<address> | 0x4000\_0000) to 0x104).

## Common QPLL Control

For accesses to the Common QPLL Control registers ensure that register `cmm_interface_sel` (0x020) is programmed with index of the Common QPLL that is required. The range is 0 to N-1, where N is the value returned in the "Number of Common interfaces" register (0x008).

Table 2-46: QPLL Power Down (QPLL0 UltraScale)

Bits	Default Value	Description
0	0	1 = Power Down QPLL (7 series) QPLL0 (UltraScale)

[Register Address Map](#)

Table 2-47: QPLL1 Power Down (UltraScale Only)

Bits	Default Value	Description
0	0	1 = Power Down QPLL1 (UltraScale Only)

[Register Address Map](#)

## Transceiver Control

The following controls are split into three banks internally to minimize the clock domain crossings required for each interface. This does require three separate writes to the select registers. The AXI read/write manages the clock domain crossing, with the result that the AXI accesses are longer than standard.

For accesses to the Transceiver register banks 1 to 3, ensure that register `gt_interface_sel` (0x024) is programmed with the index of the Transceiver that is required. The range is 0 to M-1, where M is the value returned in the "Number of Transceiver interfaces" register (0x00C).

Table 2-48: RXPDP

Bits	Default Value	Description
1:0	0	Power up or down the RX of the GT transceiver. 00 = Power state for normal operation. 11 = Power saving state with lowest power.

Table 2-49: CPLLPD

Bits	Default Value	Description
0	0	1 = Power Down CPLL

[Register Address Map](#)

 Table 2-50: Transmit PLL Clock Select<sup>(1)</sup>

Bits	Default Value	Description
1:0	0	Selects the PLL to drive the TX datapath: 00 = CPLL 10 = QPLL1 (UltraScale Only) 11 = QPLL (7 series) QPLL0 (UltraScale)

1. The PHY IP core contains a single *Transmit PLL Clock Select* register which can be programmed to any index.

 Table 2-51: Receive PLL Clock Select<sup>(1)</sup>

Bits	Default Value	Description
1:0	0	Selects the PLL to drive the RX datapath: 00 = CPLL 10 = QPLL1 (UltraScale Only) 11 = QPLL (7 series) QPLL0 (UltraScale)

1. The PHY IP core contains a single *Receive PLL Clock Select* register which can be programmed to any index.

[Register Address Map](#)

Table 2-52: TX Postcursor

Bits	Default Value	Description
4:0	0	Driver Swing Control. Refer to the relevant transceiver user guide <a href="#">[Ref 10]</a> , <a href="#">[Ref 11]</a> , <a href="#">[Ref 12]</a> .

[Register Address Map](#)

Table 2-53: TX Precursor

Bits	Default Value	Description
4:0	0	Transmitter precursor TX pre-emphasis control. Refer to the relevant transceiver user guide <a href="#">[Ref 10]</a> , <a href="#">[Ref 11]</a> , <a href="#">[Ref 12]</a> .

[Register Address Map](#)

Table 2-54: Loopback

Bits	Default Value	Description
2:0	0	Loopback modes are specialized configurations of the transceiver datapath where the traffic stream is folded back to the source. 000 = Normal operation 001 = Near-end PCS Loopback 010 = Near-end PMA Loopback 011 = Reserved 100 = Far-end PMA Loopback 101 = Reserved 110 = Far-end PCS Loopback

[Register Address Map](#)

 Table 2-55: TX System Reset<sup>(1)</sup>

Bits	Default Value	Description
0	0	Reset all the TX logic. Writing 1 to this bit will reset both the TX channel datapath logic and the PLL selected for use by the TX. This bit does not self clear.

1. The PHY IP core contains a single *TX System Reset* register which can be programmed to any index.

[Register Address Map](#)

 Table 2-56: RX System Reset<sup>(1)</sup>

Bits	Default Value	Description
0	0	Reset all the RX logic. Writing 1 to this bit will reset both the RX channel datapath logic and the PLL selected for use by the RX. This bit does not self clear.

1. The PHY IP core contains a single *RX System Reset* register which can be programmed to any index.

[Register Address Map](#)

Table 2-57: TXPD

Bits	Default Value	Description
1:0	0	Power up or down the TX of the GT transceiver. 00 = Power state for normal operation. 11 = Power saving state with lowest power.

Table 2-58: TXDIFFCTRL

Bits	Default Value	Description
N:0 <sup>(1)</sup>	4'b1100	Driver Swing Control.

1. 'N' is transceiver dependent - refer to the relevant transceiver user guide.

[Register Address Map](#)

Table 2-59: TXINHIBIT

Bits	Default Value	Description
0	0	When High, this signal blocks transmission of TXDATA and forces MGTHTXP to 0 and MGTHTXN to 1.

[Register Address Map](#)

Table 2-60: TXPOLARITY

Bits	Default Value	Description
0	0	The TXPOLARITY port is used to invert the polarity of outgoing data. 0 = Not inverted. TXP is +ve, and TXN is -ve. 1 = Inverted. TXP is -ve, and TXN is +ve.

[Register Address Map](#)

Table 2-61: RXPOLARITY

Bits	Default Value	Description
0	0	The RXPOLARITY port can invert the polarity of incoming data: 0 = Not inverted. RXP +ve and RXN -ve. 1 = Inverted. RXP is -ve and RXN is +ve.

[Register Address Map](#)

Table 2-62: RXLPMEN

Bits	Default Value	Description
0	1	Select the RX equalizer setting. There are two types of adaptive filtering depending on system level trade-offs between power and performance. Optimized for power with lower channel loss, the receiver has a power-efficient adaptive mode named the low-power mode (LPM). For equalizing lossier channels, the DFE mode is available. 0:DFE 1:LPM

Table 2-63: RXDFELPMRESET

Bits	Default Value	Description
0	0	Reset for LPM and DFE datapath. Must be toggled after switching between modes to initialize adaptation.

[Register Address Map](#)

Table 2-64: RX Invalid SYNC Header Max

Bits	Default Value	Description
3:0	8	Set max number of invalid sync headers before dropping block sync. Valid values are 1-8.

[Register Address Map](#)

---

## Line Rate Switching

The recommended sequence for line rate switching is as follows:<sup>(1)</sup>

- Ensure all valid data has been sent/received
- Power down the PLL (optional)
- Modify the PLL dividers through the appropriate DRP interface
- Select the correct `refclk` source for each transceiver in RX and TX
- Note if only using one direction the other can be powered down
- Adjust any other control signals
- Power up the PLLs (can be optional)
- Reset the PLL

## DRP Mailboxes

The DRP mailbox interface gives complete access to the common and transceiver DRP address maps as given in the following:

- *Ultrascale GTH UltraScale Architecture GTH Transceivers User Guide* (UG576) [Ref 10]
- *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) [Ref 11]
- *UltraScale Architecture GTY Transceivers User Guide* (UG578) [Ref 18]

Xilinx recommends to reference the appropriate user guide, along with the data sheet for minimum/maximum `refclk` frequencies, line rates, etc. for the correct speed/package combination as well as consideration of system supply voltage.

The following sections highlight the registers of interest in the DRP register space when line rate switching. In general, the DRP registers are tightly packed and read modify write sequences should be used to modify the required bits.

---

1. The JESD204 PHY core must have been generated with the “Dynamic Line Rate” option selected in the “Line rate Switching” section of the configuration GUI.

## UltraScale+ Devices DRP Registers

Ultrascale+ devices can contain GTH or GTY transceivers. Where appropriate, these differences are highlighted in the following tables.

Three PLLs are available in UltraScale+ devices:

- QPLL0
- QPLL1
- CPLL

### Outclk Dividers

The divider encodings and DRP address shown in [Table 2-70](#) are applicable to all three PLLs available in Ultrascale+ devices.

Table 2-65: DRP Address Map

DRP Addr (Hex)	DRP Bits	R/W	Attribute Name	Bits	Encoding	DRP Encoding	Notes
007C	10:8	R/W	TXOUT_DIV	2:0	1	0	
					2	1	
					4	2	
					8	3	
					16	4	
					32	5	
0063	2:0	R/W	RXOUT_DIV	2:0	1	0	
					2	1	
					4	2	
					8	3	
					16	4	
					32	5	

## QPLL0/1

The QPLL VCOs have different operating bands, see the device specific data sheet for more information.

The frequency out of the PLL is given by:

$$F_{(pllClkOut)} = F_{(pllClkIn)} \times \frac{N.FractionalPart}{M \times QPLL\_CLKOUTRATE}$$

Where N = QPLL(0/1)\_FBDIV and M = QPLL(0/1)\_REFCLK\_DIV.

**Note:** For GTH, FractionalPart=1 and QPLL\_CLKOUTRATE=2

To calculate the line rate use:

$$F_{(LineRate)} = \frac{F_{(pllClkOut)} \times 2}{D}$$

Where D = (R/T)XOUT\_DIV

To determine the Fractional part use:

$$FractionalPart = \frac{SDMDATA}{2^{SDMWIDTH}}$$

**Note:** The JESD204 PHY does not currently support fractional parts other than 1.

Table 2-66: Valid Divider Settings

Factor	Attribute	Valid Settings
M	QPLL0_REFCLK_DIV QPLL1_REFCLK_DIV	1, 2, 3, 4
QPLL_CLKOUTRATE (GTU Only)	QPLL0CLKOUT_RATE QPLL1CLKOUT_RATE	1, 2
N	QPLL0_FBDIV QPLL1_FBDIV	<b>GTH:</b> 16, 20, 32, 40, 60, 64, 66, 75, 80, 84, 90, 96, 100, 112, 120, 125, 150 <b>GTU:</b> 16-160
D	RXOUT_DIV TXOUT_DIV	1, 2, 4, 8, 16 32 (GTU Only)

Table 2-72 shows addresses of interest as well as the bits and encoding that must be used to select the correct divider values when interpreting the register content.

Table 2-67: DRP Address Map

DRP Addr (Hex)	DRP Bits	R/W	Attribute Name	Bits	Encoding	DRP Encoding	Notes
0008	15:0	R/W	QPLL0_CFG0	15:0	0 to 65535	0 to 65535	
0009	15:0	R/W	COMMON_CFG0	15:0	0 to 65535	0 to 65535	
000Eh	[0]	R/W	QPLL0CLKOUT_RATE	[0]	HALF	0	
					FULL	1	
0010	15:0	R/W	QPLL0_CFG1	15:0	0 to 65535	0 to 65535	
0011	15:0	R/W	QPLL0_CFG2	15:0	0 to 65535	0 to 65535	
0014	7:0	R/W	QPLL0_FBDIV	7:0	16	14	
					20	18	
					32	30	
					40	38	
					64	62	
					66	64	
					80	78	
0018	11:7	R/W	QPLL0_REFCLK_DIV	7:0	1	16	
					2	0	
					3	1	
					4	2	
0088	15:0	R/W	QPLL1_CFG0	15:0	0 to 65535	0 to 65535	
0089	15:0	R/W	COMMON_CFG1	15:0	0 to 65535	0 to 65535	
0090	15:0	R/W	QPLL1_CFG1	15:0	0 to 65535	0 to 65535	
0091	15:0	R/W	QPLL1_CFG2	15:0	0 to 65535	0 to 65535	
0094	7:0	R/W	QPLL1_FBDIV	7:0	16	14	
					20	18	
					32	30	
					40	38	
					64	62	
					66	64	
					80	78	
					100	98	



Table 2-67: DRP Address Map (Cont'd)

DRP Addr (Hex)	DRP Bits	R/W	Attribute Name	Bits	Encoding	DRP Encoding	Notes
0098	11:7	R/W	QPLL1_REFCLK_DIV	7:0	1	16	
					2	0	
					3	1	
					4	2	

## CPLL

The CPLL operating limits vary for each transceiver type. See the device specific data sheet for more information.

The frequency out of the PLL is given by:

$$F_{(pllClkOut)} = F_{(pllClkIn)} \times \frac{N1 \times N2}{M}$$

Where N = QPLL(0/1)\_FBDIV and M = QPLL(0/1)\_REFCLK\_DIV.

To calculate the line rate use:

$$F_{(LineRate)} = \frac{F_{(pllClkOut)} \times 2}{D}$$

Where D = (R/T)XOUT\_DIV.

Table 2-68: Valid Divider Settings

Factor	Attribute	Valid Settings
M	CPLL_REFCLK_DIV	1, 2
N2	CPLL_FBDIV	1, 2, 3, 4, 5
N1	CPLL_FBDIV_45	4, 5
D	RXOUT_DIV TXOUT_DIV	1, 2, 4, 8

Table 2-74 shows the addresses of interest as well as the bits and encoding that must be used to select the correct divider values when interpreting the register content.

Table 2-69: DRP Address Map

DRP Addr (Hex)	DRP Bits	R/W	Attribute Name	Bits	Encoding	DRP Encoding	Notes
0028	15:8	R/W	CPLL_FBDIV	7:0	1	16	N2
					2	0	
					3	1	
					4	2	
					5	3	
0028	7	R/W	CPLL_FB_DIV_45	0	4	0	N1
					5	1	
002A	15:11	R/W	CPLL_REFCLK_DIV	4:0	1	16	M
					2	0	
002B	15:0	R/W	CPLL_INIT_CFG0	15:0	0 to 65535	0 to 65535	

## UltraScale Devices DRP Registers

Three PLLs are available in UltraScale devices:

- QPLL0
- QPLL1
- CPLL

### Outclk Dividers

The divider encodings and DRP address shown in [Table 2-70](#) are applicable to all three PLLs available in Ultrascale devices.

Table 2-70: DRP Address Map

DRP Addr (Hex)	DRP Bits	R/W	Attribute Name	Bits	Encoding	DRP Encoding	Notes
007C	10:8	R/W	TXOUT_DIV	2:0	1	0	
					2	1	
					4	2	
					8	3	
					16	4	
0063	2:0	R/W	RXOUT_DIV	2:0	1	0	
					2	1	
					4	2	
					8	3	
					16	4	

### QPLL0/1

The QPLL VCOs have different operating bands, see the device specific data sheet for more information.

The frequency out of the PLL is given by:

$$F_{(pllClkOut)} = F_{(pllClkIn)} \times \frac{N.FractionalPart}{M \times QPLL\_CLKOUTRATE}$$

Where N = QPLL(0/1)\_FBDIV and M = QPLL(0/1)\_REFCLK\_DIV.

**Note:** For GTH, FractionalPart=1 and QPLL\_CLKOUTRATE=2

To calculate the line rate use:

$$F_{(LineRate)} = \frac{F_{(pllClkOut)} \times 2}{D}$$

Where D = (R/T)XOUT\_DIV.

Table 2-71: Valid Divider Settings

Factor	Attribute	Valid Settings
M	QPLL0_REFCLK_DIV QPLL1_REFCLK_DIV	1, 2, 3, 4
N	QPLL0_FBDIV QPLL1_FBDIV	16, 20, 32, 40, 64, 66, 80, 100
D	RXOUT_DIV TXOUT_DIV	1, 2, 4, 8, 16

Table 2-72 shows addresses of interest as well as the bits and encoding that must be used to select the correct divider values when interpreting the register content.

Table 2-72: DRP Address Map

DRP Addr (Hex)	DRP Bits	R/W	Attribute Name	Bits	Encoding	DRP Encoding	Notes
0008	15:0	R/W	QPLL0_CFG0	15:0	0 to 65535	0 to 65535	
0009	15:0	R/W	COMMON_CFG0	15:0	0 to 65535	0 to 65535	
0010	15:0	R/W	QPLL0_CFG1	15:0	0 to 65535	0 to 65535	
0011	15:0	R/W	QPLL0_CFG2	15:0	0 to 65535	0 to 65535	
0014	7:0	R/W	QPLL0_FBDIV	7:0	16	14	
					20	18	
					32	30	
					40	38	
					64	62	
					66	64	
					80	78	
0018	11:7	R/W	QPLL0_REFCLK_DIV	7:0	1	16	
					2	0	
					3	1	
					4	2	
0088	15:0	R/W	QPLL1_CFG0	15:0	0 to 65535	0 to 65535	
0089	15:0	R/W	COMMON_CFG1	15:0	0 to 65535	0 to 65535	
0090	15:0	R/W	QPLL1_CFG1	15:0	0 to 65535	0 to 65535	

Table 2-72: DRP Address Map (Cont'd)

DRP Addr (Hex)	DRP Bits	R/W	Attribute Name	Bits	Encoding	DRP Encoding	Notes
0091	15:0	R/W	QPLL1_CFG2	15:0	0 to 65535	0 to 65535	
0094	7:0	R/W	QPLL1_FBDIV	7:0	16	14	
					20	18	
					32	30	
					40	38	
					64	62	
					66	64	
					80	78	
0098	11:7	R/W	QPLL1_REFCLK_DIV	7:0	1	16	
					2	0	
					3	1	
					4	2	

## CPLL

The CPLL operating limits vary for each transceiver type. See the device specific data sheet for more information.

The frequency out of the PLL is given by:

$$F_{(pllClkOut)} = F_{(pllClkIn)} \times \frac{N1 \times N2}{M}$$

Where N = QPLL(0/1)\_FBDIV and M = QPLL(0/1)\_REFCLK\_DIV.

To calculate the line rate use:

$$F_{(LineRate)} = \frac{F_{(pllClkOut)} \times 2}{D}$$

Where D = (R/T)XOUT\_DIV.

Table 2-73: Valid Divider Settings

Factor	Attribute	Valid Settings
M	CPLL_REFCLK_DIV	1, 2
N2	CPLL_FBDIV	1, 2, 3, 4, 5

Table 2-73: Valid Divider Settings

Factor	Attribute	Valid Settings
N1	CPLL_FBDIV_45	4, 5
D	RXOUT_DIV TXOUT_DIV	1, 2, 4, 8

Table 2-74 shows the addresses of interest as well as the bits and encoding that must be used to select the correct divider values when interpreting the register content.

Table 2-74: DRP Address Map

DRP Addr (Hex)	DRP Bits	R/W	Attribute Name	Bits	Encoding	DRP Encoding	Notes
0028	15:8	R/W	CPLL_FBDIV	7:0	1	16	N2
					2	0	
					3	1	
					4	2	
					5	3	
0028	7	R/W	CPLL_FB_DIV_45	0	4	0	N1
					5	1	
002A	15:11	R/W	CPLL_REFCLK_DIV	4:0	1	16	M
					2	0	
002B	15:0	R/W	CPLL_INIT_CFG0	15:0	0 to 65535	0 to 65535	

## 7 Series FPGAs DRP Registers

Two PLLs are available in 7 series FPGAs:

- QPLL
- CPLL

### Outclk Dividers

These are applicable to both PLL types.

Table 2-75: DRP Address Map

DRP Addr (Hex)	DRP Bits	R/W	Attribute Name	Bits	Encoding	DRP Encoding	Notes
0088	6:4	R/W	TXOUT_DIV	2:0	1	0	
					2	1	
					4	2	
					8	3	
					16	4	
0088	2:0	R/W	RXOUT_DIV	2:0	1	0	
					2	1	
					4	2	
					8	3	
					16	4	

### QPLL

The QPLL VCO operates within two different operating bands, see the device specific data sheet for more information. Note that if switching between these bands, Bit[6] in the QPLL\_CFG register must be modified to select the correct band.

The frequency out of the PLL is given by:

$$F_{(pllClkOut)} = F_{(pllClkIn)} \times \frac{N.FractionalPart}{M \times QPLL\_CLKOUTRATE}$$

Where N = QPLL(0/1)\_FBDIV and M = QPLL(0/1)\_REFCLK\_DIV.

**Note:** For GTH, FractionalPart=1 and QPLL\_CLKOUTRATE=2

To calculate the line rate use:

$$F_{(LineRate)} = \frac{F_{(pllClkOut)} \times 2}{D}$$

Where  $D = (R/T)XOUT\_DIV$ .

Table 2-76: Valid Divider Settings

Factor	Attribute	Valid Settings
M	QPLL_REFCLK_DIV	1, 2, 3, 4
N	QPLL_FBDIV QPLL_FBDIV_RATIO	16, 20, 32, 40, 64, 66*, 80, 100
D	RXOUT_DIV TXOUT_DIV	1, 2, 4, 8, 16

Table 2-77 shows the addresses of interest as well as the bits and encoding that must be used to select the correct divider values when interpreting the register content.

Table 2-77: DRP Address Map

DRP Addr (Hex)	DRP Bits	R/W	Attribute Name	Bits	Encoding	DRP Encoding	Notes
0032	15:0	R/W	QPLL_CFG	15:0	0 to 65535	0 to 65535	Reserved. This attribute is the configuration setting for the QPLL. QPLL_CFG[6] selects the QPLL frequency band. 0 = Upper band 1 = Lower band The recommended value from the 7 series FPGAs Transceivers Wizard should be used.
0033	15:11	R/W	QPLL_REFCLK_DIV	4:0	1	16	
					2	0	
					3	1	
					4	2	
0033	10:0	R/W	QPLL_CFG	26:16	0 to 2047	0 to 2047	
0036	9:0	R/W	QPLL_FBDIV	9:0	0 to 1023	0 to 1023	Supported divider values (16,20,32,40,64,66,80,100)
0037	6	R/W	QPLL_FBDIV_RATIO	0	0 to 1	0 to 1	*Set to 1 for all N values apart from N = 66, then set to 0

## CPLL

The CPLL operating limits vary for each transceiver type. see the device specific data sheet for more information.



The frequency out of the PLL is given by:

$$F_{(pllClkOut)} = F_{(pllClkIn)} \times \frac{N1 \times N2}{M}$$

Where N = QPLL(0/1)\_FBDIV and M = QPLL(0/1)\_REFCLK\_DIV.

To calculate the line rate use:

$$F_{(LineRate)} = \frac{F_{(pllClkOut)} \times 2}{D}$$

Where D = (R/T)XOUT\_DIV.

Table 2-78: Valid Divider Settings

Factor	Attribute	Valid Settings
M	CPLL_REFCLK_DIV	1, 2
N2	CPLL_FBDIV	1, 2, 3, 4, 5
N1	CPLL_FBDIV_45	4, 5
D	RXOUT_DIV TXOUT_DIV	1, 2, 4, 8

Table 2-79 shows the addresses of interest as well as the bits and encoding that must be used to select the correct divider values when interpreting the register content.

Table 2-79: DRP Address Map

DRP Addr (Hex)	DRP Bits	R/W	Attribute Name	Bits	Encoding	DRP Encoding	Notes
005C	15:8	R/W	CPLL_CFG	7:0	0 to 255	0 to 255	
005D	15:0	R/W	CPLL_CFG	23:8	0 to 65535	0 to 65535	
005E	12:8	R/W	CPLL_REFCLK_DIV	4:0	1	16	M
					2	0	
005E	7	R/W	CPLL_FB_DIV_45	0	4	0	N1
					5	1	
005E	6:0	R/W	CPLL_FBDIV	6:0	1	16	N2
					2	0	
					3	1	
					4	2	
					5	3	

# Designing with the Core

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

- When a JESD204 core is generated with shared logic in the core selected, the JESD204 PHY core is instantiated internally within the JESD204 core during generation. All ports and parameters are controlled by the JESD204 IP. Using the IP in this mode requires no user intervention and is not described in this document; see the *JESD204 LogiCORE IP Product Guide* (PG066) [Ref 2] for details about this mode.
- When a JESD204 core is generated with shared logic in the example design selected, the JESD204 PHY core is instantiated as a stand-alone IP core in the example design provided with the JESD204 IP. In this case, the JESD204 PHY IP top level is available directly for instantiation in designs, and the JESD204 PHY IP GUI is available. This chapter describes using the JESD204 PHY in this mode.

---

## JESD204 PHY Configuration Options

The JESD204 PHY can be generated in eight main logical configurations.

**Notes:**

1. The 32-bit JESD204 PHY can be connected only with the JESD204B IP. The 64-bit JESD204 PHY cannot be connected to the JESD204B IP.
2. The AXI Enabled parameter is not available on GTP transceiver devices.

*Table 3-1: JESD204 PHY Configuration Options*

AXI Enabled	Shared Logic	Transceiver Debug	Description
0	0	0	No AXI logic present, no COMMON.
0	0	1	No AXI logic present, no COMMON, current JESD204 transceiver ports list.
0	1	0	No AXI logic present, COMMON in core.
0	1	1	No AXI logic present, COMMON in core, current JESD204 transceiver ports list.
1	0	0	AXI logic present, no COMMON or COMMON AXI control registers.

Table 3-1: JESD204 PHY Configuration Options (Cont'd)

AXI Enabled	Shared Logic	Transceiver Debug	Description
1	0	1	AXI logic present, no COMMON or COMMON AXI control registers, transceiver debug ports are minus ports mapped to AXI control.
1	1	0	AXI logic present, COMMON in core.
1	1	1	AXI logic present, COMMON in core, transceiver debug ports are minus ports mapped to AXI control.

The common PLL DRP interface is not presented at the JESD204 PHY core output ports under any circumstance and can only be accessed with the AXI interface enabled and a QPLL selected as a `refclk` source.

In UltraScale devices, when the AXI interface is enabled and QPLL0/1 is selected as one of the PLLs, both PLL `refclk` ports appear. This is different to non-AXI mode where only the `refclk` of the selected PLL appears. This is to maximize flexibility when using the AXI interface for line rate switching.



**IMPORTANT:** When used as a sub-core of the JESD204 core, AXI is disabled by default and cannot be enabled. You must use the JESD204 core with **Shared Logic in Example Design** and hence a separate JESD204 PHY to get access to the AXI JESD204 PHY interface.

## General Design Guidelines

This section includes tips about getting started with the JESD204 PHY core.

### Use the Example Design as a Starting Point

Each instance of the JESD204 PHY core created by the Vivado® Design Suite is delivered with an example design that can be implemented in an FPGA and simulated. This design can be used as a starting point for your own design or can be used to troubleshoot your application, if necessary.

See [Chapter 5, Example Design](#) for information about using and customizing the example designs for the JESD204 PHY core. For more information on the Vivado IP integrator, see the *JESD204 LogicCORE IP Product Guide* (PG066) [\[Ref 2\]](#).

## Degree of Difficulty

JESD204 designs are challenging to implement in any technology, and the degree of difficulty is further influenced by:

- Maximum system clock frequency
- Targeted device architecture
- Nature of your application

All JESD204 implementations require careful attention to system performance requirements. Pipelining, logic mapping, placement constraints, and logic duplication are all methods that help boost system performance.

---

## Clocking

This section describes the options available for clocking the JESD204 PHY core and the transceiver(s). The following clocks are used in the JESD204 PHY core.

- **DRP Clock** – The transceiver requires an auxiliary clock for internal use and also for the reset state machines within the JESD204 PHY core. See the appropriate device family data sheet for the min and max DRP clock frequencies permitted.
- **Reference Clock** – The GTP/GTX/GTH serial transceivers require a stable, low-jitter reference clock that has a device and speed grade dependent range. In some circumstances, the same source clock can supply both the reference clock and core clock. Two reference clocks are required if both CPLL and QPLL (GTX/GTH devices) or PLL0 and PLL1 (GTP devices) are selected in the GUI.
- **AXI4-Lite Configuration Interface Clock** – Required if the AXI is enabled. This is asynchronous to any other clock and can be driven by the processor subsystem.

## 32-bit configurations

### Core Clock

The JESD204 PHY core operates using a 32-bit (4-byte) datapath. The device clock for the core logic therefore runs at one quarter of the byte clock rate (1/ 40th of the serial line rate). For the JESD204 and JESD204 PHY cores, this is referred to as the core clock.

## 64-bit configurations

### *Core Clock*

The JESD204 PHY core operates using a 64-bit (8-byte) datapath plus a 2 bit header. The device clock for the core logic therefore runs at 1/ 66th of the serial line rate. For the JESD204 and JESD204 PHY cores, this is referred to as the core clock. This clock must be supplied externally to the core, and must be derived from the same clock source as the REFCLK supplied to the transceiver.

**Note:** 64-bit configuration only supports 64B66B encoding. Do not use this configuration when connected to a JESD204B core.

## Resets

There are two system resets (tx\_sys\_reset and rx\_sys\_reset) and two data path resets (tx\_reset\_gt and rx\_reset\_gt).

These enable the JESD204 PHY core to be used by a transmit JESD204 link and a receive JESD204 link independently.

### Transmit Reset:

- The transmit reset input (tx\_reset\_gt) initiates a data path only reset sequence for the transmit logic data path, and tx\_reset\_done is asserted when the reset sequence is complete.
- The (tx\_sys\_reset) input initiates a complete data path and PLL reset sequence and asserts tx\_reset\_done when complete.

### Receive Reset:

- The receive reset input (rx\_reset\_gt) initiates a data path only reset sequence for the receive logic data path, and rx\_reset\_done is asserted when the reset sequence is complete.
- The (rx\_sys\_reset) input initiates a complete data path and PLL reset sequence and asserts rx\_reset\_done when complete.

Figure 3-1 shows 7 Series GT Reset Control.

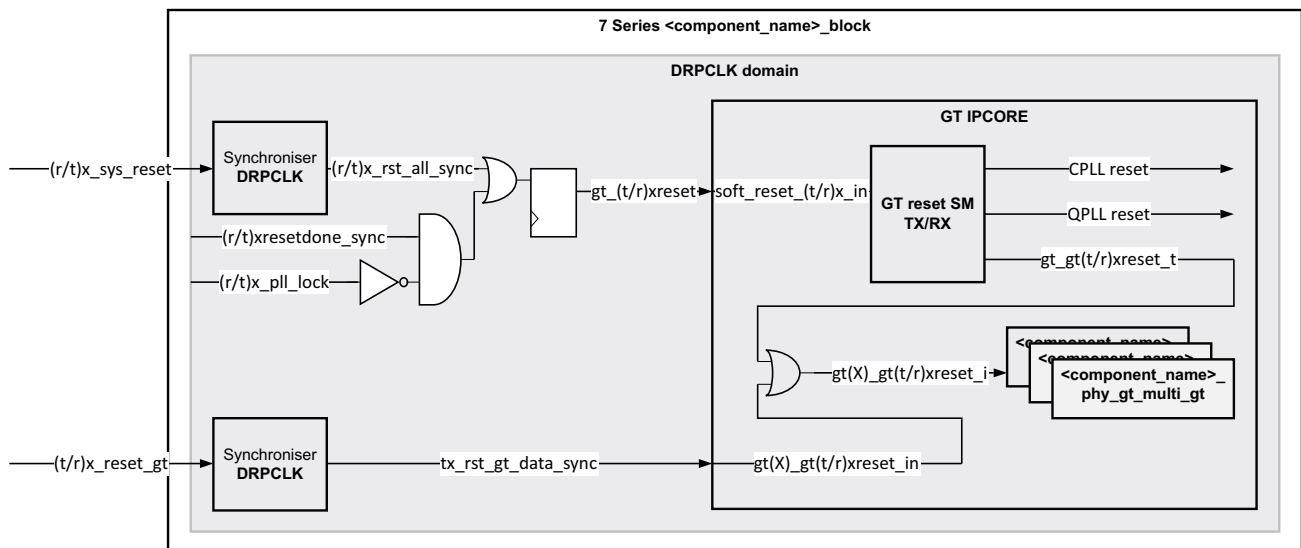


Figure 3-1: 7 Series GT Reset Control

Figure 3-2 shows UltraScale GT Reset Control.

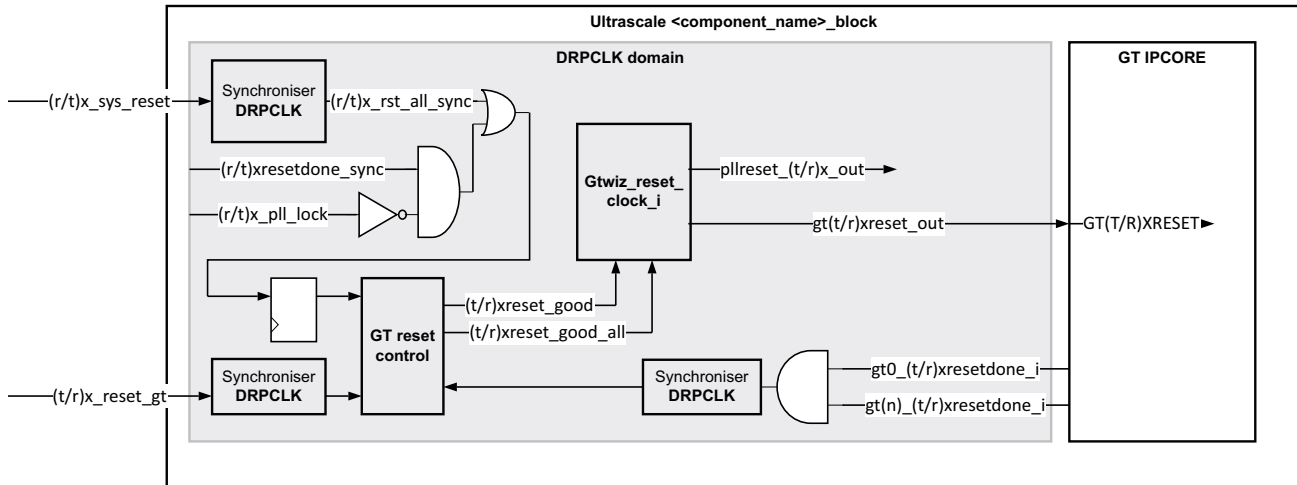


Figure 3-2: UltraScale GT Reset Control

## Protocol Description

See the *JESD204 LogiCORE IP Product Guide* (PG066) for a full description of the protocol [Ref 2].

# Design Flow Steps

This chapter describes customizing and generating the core, constraining the core, and the simulation, synthesis and implementation steps that are specific to this IP core. More detailed information about the standard Vivado® design flows can be found in the following Vivado Design Suite user guides:

- *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 3]
- *Vivado Design Suite User Guide: Getting Started* (UG910) [Ref 5]
- *Vivado Design Suite User Guide: Logic Simulation* (UG900) [Ref 6]

---

## Customizing and Generating the Core

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

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

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

For details, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 4] and the *Vivado Design Suite User Guide: Getting Started* (UG910) [Ref 5].

**Note:** Figures in this chapter are illustrations of the Vivado IDE. The layout depicted here might vary from the current version.



## Configuration Tab

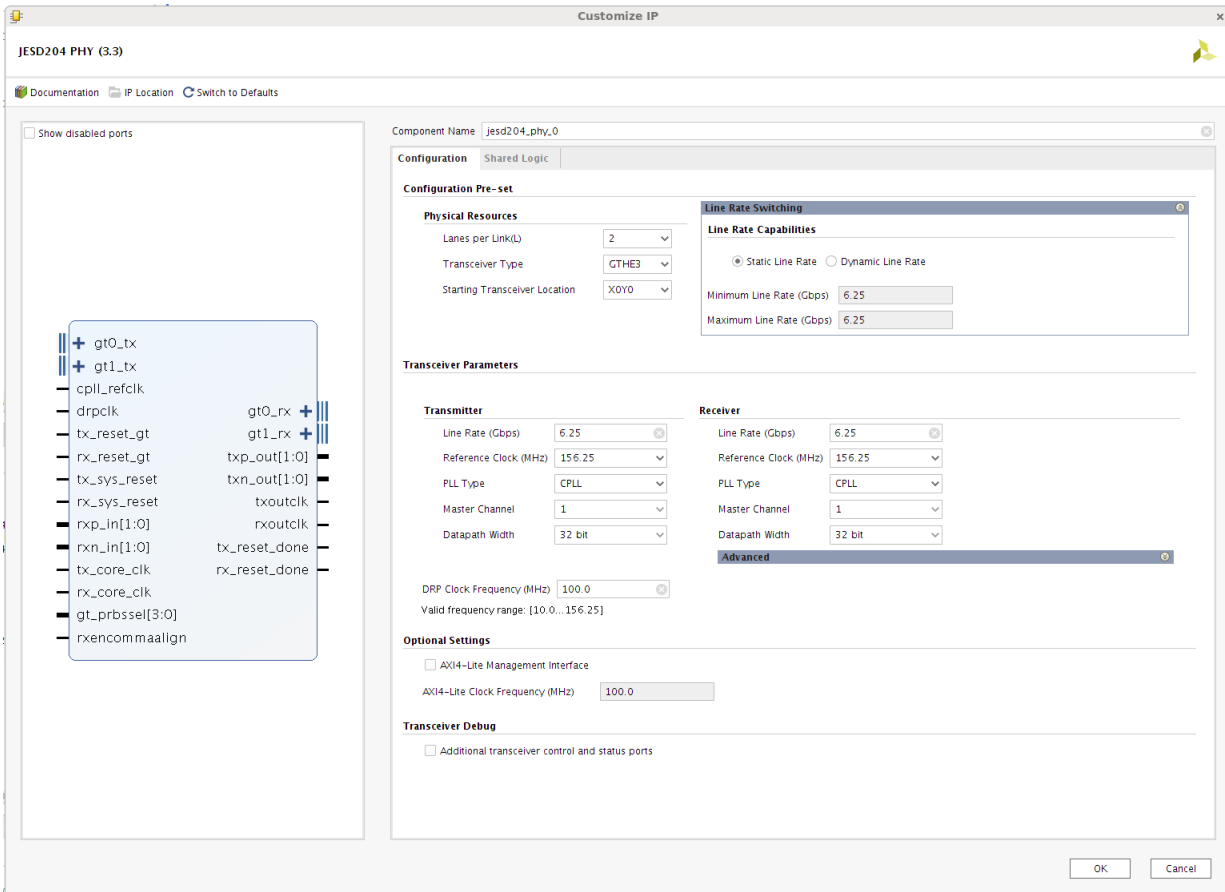


Figure 4-1: Configuration Tab

- **Configuration Preset**

- **Physical Resources**

- **Lanes per Link** - The core supports 1 to 12 lanes. The number of transmit lanes always matches the number of receive lanes. For asymmetric interfaces, multiple cores can be generated and multiple JESD204 PHY cores can be connected to a single JESD204 core.
    - **Transceiver Type** - For Devices with Multiple transceiver types, this option allows the selection of transceiver type for which to generate the core.
    - **Starting Transceiver Location** - Select starting location of lane 0. This allows Vivado to generate the correct location constraints for the transceiver during IP generation. This is only available on Ultrascale and Ultrascale+ devices.

- **Line Rate Switching**

- **Line Rate Capabilities** - Select "Static" or "Dynamic" Line rate. This controls how the core is generated. Selecting "Static" generates a design that is exactly

as specified in the GUI. Selecting "Dynamic" generates a core with the defaults set to exactly as specified in the GUI, but it will also include any additional logic required to switch via software control to any line rate between the following minimum and maximum values:

**Minimum Line Rate** - Set the minimum line rate value required for dynamic line rate switching.

**Maximum Line Rate** - Set the Maximum line rate required for line dynamic line rate switching.

- **Transceiver Parameters, Transmitter, and Receiver**
  - **Line Rate** – The serial line rate in Gb/s can be selected for transmit and receive independently. The minimum rate is 1 Gb/s and the maximum depends on the chosen device and speed grade. If the core is generated with dynamic line rate switching capability, these values are the default configuration at power up.
  - **Reference Clock** – The reference clock must be selected from the drop-down list, which presents a list of valid reference clock frequencies for the selected line rate. Independent reference clocks can only be selected if different PLLs are selected for transmit and receive. If the core is generated with dynamic line rate switching capability these values are the default configuration at power up.
  - **PLL Type** – Select the QPLL or CPLL for transmit and receive. See the appropriate device transceiver user guide for more details and limitations. If the core is generated with dynamic line rate switching capability these values are the default configuration at power up. If dynamic switching between PLL types (QPLL / CPLL) is required, the core must be generated with TX/RX set to use one of each type.
  - **Master channel** – Select the transceiver channel to source TX/RXOUTCLK from. This channel should not be powered down if TX/RXOUTCLK is used, as this will switch off the CLK.
  - **Data Path Width** (UltraScale only) – Select between a 32 or 64-bit datapath.
  - **DRP Clock Frequency** – The frequency of the DRP clock being applied to the core so reset delays can be adjusted by the reset state machines.
- **Advanced (Receiver only)**
  - **Channel Attenuation** – Select RX equalization mode, Auto, Low loss or High Loss. Refer to the appropriate transceiver user guide for more information.

This parameter affects the value set on the RXLP MEN port into the transceiver.

Low loss = LPM

High Loss = DFE

Auto = LPM or DFE selected automatically based on Insertion Loss at Nyquist and RX Line rate.

- **Insertion loss at Nyquist** - Enter the insertion loss at Nyquist of the channel.  
*Note:* It is not recommended to change this value from the default.
- **Optional Settings**
  - **AXI4-Lite Management Interface** – Select to include the AXI4-Lite configuration interface. This allows AXI-based access to the Transceiver and Common DRPs.  
*Note:* This is not available on GTP transceiver devices.
  - **AXI4-Lite Clock Frequency** – The AXI4-Lite clock can be connected to the main processor clock. When applicable, the clock domain boundary crossings are handled inside the IP to simplify implementation. Also, when a clock boundary is involved the AXI access is stretched, resulting in an extended access time.
  - **Extend reset to 3 ms (7-Series only)** – Increases reset duration to 3ms in the transceiver logic. When enabled, simulation times are increased.
- **Transceiver Debug** – Select to include additional transceiver control and status ports for debugging purposes. See [Transceiver Debug Interface in Chapter 2](#) for more information.

## Shared Logic Tab

The JESD204 PHY can be generated with Shared Logic (Quad PLL(s)) included in the core or with Shared Logic in the example design. When using a single JESD204 PHY, Shared Logic should always be included in the core to ensure all the necessary clocking logic is included in your design.

*Note:* This includes Common QPLL blocks and any required clocking logic based on the configuration.

## User Parameters

[Table 4-1](#) shows the relationship between the GUI fields in the Vivado IDE and the User Parameters (which can be viewed in the Tcl console).

*Table 4-1: Vivado IDE Parameter to User Parameter Relationship<sup>(1)</sup>*

Vivado IDE Parameter/Value	User Parameter/Value	Default Value
Lanes per Link	C_LANES	2
Line Rate		
Transmit	GT_Line_Rate	6.25
Receive	RX_GT_Line_Rate	6.25
Reference Clock		
Transmit	GT_REFCLK_FREQ	156.25
Receive	RX_GT_REFCLK_FREQ	156.25
PLL Type		

Table 4-1: Vivado IDE Parameter to User Parameter Relationship<sup>(1)</sup> (Cont'd)

Vivado IDE Parameter/Value	User Parameter/Value	Default Value
Transmit	C_PLL_SELECTION	0 (= CPLL)
Receive	RX_PLL_SELECTION	0 (= CPLL)
DRP Clock Frequency	DRPCLK_FREQ	10.0
Shared Logic	SupportLevel	1 (= Include Shared Logic in Core)
Transceiver Debug	TransceiverControl	FALSE
AXI-Lite Interface	AXI_Lite	FALSE
Transceiver Type	Transceiver	GTHE3
Starting Transceiver Location	GT_Location	X0Y0
Extend Reset to 3 ms	gt_val_extended_timeout	False
Channel Attenuation	Equalization_Mode	Auto
Data Path Width (TX)	Tx_use_64b	0
Data Path Width (RX)	Rx_use_64b	0
Master Channel (TX)	Tx_MasterChan	1
Master Channel (RX)	Rx_MasterChan	1
Insertion Loss at Nyquist	Ins_Loss	12 (UltraScale only)
Configuration Type	Config_Type	0 (Not Artix)
Min Line Rate	Min_Line_Rate	6.25 (Not Artix)
Max Line Rate	Max_Line_Rate	6.25 (Not Artix)

**Notes:**

- Parameters and default values will differ based on the selected device.

## Output Generation

For details, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 4].

---

## Constraining the Core

This section describes how to constrain a design containing the JESD204 core. This is accomplished by using the XDC delivered with the core at generation time. An additional XDC file is generated with the IP example design; only the core XDC file should be used in user designs.

### Required Constraints

This section defines the constraint requirements for the core. Constraints are provided in several XDC files which are delivered with the core and the example design to give a starting point for constraints for the user design.

There are four XDC constraint files associated with this core:

- `<corename>_example_design.xdc`
- `<corename>_ooc.xdc`
- `<corename>.xdc`
- `<corename>_clocks.xdc`

The first is used only by the example design; the second file is used for Out Of Context support where this core can be synthesized without any wrappers; the third file is the main XDC file for this core.

### Clock Frequencies

The reference clock and core clock frequency constraints vary depending on the selected line rate and reference clock when generating the core. See the generated XDC for details.

### Clock Domains

There are also several paths where clock domains are crossed. These include the management interface. See the generated XDC file for details.

### Clock Management

Reference clock and core clock resources require location constraints appropriate to your top level design.

## Clock Placement

Reference clock input should be given location constraints appropriate to your top level design and to the placement of the transceivers.

**Note:** Transceiver location constraints are only required for 7-Series devices.

Core clock input (if required) should be given location constraints appropriate to your top level design.

## Banking

This section is not applicable for this IP core.

## Transceiver Placement

Transceivers should be given location constraints appropriate to your design. In some cases, example transceiver location constraints can be found in the example design XDC file. For 7 series devices, the GT location constraints are in the transceiver's XDC file.

For UltraScale devices, it is recommended that the location of the transceivers is configured during IP customization. This allows Vivado to generate the correct location constraints.

## I/O Standard and Placement

All ports should be given I/O standard and location constraints appropriate to your top level design.

---

## Simulation

For comprehensive information about Vivado simulation components, as well as information about using supported third-party tools, see the *Vivado Design Suite User Guide: Logic Simulation* (UG900) [Ref 6].



---

**IMPORTANT:** For cores targeting 7 series or Zynq-7000 AP SoC devices, UNIFAST libraries are not supported. Xilinx IP is tested and qualified with UNISIM libraries only.

---

---

## Synthesis and Implementation

For details about synthesis and implementation, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 4].

## Example Design

This chapter contains information about the example design provided in the Vivado® Design Suite.

Although the JESD204 PHY core is not intended to be used as a standalone solution, an example design is provided for the IP core. The example design is a lightweight harness that can operate in an external TX to RX loopback mode, or in an independent RX/TX channel mode.

See the example design provided with the JESD204 IP for a more detailed use case example [Ref 2]. The example design structure is the same for 32 and 64 bit operating modes (apart from the generators and checkers content).

**Note:** 8b/10b and 64b/66b are incompatible line coding schemes. So even when a matching line rate is chosen, and the TX and RX direction are of different data widths, the `demo_tb` loopback mode will not be used.

To open the example design, right-click the IP and select **Open IP Example Design** as shown in [Figure 5-1](#).

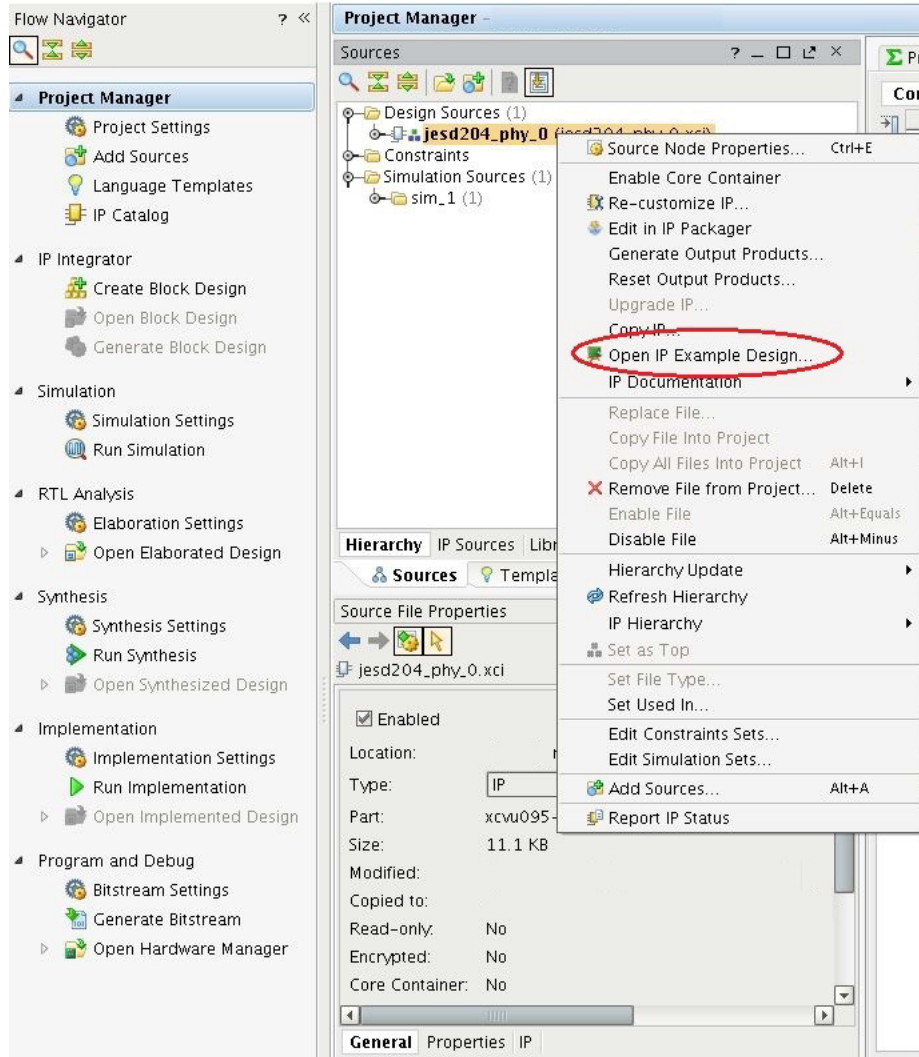


Figure 5-1: Opening the Example Design

## 32-bit Configurations

The 32-bit configuration example design generates data internally for the TX path. This is checked externally and can also be fed back into the RX path, where a simple check function exists to verify the incoming data.

The `clks_in` module places the appropriate clock buffers on the clock paths dependent on the technology chosen.

The sequencer is responsible for indicating when the example design can test the incoming data as well as sequencing the data that is transmitted. When both the RX and TX channels have completed the reset sequence, it sends out K28.5 symbols. These are followed by four `/R/.../A/` frames mimicking what is seen on a JESD204 data interface. Note `/Q/` and the 14 bytes of `/ILA/` data are not sent. They just increment counter values.



Figure 5-2 shows a block diagram for the example design.

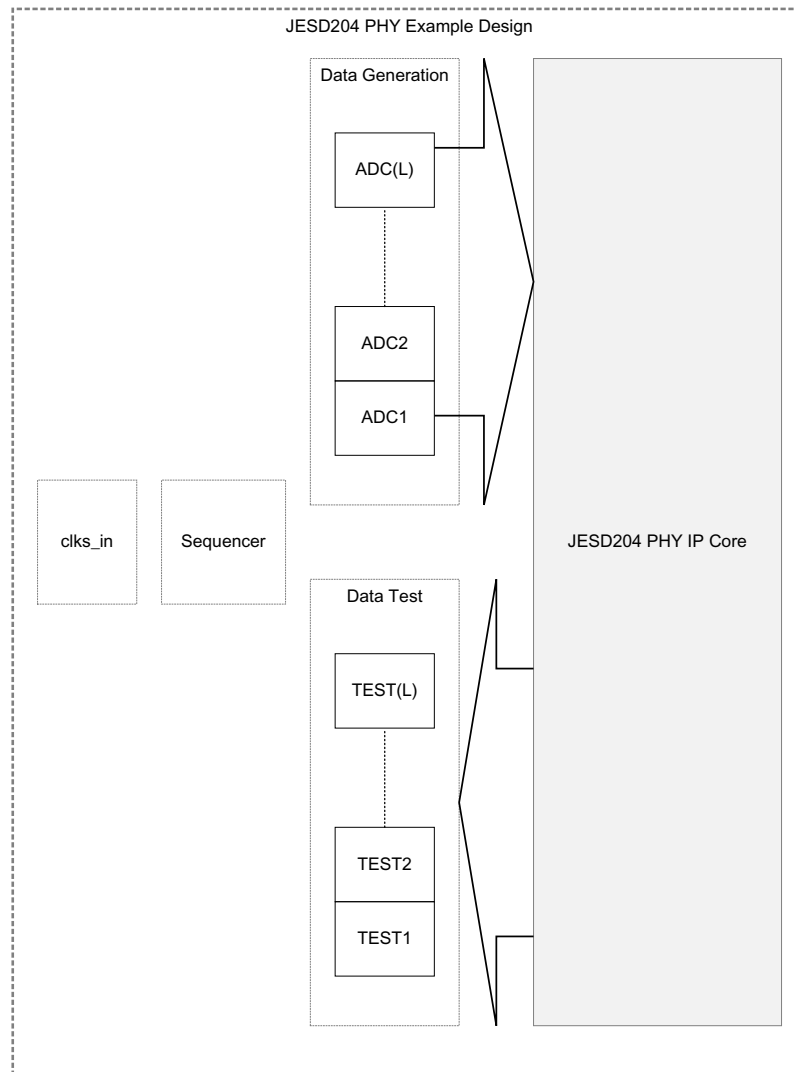


Figure 5-2: Example Design Block Diagram

Figure 5-3 and Figure 5-4 show the clock structure for different and identical PLL types, respectively.



**TIP:** If different PLLs are selected for the RX and TX paths, the port names for *refclk* are named as "rx" and "tx." If they are the same, the port is named as "common."

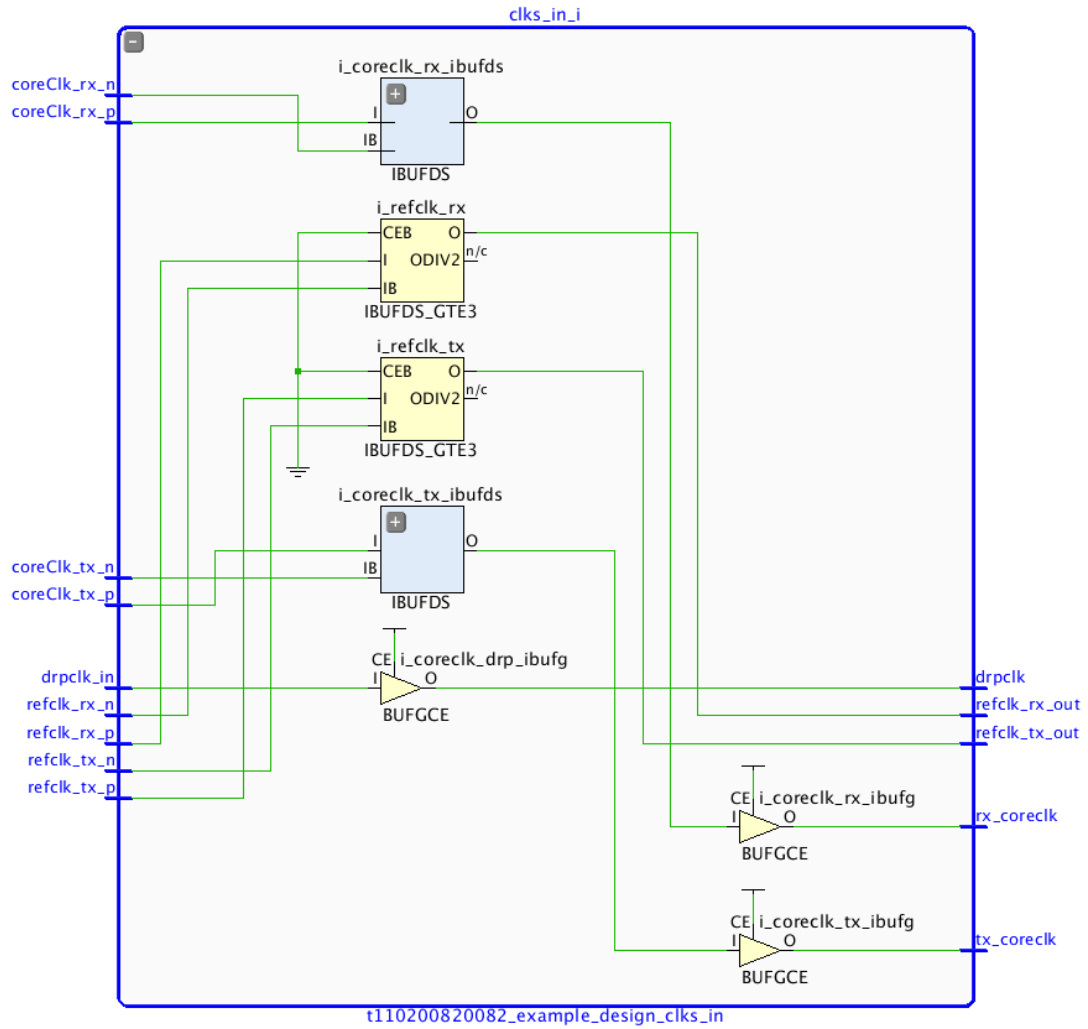


Figure 5-3: Clock Structure with Different PLL Types in 32-bit configuration

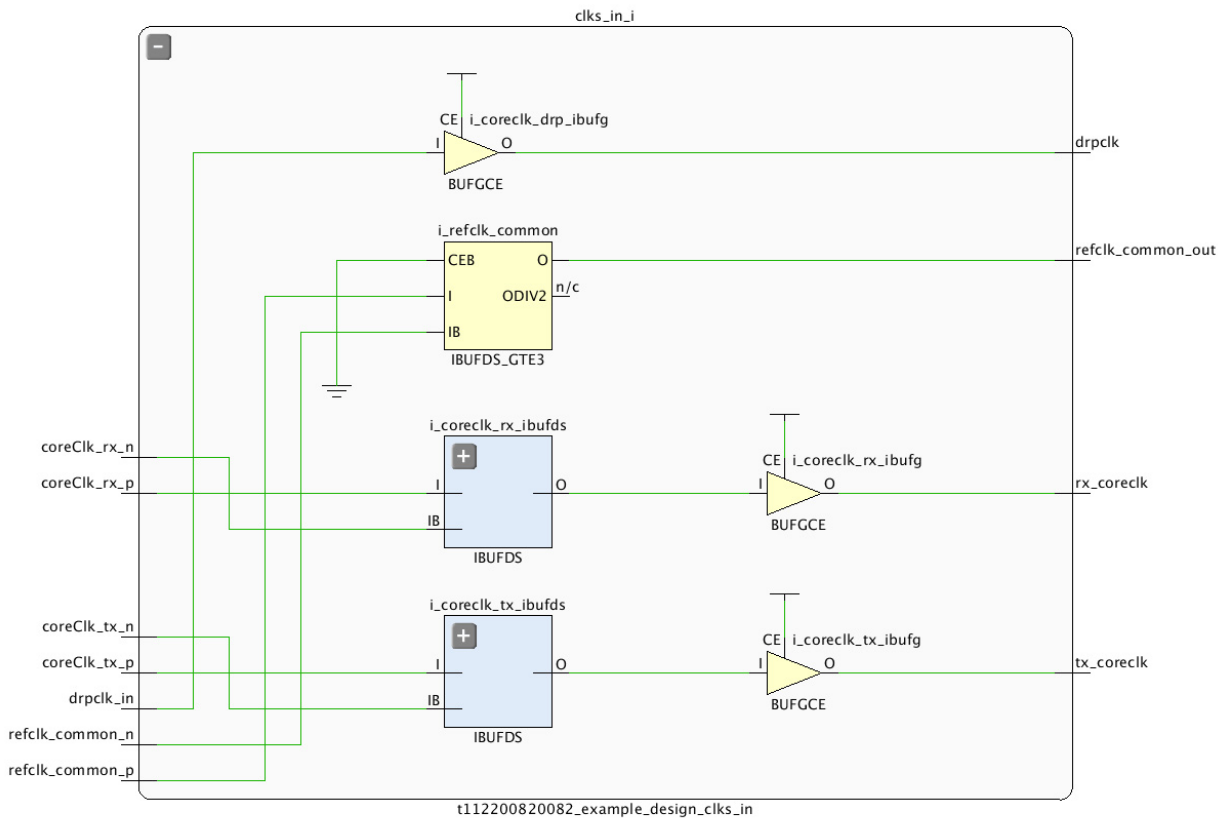


Figure 5-4: Clock Structure with Matching PLL Types in 32-bit configuration

## 64-bit Configurations

The 64-bit configuration example design uses a sequencer module which is responsible for enabling data generation and data checking on the Rx side.

A rolling 8-bit counter is used to generate data, which is replicated for each byte in the interface.

Rollover from 255 alternates from 0 to 1 to allow the block alignment functions to be able to detect the 01 or 10 header bits correctly. Header values of 00 and 11 are considered invalid.

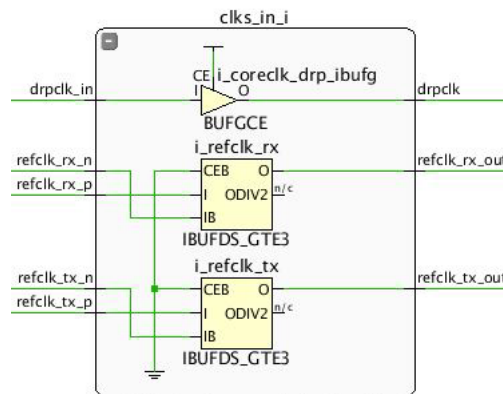


Figure 5-5: Clock Structure with Different PLL Types in 64-bit configuration

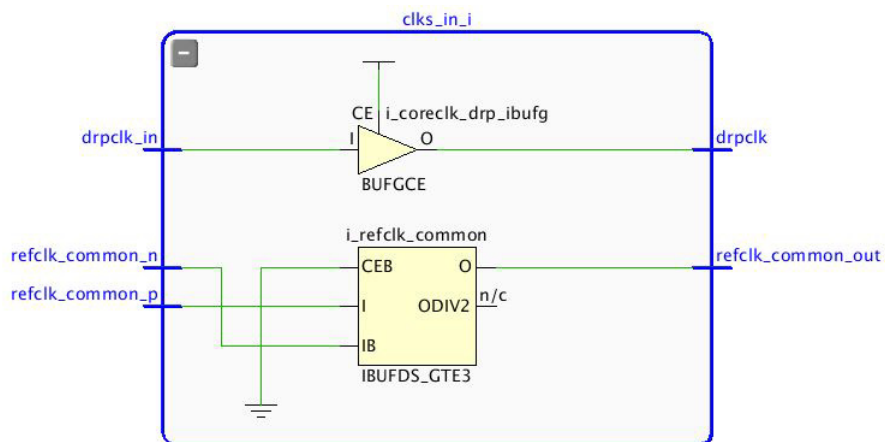
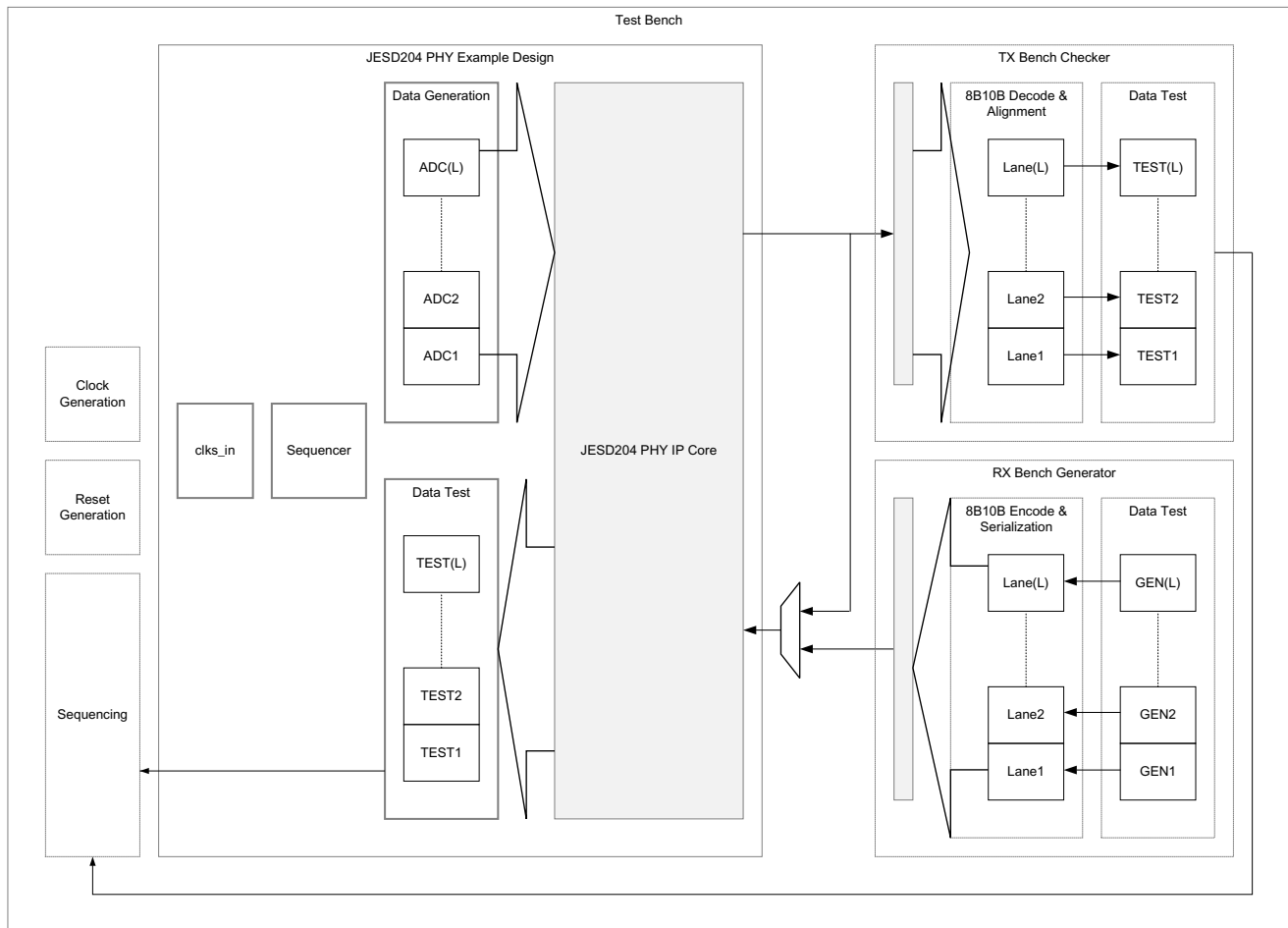


Figure 5-6: Clock Structure with Matching PLL Types in 64-bit configuration

# Test Bench

This chapter contains information about the test bench provided in the Vivado® Design Suite. [Figure 6-1](#) shows the test bench block diagram.



**Figure 6-1: Test Bench Block Diagram**

Hierarchy is used extensively to create per-lane stimulus and checker blocks which allow easier signal viewing in the waveform window.

The test bench provides all clocks required by the design. If the TX and RX line rates are equal, the loopback path is automatically selected for simulation at IP build time.

Several event messaging functions exist at the top-level. These indicate when the lanes are out of reset and the bit rates used by the RX and TX channels. A timeout function is also included.

The data generation and testing functions exist in separate modules instantiated in the top-level test bench. This enables clear navigation to a lane data stream with the waveform viewer. The data stream starts when both the TX and RX paths are out of reset. K28.5 (/K/) symbols are transmitted to allow the transceivers to bit align.

In 32-bit configurations, an ILA-type sequence consisting of just the K28.0 (/R/), K28.3 (/A/) and data, is sent to allow the test bench to align to a 32-bit boundary. The bench continues to run for a specified length of core clock cycles before finishing.

In 64-bit configurations a byte incrementing data pattern is sent. The header value alternates between 1 and 2.

**Note:** The TX serial data and clock recovery Verilog module can be reused as a plugin serial line decoder for the GT output in both 32 and 64 bit modes. The bit period is measured and data sampled and decoded accordingly. This can be a useful method for debugging designs created in IPI.

Similarly, the RX generator can be reused as a GT data injection module for a custom testbench. (If using with the JESD204B core, scrambling must be disabled as it is not supported by the recovery and generator functions.)




---

**IMPORTANT:** *To change any IP parameters, you must reconfigure the IP and regenerate the example design.*

---

## AXI Interface

If the IP is generated with the AXI interface, this is presented in the test bench along with tasks to write/read over the interface. Basic examples are given at the start of the test, however the tasks might be used to test out custom sequences.

**Note:** This interface is not available on GTP transceiver devices.

For UltraScale devices, see *UltraScale Architecture GTH Transceivers User Guide* (UG576) [Ref 10], *Appendix B* for a detailed DRP register map.

For UltraScale+ devices, see *UltraScale Architecture GTY Transceivers User Guide* (UG578) [Ref 18].

For 7 series devices, see *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) [Ref 11], *Appendix D*.

# Verification, Compliance, and Interoperability

The JESD204 core has been verified using both simulation and hardware testing.

---

## Simulation

A highly parameterizable transaction-based simulation test suite has been used to verify the core. Tests include:

- Scrambling and alignment
  - Loss and regain of synchronization
  - Frame transmission
  - Frame reception
  - Recovery from error conditions
- 

## Hardware Testing

The core has been used in many hardware test platforms within Xilinx, interfacing to a selection of DAC and ADC's from several manufacturers. Refer to the [JESD204 lounge](#) for further details and hardware examples (registration required).

# Debugging

This appendix includes details about resources available on the Xilinx Support website and debugging tools.

---

## Finding Help on Xilinx.com

To help in the design and debug process when using the JESD204 PHY, the [Xilinx Support web page](#) contains key resources such as product documentation, release notes, answer records, information about known issues, and links for obtaining further product support.

### Documentation

This product guide is the main document associated with the JESD204 PHY. This guide, along with documentation related to all products that aid in the design process, can be found on the [Xilinx Support web page](#) or by using the Xilinx Documentation Navigator.

Download the Xilinx Documentation Navigator from the [Downloads page](#). For more information about this tool and the features available, open the online help after installation.

### Answer Records

Answer Records include information about commonly encountered problems, helpful information on how to resolve these problems, and any known issues with a Xilinx product. Answer Records are created and maintained daily ensuring that users have access to the most accurate information available.



Answer Records for this core can be located by using the Search Support box on the main [Xilinx support web page](#). To maximize your search results, use proper keywords such as:

- Product name
- Tool message(s)
- Summary of the issue encountered

A filter search is available after results are returned to further target the results.

### **Master Answer Record for the JESD204 PHY Core**

AR: [61911](#)

## **Technical Support**

Xilinx provides technical support at the [Xilinx Support web page](#) for this LogiCORE™ IP product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support if you do any of the following:

- Implement the solution in devices that are not defined in the documentation.
- Customize the solution beyond that allowed in the product documentation.
- Change any section of the design labeled DO NOT MODIFY.

To contact Xilinx Technical Support, navigate to the [Xilinx Support web page](#).

---

## **Debug Tools**

There are many tools available to address IP core design issues. It is important to know which tools are useful for debugging various situations.

### **Vivado Design Suite Debug Feature**

Vivado Design Suite debug feature inserts logic analyzer and virtual I/O cores directly into your design. The debug feature also allows you to set trigger conditions to capture application and integrated block port signals in hardware. Captured signals can then be analyzed. This feature in the Vivado IDE is used for logic debugging and validation of a design running in Xilinx devices.

The Vivado logic analyzer is used with the logic debug LogiCORE IP cores, including:

- ILA 2.0 (and later versions)
- VIO 2.0 (and later versions)

See the *Vivado Design Suite User Guide: Programming and Debugging* (UG908) [\[Ref 8\]](#).

## Reference Boards

Various Xilinx development boards support the JESD204 PHY. These boards can be used to prototype designs and establish that the core can communicate with the system.

- 7 series FPGA evaluation boards:
  - AC701
  - KC705
  - ZC706
  - VC709
- UltraScale
  - KCU105
  - VCU108
- UltraScale+
  - KCU114
  - VCU118
  - ZCU102

# Simulation Debug

The simulation debug flow for QuestaSim is illustrated in [Figure B-1](#). A similar approach can be used with other simulators.

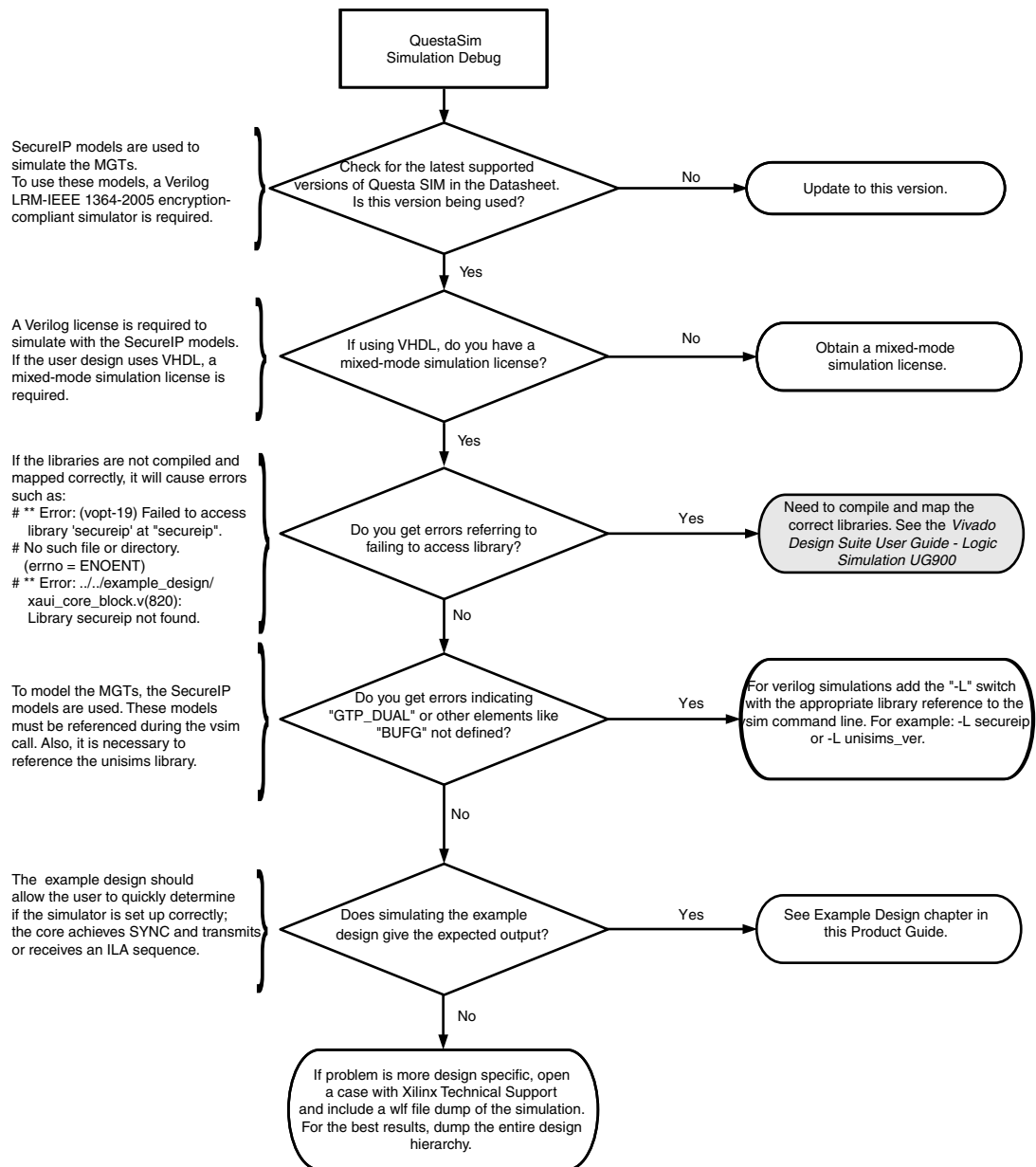


Figure B-1: QuestaSim Debug Flow Diagram

---

## Hardware Debug

Hardware issues can range from link bring-up to problems seen after hours of testing. This section provides debug steps for common issues. The debug feature is a valuable resource to use in hardware debug. The signal names mentioned in the following individual sections can be probed using the debug feature for debugging the specific problems.

### General Checks

- Ensure that all the timing constraints for the core were met during implementation.
- Ensure that all clock sources are clean and in particular that the JESD204 PHY's clocks meet the GTX/GTH/GTP/GTY transceiver requirements from the appropriate FPGA Data Sheet.
- Ensure that all GTX/GTH/GTP/GTY transceiver PLLs have obtained lock by monitoring the QPLLLOCK\_OUT and/or CPLLLOCK\_OUT port using the debug feature.
- Ensure that the core is correctly wired up.

# Linux Drivers

This appendix includes details about the JESD204 Linux Driver.

---

## Downloading Linux Drivers

The JESD204 driver can be downloaded from the Xilinx [Github](#) site.

Follow standard Linux procedures to build and insert into the kernel.

Xilinx Petalinux reference material can be found at:

<https://www.xilinx.com/products/design-tools/embedded-software/petalinux-sdk.html>

# Additional Resources and Legal Notices

---

## Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see [Xilinx Support](#).

---

## References

These documents provide supplemental material useful with this product guide:

1. *Serial Interface for Data Converters* ([JESD204B](#))
2. *JESD204 LogiCORE IP Product Guide* ([PG066](#))
3. *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* ([UG994](#))
4. *Vivado Design Suite User Guide: Designing with IP* ([UG896](#))
5. *Vivado Design Suite User Guide: Getting Started* ([UG910](#))
6. *Vivado Design Suite User Guide: Logic Simulation* ([UG900](#))
7. *ISE to Vivado Design Suite Migration Guide* ([UG911](#))
8. *Vivado Design Suite User Guide: Programming and Debugging* ([UG908](#))
9. *Vivado Design Suite User Guide: Implementation* ([UG904](#))
10. *UltraScale Architecture GTH Transceivers User Guide* ([UG576](#))
11. *7 Series FPGAs GTX/GTH Transceivers User Guide* ([UG476](#))
12. *7 Series FPGAs GTP Transceivers User Guide* ([UG482](#))
13. *Artix-7 FPGAs Data Sheet* ([DS181](#))
14. *Kintex-7 FPGAs Data Sheet* ([DS182](#))
15. *Virtex-7 T and XT FPGAs Data Sheet* ([DS183](#))
16. *Kintex UltraScale Architecture Data Sheet* ([DS892](#))
17. *Virtex UltraScale Architecture Data Sheet* ([DS893](#))

18. UltraScale Architecture GTY Transceivers User Guide ([UG578](#))

## Revision History

The following table shows the revision history for this document.

Date	Version	Revision
04/05/2017	3.3	For 64-bit interface only. The following ports were removed: <ul style="list-style-type: none"> <li>• tx/rx_core_clk_out</li> <li>• tx/rx_userclk</li> <li>• tx/rx_userclk_out</li> <li>• and tx/rx_core_clk was added</li> </ul>
10/05/2016	3.2	<ul style="list-style-type: none"> <li>• Added new configuration parameters for static and dynamic line rate and insertion loss at Nyquist</li> <li>• Added ports rxencommaalign and pcsrsvdin</li> </ul>
06/08/2016	3.1	<ul style="list-style-type: none"> <li>• Updated System Reset section.</li> <li>• Updated and clarified RXLPMEN and TXDIFFCTRL registers.</li> </ul>
04/06/2016	3.1	<ul style="list-style-type: none"> <li>• Added support for 64 bit interface with 64b/66b encoding</li> <li>• Added support for UltraScale+</li> </ul>
11/18/2015	3.0	Added support for UltraScale+ families.
09/30/2015	3.0	<ul style="list-style-type: none"> <li>• Resource Utilization removed (now online).</li> <li>• Updated Figures 1-1 and 1-2 to add tx/rx_sys_reset signals.</li> <li>• Added support for GTY Transceivers.</li> <li>• Removed registers Common DRP select (0x100) and PLL select (0x300)</li> <li>• Removed registers Transceiver DRP select (0x200), Transceiver Select Bank 1 (0x400), Transceiver Select Bank 2 (0x500), Transceiver Select Bank 3 (0x600)</li> <li>• Added mmcm_lock ports for GTP transceivers</li> </ul>

Date	Version	Revision
04/01/2015	2.0	<ul style="list-style-type: none"> <li>• Updated Applications section.</li> <li>• Added GT Port important note in Transceiver Control and Status Ports section.</li> <li>• Updated Table 2-4: Common Clock and Reset Ports.</li> <li>• Added qpll0_reset_out and qpll1_reset_out to Table 2-5: Clocks and Resets for Shared Logic in Example Design.</li> <li>• Added Register Space and Line Rate Switching section.</li> <li>• Added JESD204 PHY Configuration Options section.</li> <li>• Updated Clocking section.</li> <li>• Updated Fig. 4-1: Configuration Tab.</li> <li>• Added Optional Settings in Configuration Tab section.</li> <li>• Updated User Parameters.</li> <li>• Added constraint file in Required Constraints section.</li> <li>• Added UNISIM important note in Simulation section.</li> <li>• Added AXI Interface section in Test Bench chapter.</li> </ul>
10/01/2014	1.0	Initial Xilinx release.



---

## Please Read: Important Legal Notices

The information disclosed to you hereunder (the "Materials") is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of Xilinx's limited warranty, please refer to Xilinx's Terms of Sale which can be viewed at <http://www.xilinx.com/legal.htm#tos>; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx's Terms of Sale which can be viewed at <http://www.xilinx.com/legal.htm#tos>.

### **AUTOMOTIVE APPLICATIONS DISCLAIMER**

AUTOMOTIVE PRODUCTS (IDENTIFIED AS "XA" IN THE PART NUMBER) ARE NOT WARRANTED FOR USE IN THE DEPLOYMENT OF AIRBAGS OR FOR USE IN APPLICATIONS THAT AFFECT CONTROL OF A VEHICLE ("SAFETY APPLICATION") UNLESS THERE IS A SAFETY CONCEPT OR REDUNDANCY FEATURE CONSISTENT WITH THE ISO 26262 AUTOMOTIVE SAFETY STANDARD ("SAFETY DESIGN"). CUSTOMER SHALL, PRIOR TO USING OR DISTRIBUTING ANY SYSTEMS THAT INCORPORATE PRODUCTS, THOROUGHLY TEST SUCH SYSTEMS FOR SAFETY PURPOSES. USE OF PRODUCTS IN A SAFETY APPLICATION WITHOUT A SAFETY DESIGN IS FULLY AT THE RISK OF CUSTOMER, SUBJECT ONLY TO APPLICABLE LAWS AND REGULATIONS GOVERNING LIMITATIONS ON PRODUCT LIABILITY.

© Copyright 2014–2017 Xilinx, Inc. Xilinx, the Xilinx logo, Artix, ISE, Kintex, Spartan, Virtex, Vivado, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. All other trademarks are the property of their respective owners.