

Versal Adaptive SoC GTM Transceivers

Architecture Manual

AM017 (v1.1) September 5, 2024

AMD Adaptive Computing is creating an environment where employees, customers, and partners feel welcome and included. To that end, we're removing non-inclusive language from our products and related collateral. We've launched an internal initiative to remove language that could exclude people or reinforce historical biases, including terms embedded in our software and IPs. You may still find examples of non-inclusive language in our older products as we work to make these changes and align with evolving industry standards. Follow this [link](#) for more information.



Table of Contents

Chapter 1: Transceiver and Tool Overview.....	4
Introduction to Versal Adaptive SoCs.....	4
Features.....	6
Simulation.....	9
Implementation.....	11
Chapter 2: Shared Features.....	12
Reference Clock Input/Output Structure.....	12
Reference Clock Selection and Distribution.....	18
LC Tank PLL.....	25
Reset and initialization.....	32
Rate Change.....	59
Power Down.....	62
Loopback.....	63
Fabric Configuration Interface.....	65
Digital Monitor.....	68
Chapter 3: Transmitter.....	75
TX Interface.....	76
TX FIFO.....	80
TX Pattern Generator.....	83
TX Gray Encoder.....	86
TX Pre-Coder.....	87
TX Polarity Control.....	88
TX Fabric Clock Output Control.....	88
TX Configurable Driver.....	92
Chapter 4: Receiver.....	102
RX Analog Front End.....	103
RX Equalizer.....	108
RX CDR.....	117
RX Fabric Clock Output Control.....	118



RX Margin Analysis.....	123
RX Polarity Control.....	124
RX Pre-Coder.....	124
RX Gray Decoder.....	125
RX Pattern Checker.....	126
RX FIFO.....	129
RX Interface.....	132
Chapter 5: Board Design Guidelines.....	136
Pin Description and Design Guidelines.....	136
Reference Clock.....	139
GTM Transceiver Reference Clock Checklist.....	140
Reference Clock Interface.....	141
Power Supply.....	143
Power Supply Distribution Network Staged Decoupling.....	149
PCB Design Checklist.....	150
Appendix A: Additional Resources and Legal Notices.....	154
Finding Additional Documentation.....	154
Support Resources.....	155
References.....	155
Revision History.....	155
Please Read: Important Legal Notices.....	157

Transceiver and Tool Overview

Introduction to Versal Adaptive SoCs

AMD Versal™ adaptive SoCs combine programmable logic (PL), processing system (PS), and AI Engines with leading-edge memory and interfacing technologies to deliver powerful heterogeneous acceleration for any application. The hardware and software are targeted for programming and optimization by data scientists and software and hardware developers. A host of tools, software, libraries, IP, middleware, and frameworks enable Versal adaptive SoCs to support all industry-standard design flows.

The Versal portfolio is the first platform to combine software programmability and domain-specific hardware acceleration with the adaptability necessary to meet today's rapid pace of innovation. The portfolio is uniquely architected to deliver scalability and AI inference capabilities for a host of applications across different markets—from cloud—to networking—to wireless communications—to edge computing and endpoints.

The Versal architecture has a wealth of connectivity and communication capability and a programmable network on chip (NoC) to enable seamless memory-mapped access to the full height and width of the device. AI Engines are SIMD VLIW vector processors for adaptive inference and advanced signal processing compute. The PL combines configurable logic blocks, memory, and DSP Engines architected for high-compute density. The PS includes application and real-time processors from Arm® for intensive compute tasks.

The Versal AI Edge Series focuses on AI performance per watt for real-time systems in automated drive, predictive factory and healthcare systems, multi-mission payloads in aerospace & defense, and a breadth of other applications. More than just AI, the Versal AI Edge Series accelerates the whole application from sensor to AI to real-time control, all with the highest levels of safety and security to meet critical standards such as ISO26262 and IEC 61508.

The Versal AI Core Series delivers breakthrough AI inference acceleration and compute performance. This series is designed for a breadth of applications, including cloud for dynamic workloads and network for massive bandwidth, all while delivering advanced safety and security features. AI and data scientists, as well as software and hardware developers, can all take advantage of the high-compute density to accelerate the performance of any application.

The Versal Prime Series is the foundation and the mid-range of the Versal portfolio, serving the broadest range of uses across multiple markets. These applications include 100G to 200G networking equipment, network and storage acceleration in the data center, communications test equipment, broadcast, and aerospace & defense. The series integrates mainstream 58G transceivers and optimized I/O and DDR connectivity, achieving low-latency acceleration and performance across diverse workloads.

The Versal Premium Series provides breakthrough heterogeneous integration, very high-performance compute, connectivity, and security in an adaptable platform with a minimized power and area footprint. The series is designed to exceed the demands of high-bandwidth, compute-intensive applications in communications, data center, test & measurement, and other applications. The Versal Premium Series includes 112G PAM4 transceivers and integrated blocks for 600G Ethernet, 600G Interlaken, PCI Express® Gen5, and high-speed cryptography.

The Versal HBM Series enables the convergence of fast memory, adaptable compute, and secure connectivity in a single platform. The series is architected to keep up with the higher memory needs of the most compute intensive, memory-bound applications, providing adaptable acceleration for data center, communications, test & measurement, and aerospace & defense applications. Versal HBM adaptive SoCs integrate the most advanced HBM2e DRAM, providing high memory bandwidth and capacity within a single device.

The Versal architecture documentation suite is available at: <https://www.amd.com/versal>.

Navigating Content by Design Process

AMD Adaptive Computing documentation is organized around a set of standard design processes to help you find relevant content for your current development task. You can access the AMD Versal™ adaptive SoC design processes on the [Design Hubs](#) page. You can also use the [Design Flow Assistant](#) to better understand the design flows and find content that is specific to your intended design needs. This document covers the following design processes:

- **System and Solution Planning:** Identifying the components, performance, I/O, and data transfer requirements at a system level. Includes application mapping for the solution to PS, PL, and AI Engine. Topics in this document that apply to this design process include:
 - [Features](#)
 - [LC Tank PLL](#)
- **Hardware, IP, and Platform Development:** Creating the PL IP blocks for the hardware platform, creating PL kernels, functional simulation, and evaluating the AMD Vivado™ timing, resource use, and power closure. Also involves developing the hardware platform for system integration. Topics in this document that apply to this design process include:
 - [Simulation](#)
 - [Implementation](#)

- **System Integration and Validation:** Integrating and validating the system functional performance, including timing, resource use, and power closure. Topics in this document that apply to this design process include:
 - [Implementation](#)
 - [Reference Clock Selection and Distribution](#)
 - **Board System Design:** Designing a PCB through schematics and board layout. Also involves power, thermal, and signal integrity considerations. Topics in this document that apply to this design process include:
 - [Reference Clock Selection and Distribution](#)
-

Features

The Versal architecture GTM transceiver provides the highest performances and integration at 7 nm, including serial I/O bandwidth and logic capacity. As the industry's high-end FPGA at the 7 nm process node, this family is ideal for applications including 400G networking, large-scale ASIC prototyping, and emulation.

The GTM transceiver in the Versal architecture is a power-efficient transceiver, supporting line rates between 9.5 Gb/s and 112 Gb/s. Based on the available PLL divider configurations in the GTM transceivers, the following line rates are supported:

- NRZ Modulation:
 - 9.5 Gb/s – 15 Gb/s
 - 19 Gb/s – 29 Gb/s
- PAM4 Modulation
 - 19 Gb/s – 29 Gb/s
 - 38 Gb/s – 58 Gb/s
 - 76 Gb/s – 112 Gb/s

The Versal adaptive SoC GTM transceiver is AMD's highest performance PAM4 enabled transceiver and is highly configurable and tightly integrated with the programmable logic resources of the FPGA. The following table summarizes the features by functional group that support a wide variety of applications.

Table 1: GTM Transceiver Features

Group	Feature
PCS	PRBS generator and checker
	Programmable FPGA logic interface
PMA	LC tank oscillator PLL (LCPLL) for best jitter performance
	Flexible clocking with two LCPLLs per Quad ¹ (four channels)
	Programmable transmitter (TX) output
	TX FIR filter with de-emphasis controls
	Continuous time linear equalization (CTLE)
	Decision feedback equalization (DFE)
	Feed forward equalization (FFE)

Notes:

1. A Quad is a cluster or set of four GTM transceiver channels, one GTME5_QUAD primitive, two differential reference clock pin pairs, and analog supply pins. There is no channel primitive.
2. A GTM Quad can function in full density or half density mode. In full density mode, all four channels in the Quad are functional. In half density mode, only two channels in the quad are functional: channel 0 or 1, and channel 2 or 3.
3. Full density mode is supported for all NRZ line rates.
4. Full density mode is supported for line rates up to 58 Gb/s PAM4. Line rates greater than 58 Gb/s PAM4 must use half density mode.
5. Half density mode must be used by designs using fabric interface data widths of 320 and 512 bits. For more information, see [TX Interface](#) and [RX Interface](#).

The GTM transceiver supports NRZ and PAM4 modulation as well as the following protocols:

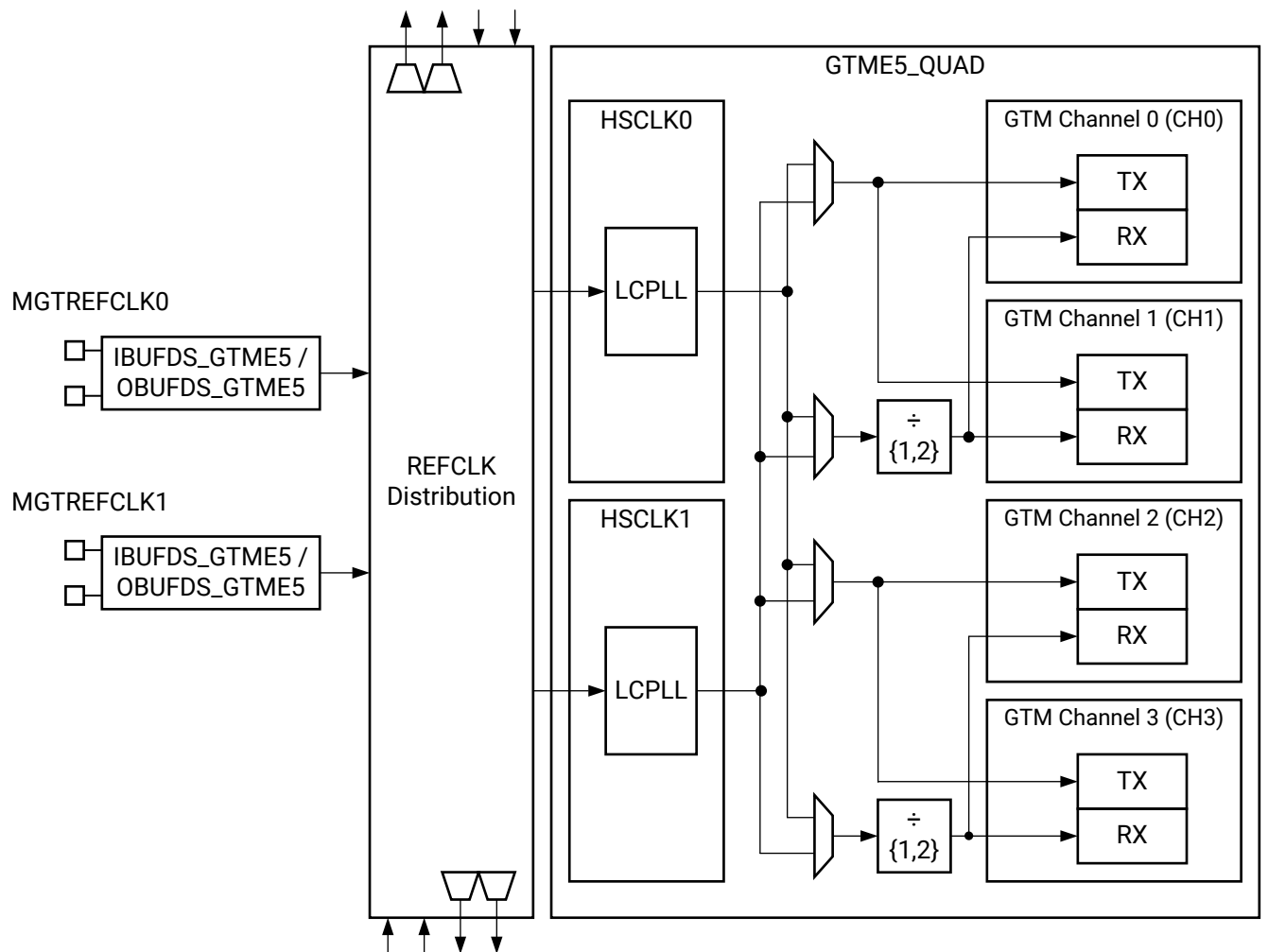
- 400GE CDAUI4
- 400GE CDAUI8
- 100GE CAUI2
- 100GE CAUI4
- 100GE CAUI1
- 50GE LAUI
- 50GE LAUI2
- Multirate CPRI from 10.1 Gb/s to 100 Gb/s
- Interlaken at 51.5625 Gb/s, 25 Gb/s, 12.5 Gb/s
- OTU4
- 112G XSR
- 56 Gb/s PAM4 Backplane
- 56.52 Gb/s PAM4 Midplanes
- 28.21 Gb/s PAM4 Backplanes
- 10GBASE-KR

- 802.3bj Backplanes and Cables

The first-time user is recommended to read [High-Speed Serial I/O Made Simple](#), which discuss high-speed serial transceiver technology and its applications. The AMD Vivado™ IP integrator or the transceiver Wizard design flow is recommended to automatically configure the GTM transceiver to support configurations for different protocols or perform custom configurations. The GTM transceiver offers a data rate range and features that allow physical layer support for various protocols.

The following figure illustrates the clustering of four transceiver channels and two high speed clocking (HSClk) blocks to form the GTME5_QUAD primitive.

Figure 1: GTM Quad Configuration

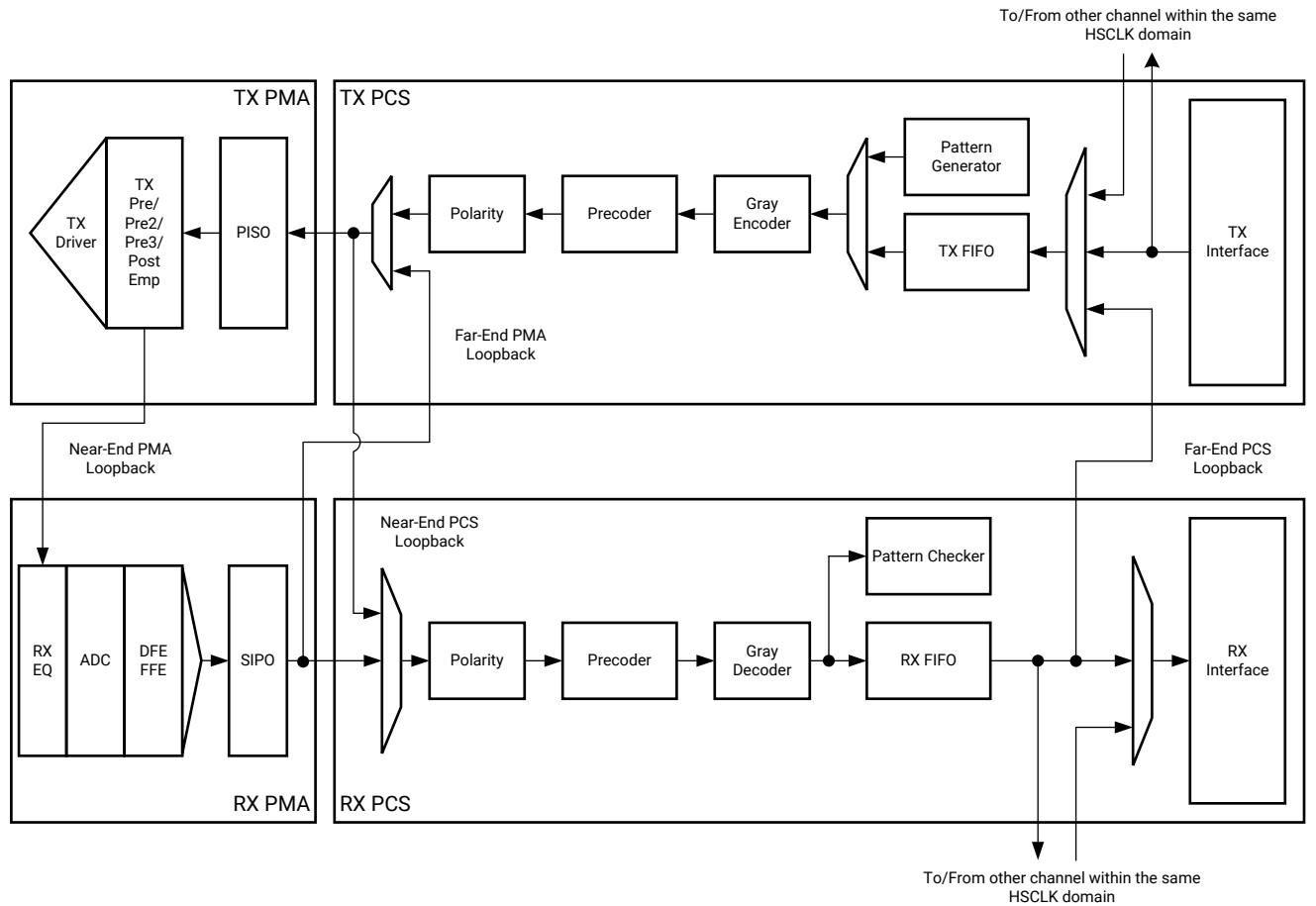


X23627-053020

Four channel clustered together with two HSCLK blocks form a Quad or Q. Each HSCLK block contain one LC-tank PLL (LCPLL). The LCPLL from each HSCLK can provide clock to any of the four channels inside the same Quad. Each channel consists of a transmitter and a receiver. The following figure illustrates the topology of the GTM channel.

★ IMPORTANT! Half density mode must be used for line rates greater than 58 Gb/s (PAM4), and for fabric interface data widths of 320 and 512 bits. For more information, see [TX Interface](#) and [RX Interface](#).

Figure 2: GTM Channel Topology



X23626-061721

Simulation

The simulation environment and the test bench must fulfill specific prerequisites before running simulation using the GTME5_QUAD primitives. For instructions on how to set up the simulation environment for supported simulators depending on the used hardware description language (HDL) used, see the *Vivado Design Suite User Guide: Logic Simulation* ([UG900](#)).

The prerequisites for simulating a design with the GTME5_QUAD primitives are as follows:

- A simulator with support for Secure IP models: SecureIP is an IP encryption methodology. SecureIP models are encrypted versions of the System Verilog HDL used for implementation of the modeled block. To support SecureIP models, a simulator that complies with the encryption standards described in the [IEEE Standard for Verilog Hardware Description Language \(IEEE Std 1364-2005\)](#) is required.
- A mixed-language simulator for VHDL simulation: SecureIP models use a System Verilog standard. To use them in a VHDL design, a mixed-language simulator is required. The simulator must be able to simulate VHDL and Verilog simultaneously.
- An installed GTM transceiver SecureIP model.
- The correct setup of the simulator for SecureIP use (initialization file, environment variables).
- The correct simulator resolution (Verilog).

Ports and Attributes

There are no simulation-only ports on the GTME5_QUAD primitives.

GTME5_QUAD Attributes

The GTME5_QUAD primitive has attributes intended only for simulation. The following table lists the simulation-only attributes of the GTME5_QUAD primitive. The names of these attributes start with SIM_.

Table 2: GTME5_QUAD Simulation-Only Attributes

Attribute	Type	Description
SIM_VERSION	String	<p>This attribute selects the simulation version to match different revisions of silicon:</p> <p>VERSAL_[*]_ES1: Engineering samples 1 VERSAL_[*]_ES2: Engineering samples 2 VERSAL_[*]: Production silicon</p> <p>Where [*] denotes the actual Versal device:</p> <p>AI_CORE AI_EDGE AI_RF HBM PREMIUM PRIME</p>
QUAD_SIM_MODE	String	This attribute selects the simulation mode. The default value is FAST.

Table 2: GTME5_QUAD Simulation-Only Attributes (cont'd)

Attribute	Type	Description
QUAD_SIM_RESET_SPEEDUP	String	When set to TRUE (default), an approximate reset sequence is used to speed up the reset time for simulations, where faster reset times and faster simulation time are desired. When set to FALSE, the model emulates hardware reset behavior in detail.
CH[0/1/2/3]_SIM_MODE	String	This attribute selects the simulation mode. The default value is FAST. The value should match QUAD_SIM_MODE.
CH[0/1/2/3]_SIM_RESET_SPEEDUP		When set to TRUE (default), an approximate reset sequence is used to speed up the reset time for simulations, where faster reset times and faster simulation time are desired. When set to FALSE, the model emulates hardware reset behavior in detail. The value should match QUAD_SIM_RESET_SPEEDUP.
CH[0/1/2/3]_SIM_RECEIVER_DETECT_PASS	String	Determines if receiver detection should indicate a pass or fail by setting this attribute to TRUE (default) or FALSE, respectively.

Implementation

It is a common practice to define the location of transceiver Quads early in the design process to ensure correct usage of clock resources and to facilitate signal integrity analysis during board design. The implementation flow facilitates this practice through the use of location constraints in the XDC file.

The position of each GTM transceiver Quad primitive is specified by an XY coordinate system that describes the column number and the relative position within that column.

Use the I/O planner in the Vivado tools to set the transceiver locations. Alternatively, user designs can manually add placement locations for the GTM transceiver. Care must be taken to ensure that all of the parameters needed to configure the transceiver are correctly entered. For more information on GTM transceiver placement, see *Versal Adaptive SoC Transceivers Wizard LogiCORE IP Product Guide* ([PG331](#)).

Shared Features

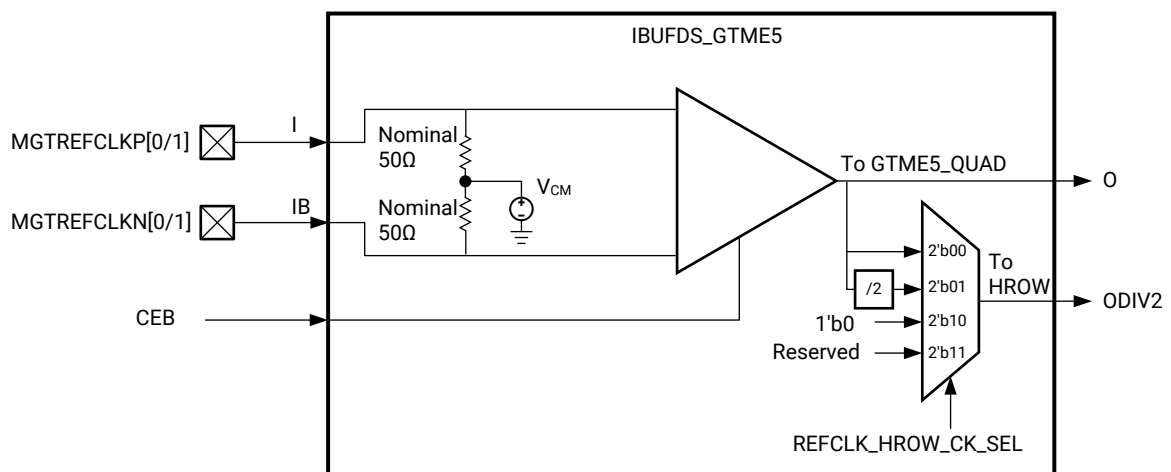
Reference Clock Input/Output Structure

The reference clock structure in the GTM transceivers support two modes of operation: input mode and output mode. In the input mode of operation, the design provides a clock on the dedicated reference clock I/O pins that are used to drive the LCPLLs. In the output mode of operation, the recovered clocks (HSCLK*_RXRECCLKOUT0/1) from any of the four channels within the same Quad can be routed to the dedicated reference clock I/O pins. This output clock can then be used as the reference clock input at a different location. The mode of operation cannot be changed during run time.

Input Mode

The reference clock input mode structure is illustrated in the following figure. The input is terminated internally with 50Ω on each leg to V_{CM} . The reference clock is instantiated in software with the IBUFDS_GTME5 software primitive. The ports and attributes controlling the reference clock input are tied to the IBUFDS_GTME5 software primitive.

Figure 3: Reference Clock Input Structure



X25485-071024

IMPORTANT! Upon device configuration, the clock output from the IBUFDS_GTME5 which takes inputs from MGTREFCLKP[0/1] and MGTREFCLKN[0/1] can only be used under the following conditions:

- The GTPOWERGOOD signal has already asserted High.

Ports and Attributes

The following table defines the reference clock input ports in the IBUFDS_GTME5 software primitive.

Table 3: Reference Clock Input Ports (IBUFDS_GTME5)

Port	Dir	Clock Domain	Description
CEB	In	N/A	This is the active-Low asynchronous clock enable signal for the clock buffer. Setting this signal High powers down the clock buffer. This port cannot be left unconnected.
I	In (pad)	N/A	These are the reference clock input ports that get mapped to MGTREFCLKP0 and MGTREFCLKP1.
IB	In (pad)	N/A	These are the reference clock input ports that get mapped to MGTREFCLKN0 and MGTREFCLKN1.
O	Out	N/A	This output drives the GTME5_QUAD primitive. Refer to Reference Clock Selection and Distribution for more details.
ODIV2	Out	N/A	This output can be configured to output either the O signal or a divide-by-2 version of the O signal. It can drive the BUFG_GT via the HROW routing. Refer to Reference Clock Selection and Distribution for more details.

The following table defines the attributes in the IBUFDS_GTME5 software primitive that configure the reference clock input.

Table 4: Reference Clock Input Attributes (IBUFDS_GTME5)

Attribute	Type	Address	Description
REFCLK0_REFCLK_EN_TX_PATH	1-bit Binary	0x0CBC[17]	Reserved. This attribute must always be set to 1'b0.
REFCLK1_REFCLK_EN_TX_PATH	1-bit Binary	0x0EBC[17]	Reserved. This attribute must always be set to 1'b0.
REFCLK0_REFCLK_HROW_CK_SEL	Integer	0x0CBC[24:23]	Configures the ODIV2 output port: 0: ODIV2 = O. 1: ODIV2 = Divide-by-2 version of O. 2: ODIV2 = 1'b0. 3: ODIV2 = Reserved.
REFCLK1_REFCLK_HROW_CK_SEL	Integer	0x0EBC[24:23]	Configures the ODIV2 output port: 0: ODIV2 = O. 1: ODIV2 = Divide-by-2 version of O. 2: ODIV2 = 1'b0. 3: ODIV2 = Reserved.

Table 4: Reference Clock Input Attributes (IBUFDS_GTME5) (cont'd)

Attribute	Type	Address	Description
REFCLK0_REFCLK_ICNTL_RX	Integer	0x0CBC[22:21]	Reserved. Use the recommended value from the Wizard.
REFCLK1_REFCLK_ICNTL_RX	Integer	0x0EBC[22:21]	Reserved. Use the recommended value from the Wizard.

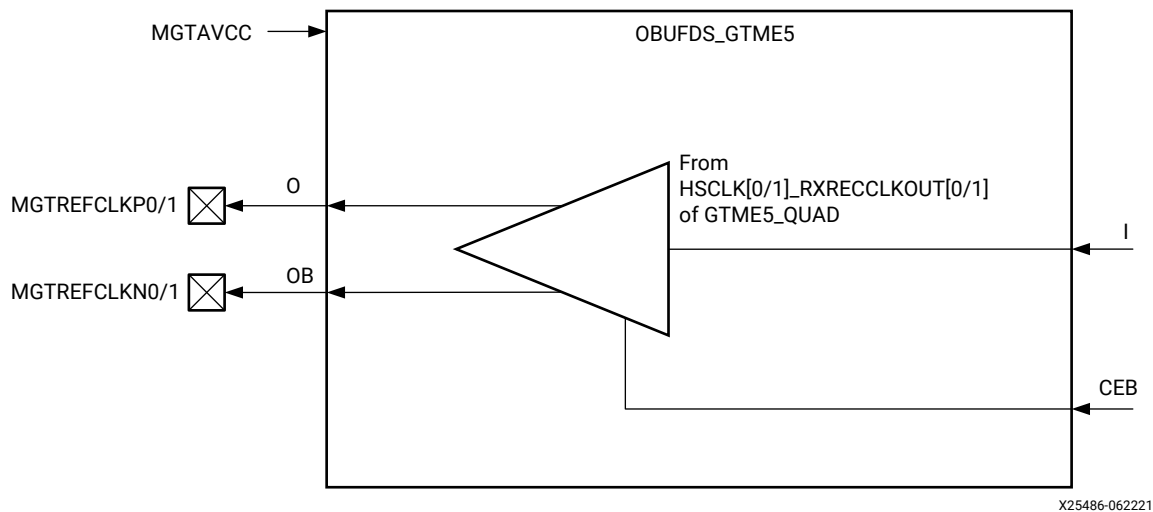
Output Mode

The reference clock output mode can be accessed via one of the two software primitives: OBUFDS_GTME5 and OBUFDS_GTME5_ADV. The choice of the primitive depends on your application. Use OBUFDS_GTME5 when the CH*_RXRECCLKOUT[0/1] is always derived from the same channel. Use OBUFDS_GTME5_ADV if the channel providing CH*_RXRECCLKOUT[0/1] can change during run time.

OBUFDS_GTME5

The reference clock output mode structure with the OBUFDS_GTME5 primitive is shown in the following figure. The ports and attributes controlling the reference clock output are tied to the OBUFDS_GTME5 software primitive.

Figure 4: Reference Clock Output Use Model with OBUFDS_GTME5



Ports and Attributes

The following table defines the ports in the OBUFDS_GTME5 software primitive.

Table 5: Reference Clock Output Ports (OBUFDS_GTME5)

Port	Dir	Clock Domain	Description
CEB	In	N/A	This is the active-Low asynchronous clock enable signal for the clock buffer. Setting this signal High powers down the clock buffer.
I	In	N/A	Recovered clock input. Connect to the output port HSCLK[0/1]_RXRECCLKOUT[0/1] of one of the four channels in the GTME5_QUAD.
O	Out	N/A	Reference clock output port that gets mapped to MGTREFCLKP0 and MGTREFCLKP1.
OB	Out	N/A	Reference clock output port that gets mapped to MGTREFCLKN0 and MGTREFCLKN1.

The following table defines the attributes in the OBUFDS_GTME5 software primitive that configure the reference clock output.

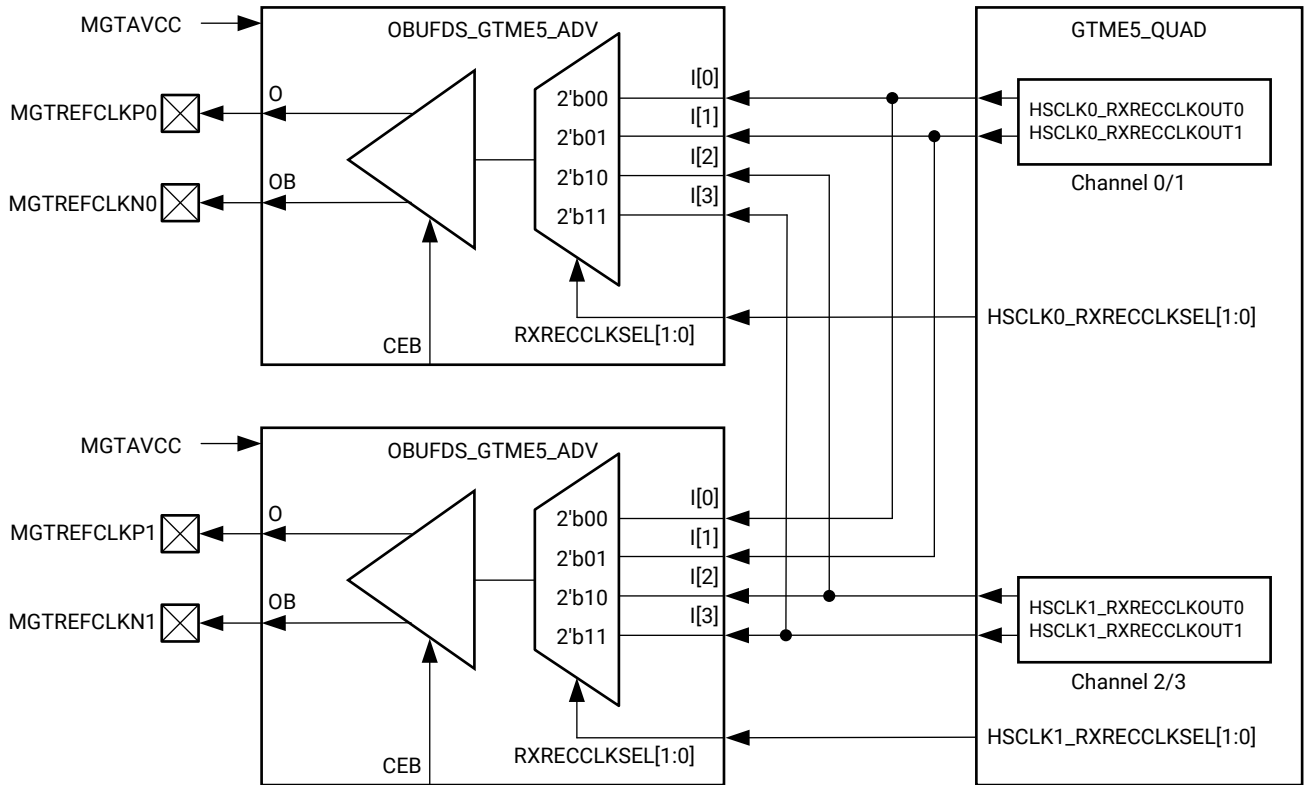
Table 6: Reference Clock Output Attributes (OBUFDS_GTME5)

Attribute	Type	Address	Description
REFCLK0_REFCLK_EN_TX_PATH	1-bit Binary	0x0CBC[17]	Reserved. This attribute must always be set to 1'b1.
REFCLK1_REFCLK_EN_TX_PATH	1-bit Binary	0x0EBC[17]	Reserved. This attribute must always be set to 1'b1.
REFCLK0_REFCLK_EN_DRV	1-bit Binary	0x0CBC[1]	Reserved. This attribute must always be set to 1'b1.
REFCLK1_REFCLK_EN_DRV	1-bit Binary	0x0EBC[1]	Reserved. This attribute must always be set to 1'b1.

OBUFDS_GTME5_ADV

The reference clock output mode structure with the OBUFDS_GTME5_ADV primitive is shown in the following figure. The ports and attributes controlling the reference clock output are tied to the OBUFDS_GTME5_ADV software primitives. The port RXRECCLKSEL controls the multiplexer that selects between HSCLK*_RXRECCLKOUT[0/1] from the four different channels in a Quad.

Figure 5: Reference Clock Output Use Model with OBUFDS_GTME5_ADV



X25487-062221

Ports and Attributes

The following table defines the ports in the OBUFDS_GTME5_ADV software primitive.

Table 7: Reference Clock Output Ports (OBUFDS_GTME5_ADV)

Port	Dir	Clock Domain	Description
CEB	In	N/A	This is the active-Low asynchronous clock enable signal for the clock buffer. Setting this signal High powers down the clock buffer.
I[3:0]	In	N/A	Recovered clock input bus. Connect I[0] to the HSCLK0_RXRECCLKOUT0 of GTME5_QUAD mapping to channel 0. Connect I[1] to the HSCLK0_RXRECCLKOUT1 of GTME5_QUAD mapping to channel 1. Connect I[2] to the HSCLK1_RXRECCLKOUT0 of GTME5_QUAD mapping to channel 2. Connect I[3] to the HSCLK1_RXRECCLKOUT1 of GTME5_QUAD mapping to channel 3.
O	Out	N/A	Reference clock output ports that get mapped to MGTREFCLKP0 and MGTREFCLKP1.

Table 7: Reference Clock Output Ports (OBUFDS_GTME5_ADV) (cont'd)

Port	Dir	Clock Domain	Description
OB	Out	N/A	Reference clock output ports that get mapped to MGTREFCLKN0 and MGTREFCLKN1.
RXRECCLKSEL[1:0]	In	N/A	Recovered clock output selection on the MGTREFCLK[0/1] pins. This port is directly controlled by the HSCLK[0/1]_RXRECCLKSEL[1:0] port from the GTME5_QUAD primitive. The HSCLK[0/1]_RXRECCLKSEL[1:0] port from the GTME5_QUAD primitive is set by the HSCLK[0/1]_REFCLK_RXRECCLK_SEL[1:0] attribute.

Table 8: Reference Clock Output Ports (GTME5_QUAD)

Port	Dir	Clock Domain	Description
HSCLK0_RXRECCLKSEL[1:0]	Out	N/A	Directly connects to the RXRECCLKSEL[1:0] pins on the OBUFDS_GTME5_ADV primitive, which controls the recovered clock selection on the MGTREFCLK[P/N]0 pins. This GTME5_QUAD output port is set by the HSCLK0_REFCLK0_RXRECCLK_SEL attribute.
HSCLK1_RXRECCLKSEL[1:0]	Out	N/A	Directly connects to the RXRECCLKSEL[1:0] pins on the OBUFDS_GTME5_ADV primitive, which controls the recovered clock selection on the MGTREFCLK[P/N]1 pins. This GTME5_QUAD output port is set by the HSCLK1_REFCLK1_RXRECCLK_SEL attribute.

The following table defines the attributes in the OBUFDS_GTME5_ADV software primitive that configure the reference clock output.

Table 9: Reference Clock Output Attributes (OBUFDS_GTME5_ADV)

Attribute	Type	Address	Description
REFCLK0_REFCLK_EN_TX_PATH	1-bit Binary	0x0CBC[17]	Reserved. This attribute must always be set to 1'b1.
REFCLK1_REFCLK_EN_TX_PATH	1-bit Binary	0x0EBC[17]	Reserved. This attribute must always be set to 1'b1.
REFCLK0_REFCLK_EN_DRV	1-bit Binary	0x0CBC[1]	Reserved. This attribute must always be set to 1'b1.
REFCLK1_REFCLK_EN_DRV	1-bit Binary	0x0EBC[1]	Reserved. This attribute must always be set to 1'b1.

Table 10: Reference Clock Output Attributes (GTME5_QUAD)

Attribute	Type	Address	Description
HSCLK0_REFCLK0_RXRECCLK_SEL	Integer	0x0CBC[14:13]	Recovered clock output selection control on the MGTREFCLK0 pin. 2'b00: From channel 0 RXPROGDIV (HSCLK0_RXRECCLKOUT0). 2'b01: From channel 1 RXPROGDIV (HSCLK0_RXRECCLKOUT1). 2'b10: From channel 2 RXPROGDIV (HSCLK1_RXRECCLKOUT0). 2'b11: From channel 3 RXPROGDIV (HSCLK1_RXRECCLKOUT1).
HSCLK1_REFCLK1_RXRECCLK_SEL	Integer	0x0EBC[14:13]	Recovered clock output selection control on pin MGTREFCLK1. 2'b00: From channel 0 RXPROGDIV (HSCLK0_RXRECCLKOUT0). 2'b01: From channel 1 RXPROGDIV (HSCLK0_RXRECCLKOUT1). 2'b10: From channel 2 RXPROGDIV (HSCLK1_RXRECCLKOUT0). 2'b11: From channel 3 RXPROGDIV (HSCLK1_RXRECCLKOUT1).

Reference Clock Selection and Distribution

The Versal adaptive SoC transceivers provide different reference clock input options. Clock selection supports two LCPLLs.

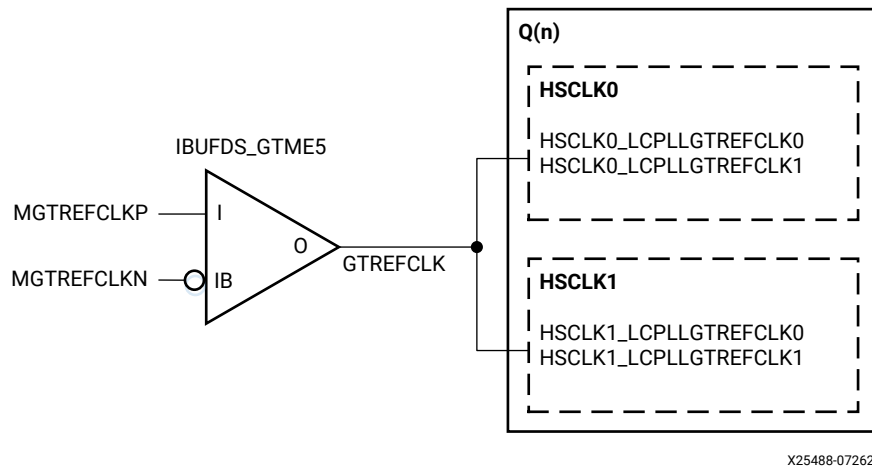
From an architecture perspective, a Quad (or Q) contains four channels, two HSCLK blocks, two dedicated external reference clock pin pairs, and dedicated reference clock routing. An additional constraint on the above resources within a Quad is that there is one LCPLL within each of the HSCLK0/1 blocks.

The GTME5_QUAD primitive must be instantiated when any PLL or transceiver channel inside the Quad is used. For duals operating at line rates from 9.5 Gb/s to 15 Gb/s, 19 Gb/s to 29 Gb/s, and 38 Gb/s to 58 Gb/s, the reference clock for a Quad (Q(n)) can also be sourced from up to two Quads below (Q(n-1) or Q(n-2)) or from up to two Quads above (Q(n+1) or Q(n+2)). For devices that support stacked silicon interconnect (SSI) technology, the reference clock sharing is limited within its own super logic region (SLR).

See the [Versal device data sheets](#) for more information about SSI technology.

Each Quad has two dedicated differential reference clock input pins (MGTREFCLK[P/N]0 or MGTREFCLK[P/N]1) that can be connected to the external clock sources. In a single external reference use model, an IBUFDS_GTME5 must be instantiated to use one of the dedicated differential reference clock sources. The following figure shows a single external reference clock connected to multiple transceivers within a single Quad. The user design connects the IBUFDS_GTME5 output (O) to the corresponding HSCLK[0/1]_LCPLLGTREFCLK[0/1] ports of the GTME5_QUAD for the GTM transceiver.

Figure 6: Single External Reference Clock with Multiple Transceivers in a Single Quad



For Versal devices, sourcing of the reference clock is limited to two Quads above and below for duals operating at line rates from 9.5 Gb/s to 29 Gb/s (NRZ) and 19 Gb/s to 58 Gb/s (PAM4). Channels should source a local reference clock (from within its own Quad) for highest performance.

Reference clock features include:

- Clock routing for northbound and southbound clocks.
- Flexible clock inputs available for the LCPLL.
- Static or dynamic selection of the reference clock for the LCPLL.

The Quad architecture has four GTM transceivers, two dedicated reference clock pin pairs, and dedicated north or south reference clock routing. Each GTM transceiver channel in a Quad has six clock inputs available according to the corresponding PLL resource assignments shown in the following table.

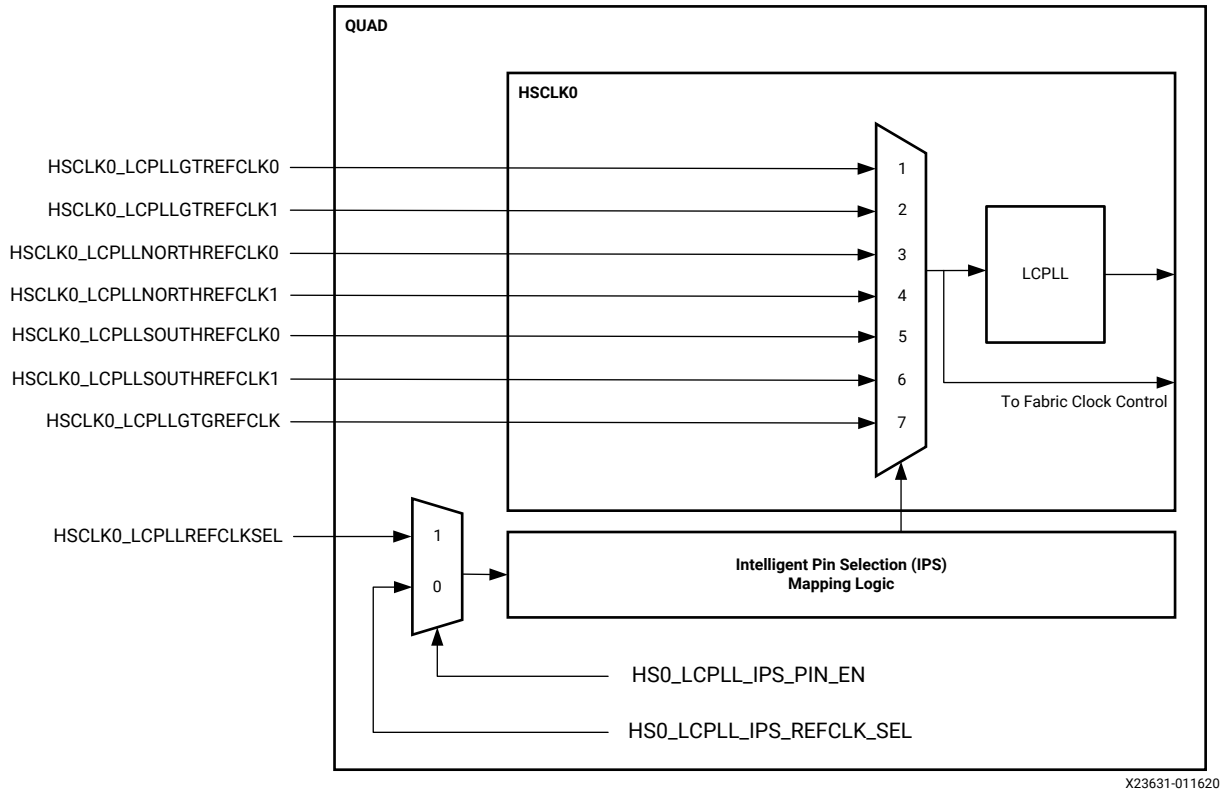
Table 11: Reference Clock Sharing for Versal Adaptive SoC GTM Transceivers

Clock Source	PLL Used	GTM Transceivers Channel 0/1	GTM Transceivers Channel 2/3
Two local reference clock pin pairs	LCPLL	HSCLK0_LCPLLGTREFCLK0 HSCLK0_LCPLLGTREFCLK1	HSCLK1_LCPLLGTREFCLK0 HSCLK1_LCPLLGTREFCLK1
Two reference clock pin pairs from Quads above	LCPLL	HSCLK0_LCPLLSOUTHREFCLK0 HSCLK0_LCPLLSOUTHREFCLK1	HSCLK1_LCPLLSOUTHREFCLK0 HSCLK1_LCPLLSOUTHREFCLK1
Two reference clock pin pairs from Quads below	LCPLL	HSCLK0_LCPLLNORTHREFCLK0 HSCLK0_LCPLLNORTHREFCLK1	HSCLK1_LCPLLNORTHREFCLK0 HSCLK1_LCPLLNORTHREFCLK1

Because there are only two south clock inputs and four potential clock sources from the two Quads above ($Q(n+1)$ and $Q(n+2)$), only a maximum of two of the four potential reference clock pin pairs from above can be physically connected up to $Q(n)$ at any given moment. The four potential reference clock pin pairs from above are reduced to two or three if the Quad above ($Q(n+1)$) is itself sourcing reference clock pin pairs from two above ($Q(n+3)$). This is because there are a total of two south reference clock routing tracks connecting the Quads. Similar rules apply when sourcing a reference clock from Quads below. Because there are two north clock inputs and four potential clock sources from the two Quads below ($Q(n-1)$ and $Q(n-2)$), only a maximum of two of the four potential reference clock pin pairs from below can be physically connected up to $Q(n)$ at any given moment. The four potential reference clock pin pairs from below is reduced to two or three if the Quad below ($Q(n-1)$) is itself sourcing reference clock pin pairs from two below ($Q(n-3)$). Again, this is because there are a total of two north reference clock routing tracks connecting the Quads. For example, $Q(n-1)$ is sourcing both reference clocks from $Q(n-3)$. In this example, $Q(n)$ would only be able to source reference clock pins below from $Q(n-1)$. $Q(n)$ would not be able to access the reference clock pins in $Q(n-2)$ because the two routing tracks have already been used to bring the two reference clocks from $Q(n-3)$ to $Q(n-1)$.

The following figure shows the detailed view of the reference clock multiplexer structure within a single HSCLK block, using LCPLL inside block HSCLK0 as an example. The same structure applies to LCPLL inside HSCLK1. When single or multiple reference clock sources are connected, the designer first needs to make sure that the connections are made to the correct PLLs using the reference clock ports assigned to the each intended PLL. The enhanced intelligent pin selection (IPS) automatically analyzes the design and maps reference clock selection during design implementation to guarantee that clocks are properly connected. For additional details on IPS, see [Intelligent Pin Selection](#).

Figure 7: LCPLL Reference Clock Selection Multiplexer



The following tables list the corresponding reference clock selection multiplexer settings.

Note: The connections for the two local GTREFCLKs are connected differently inside HCLK1 compared to HCLK0. For HCLK1, HCLK1_LCPLLGTREFCLK1 is connected to input 1 on the input multiplexer while HCLK1_LCPLLGTREFCLK0 is connected to input 2 on the input multiplexer.

Table 12: LCPLL Reference Clock Selection Multiplexer Settings

HCLK[0/1]_LCPLLREFCLKSEL	HS[0/1]_LCPLL_IPS_REFCLK_SEL	Selected Input to LCPLL
3'b001	1	For HCLK0: HCLK0_LCPLLGTREFCLK0 For HCLK1: HCLK1_LCPLLGTREFCLK1
3'b010	2	For HCLK0: HCLK0_LCPLLGTREFCLK1 For HCLK1: HCLK1_LCPLLGTREFCLK0
3'b011	3	HCLK[0/1]_LCPLLNORTHREFCLK0
3'b100	4	HCLK[0/1]_LCPLLNORTHREFCLK1
3'b101	5	HCLK[0/1]_LCPLLSOUTHREFCLK0
3'b110	6	HCLK[0/1]_LCPLLSOUTHREFCLK1
3'b111	7	HCLK[0/1]_LCPLLGTGREFCLK

Notes:

1. When HS[0/1]_LCPLL_IPS_PIN_EN = 0, HS[0/1]_LCPLL_IPS_REFCLK_SEL control the multiplexer inputs.
2. When HS[0/1]_LCPLL_IPS_PIN_EN = 1, HCLK[0/1]_LCPLLREFCLKSEL controls the multiplexer inputs.
3. The HCLK[0/1]_LCPLLGTGREFCLK input is reserved and should not be used.

As shown in Figure 7, the GTM transceiver in Versal adaptive SoCs contain multiple dedicated reference clock input ports for each PLL. The user should set the muxing of the reference clock input by using the selector port or the selector attribute. The following table lists the reference clock multiplexer selector control signals per PLL.

Table 13: Reference Clock Input Multiplexer Selection Control

PLL	Selector Control Attribute	Multiplexer Selector Port/Attribute
LCPLL (HCLK0)	HS0_LCPLL_IPS_PIN_EN = 1'b1	HSCLK0_LCPLLREFCLKSEL[2:0]
	HS0_LCPLL_IPS_PIN_EN = 1'b0	HS0_LCPLL_IPS_REFCLK_SEL
LCPLL (HCLK1)	HS1_LCPLL_IPS_PIN_EN = 1'b1	HSCLK1_LCPLLREFCLKSEL[2:0]
	HS1_LCPLL_IPS_PIN_EN = 1'b0	HS1_LCPLL_IPS_REFCLK_SEL

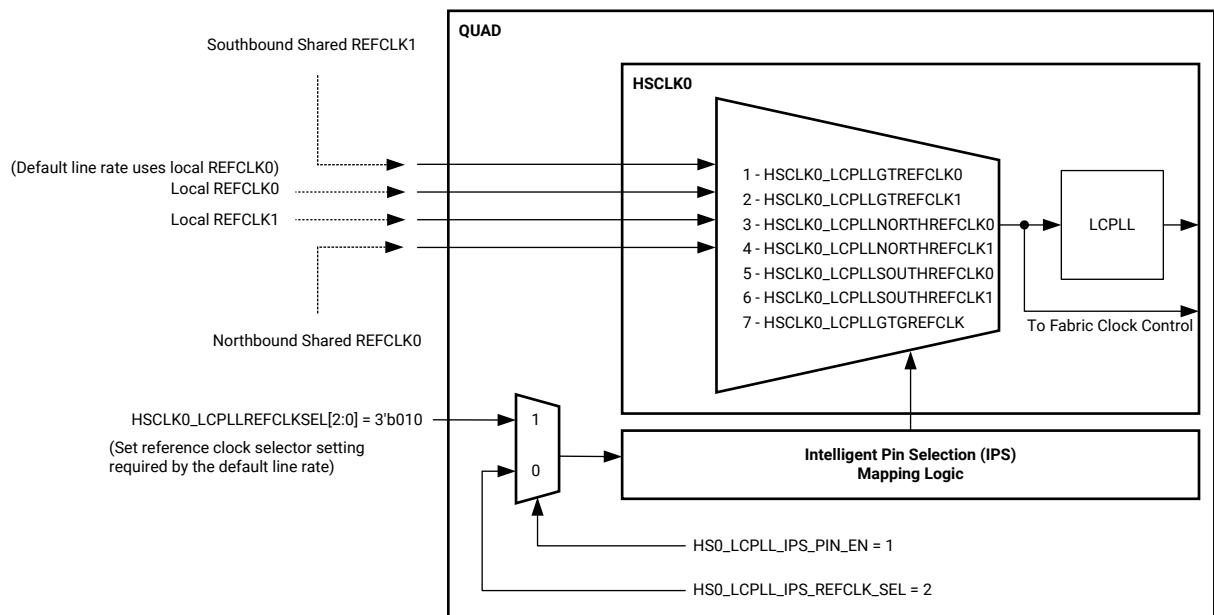
Notes:

1. When HS[0/1]_LCPLL_IPS_PIN_EN = 0, HS[0/1]_LCPLL_IPS_REFCLK_SEL controls the multiplexer inputs.
2. When HS[0/1]_LCPLL_IPS_PIN_EN = 1, HSCLK[0/1]_LCPLLREFCLKSEL controls the multiplexer inputs.
3. The HSCLK[0/1]_LCPLLGTGREFCLK input is reserved and should not be used.

Intelligent Pin Selection

In Versal devices, the intelligent pin selection (IPS) functionality provides mapping logic during implementation that attempts to route the reference clock to the desired PLL. The IPS provides a layer of mapping logic that assists the user to simplify design creation when multiple reference clocks are connected to any of the input ports on the desired PLL. This is shown in the example figure below.

Figure 8: IPS Example



X21467-050619

The recommended procedure is to connect each reference clock input from top to bottom on the input multiplexer without leaving any gaps, as shown in the example figure above. IPS automatically provides the necessary mapping during implementation to make sure each of the reference clocks is correctly connected on the PLL input multiplexer.

The list below contains other restrictions or design practices that the user should follow:

- The design must connect the reference clock to the intended PLL.
- The user must configure the reference clock input multiplexer setting to match the reference clock input port that is required by the default line rate.
- The design must not instantiate IBUFDS_GTME5 at a location that violates the reference clock sharing rules. For more information, refer to [Reference Clock Selection and Distribution](#).
- HSCLK*_LCPLLGTGREFCLK are reserved inputs and should not be used.

Ports and Attributes

The following tables define the ports and attributes relevant to reference clock selection and distribution for GTME5_QUAD primitives.

Table 14: Reference Clock Selection and Distribution Ports

Port	Direction	Clock Domain	Description
HSCLK[0/1]_LCPLLGTREFCLK0	Input	CLOCK	Local GTREFCLK0 to LCPLL in HSCLK0/1.
HSCLK[0/1]_LCPLLGTREFCLK1	Input	CLOCK	Local GTREFCLK1 to LCPLL in HSCLK0/1.
HSCLK[0/1]_LCPLLNORTHREFCLK0	Input	CLOCK	North-bound GTREFCLK0 clock from the Quad below.
HSCLK[0/1]_LCPLLNORTHREFCLK1	Input	CLOCK	North-bound GTREFCLK1 clock from the Quad below.
HSCLK[0/1]_LCPLLREFCLKSEL[2:0]	Input	ASYNC	Input to dynamically select the input reference clock to the LCPLL. Set this input to 3'b001 when only one clock source is connected to the LCPLL reference clock selection multiplexer. 3'b001: HSCLK0_LCPLLGTREFCLK0/ HSCLK1_LCPLLGTREFCLK1 3'b010: HSCLK0_LCPLLGTREFCLK1/ HSCLK1_LCPLLGTREFCLK0 3'b011: HSCLK[0/1]_LCPLLNORTHREFCLK0 3'b100: HSCLK[0/1]_LCPLLNORTHREFCLK1 3'b101: HSCLK[0/1]_LCPLLSOUTHREFCLK0 3'b110: HSCLK[0/1]_LCPLLSOUTHREFCLK1 3'b111: HSCLK[0/1]_LCPLLGTGREFCLK Reset must be applied to the LCPLL after changing the reference clock input.
HSCLK[0/1]_LCPLLSOUTHREFCLK0	Input	CLOCK	South-bound GTREFCLK0 clock from the Quad above.
HSCLK[0/1]_LCPLLSOUTHREFCLK1	Input	CLOCK	South-bound GTREFCLK1 clock from the Quad above.

Table 15: Reference Clock Selection and Distribution Attributes

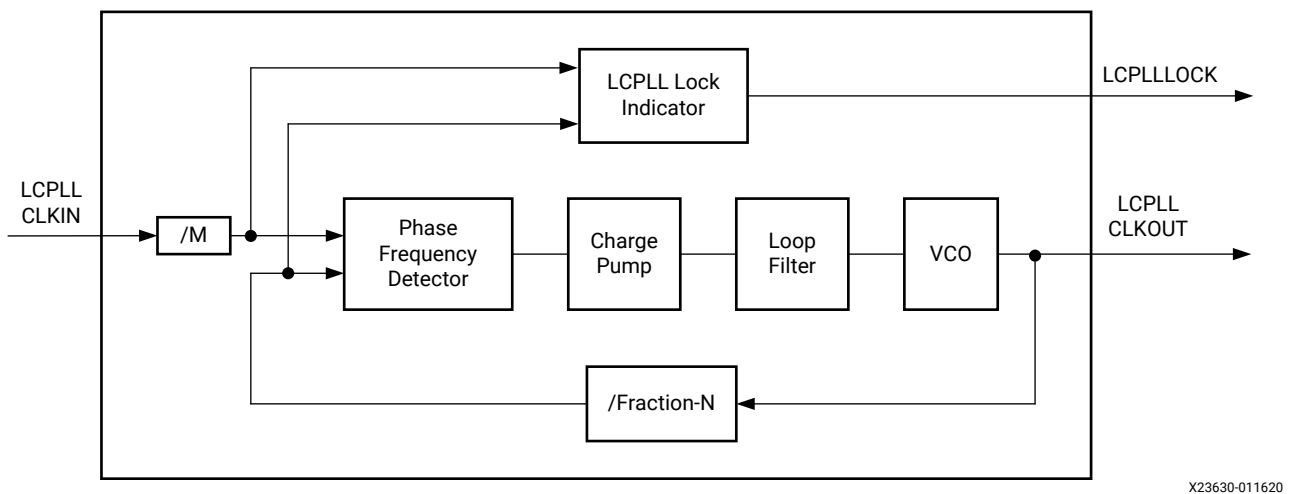
Reference Clock Selection and Distribution Attributes		
Attribute	Address	
HS0_LCPLL_IPS_PIN_EN	0x0C13	
Label	Bit Field	Description
HS0_LCPLL_IPS_PIN_EN	[27:27]	Determines whether the reference clock input to the LCPLL in block HSCLK0 is controlled by port HSCLK0_LCPLLREFCLKSEL or attribute HS0_LCPLL_IPS_REFCLK_SEL 0:HS0_LCPLL_IPS_REFCLK_SEL takes control 1:HSCLK0_LCPLLREFCLKSEL takes control
Attribute	Address	
HS0_LCPLL_IPS_REFCLK_SEL	0x0C13	
Label	Bit Field	Description
HS0_LCPLL_IPS_REFCLK_SEL	[26:24]	Control selection of reference clock input to LCPLL in block HSCLK0 1:HSCLK0_LCPLLGTREFCLK0 2:HSCLK0_LCPLLGTREFCLK1 3:HSCLK0_LCPLLNORTHREFCLK0 4:HSCLK0_LCPLLNORTHREFCLK1 5:HSCLK0_LCPLLSOUTHREFCLK0 6:HSCLK0_LCPLLSOUTHREFCLK1 7:HSCLK0_LCPLLGTGREFCLK
Attribute	Address	
HS1_LCPLL_IPS_PIN_EN	0x0F13	
Label	Bit Field	Description
HS1_LCPLL_IPS_PIN_EN	[27:27]	Determines whether the reference clock input to the LCPLL in block HSCLK1 is controlled by port HSCLK1_LCPLLREFCLKSEL or attribute HS1_LCPLL_IPS_REFCLK_SEL 0:HS1_LCPLL_IPS_REFCLK_SEL takes control 1:HSCLK1_LCPLLREFCLKSEL takes control
Attribute	Address	
HS1_LCPLL_IPS_REFCLK_SEL	0x0F13	
Label	Bit Field	Description
HS1_LCPLL_IPS_REFCLK_SEL	[26:24]	Control selection of reference clock input to LCPLL in block HSCLK1 1:HSCLK1_LCPLLGTREFCLK1 2:HSCLK1_LCPLLGTREFCLK0 3:HSCLK1_LCPLLNORTHREFCLK0 4:HSCLK1_LCPLLNORTHREFCLK1 5:HSCLK1_LCPLLSOUTHREFCLK0 6:HSCLK1_LCPLLSOUTHREFCLK1 7:HSCLK1_LCPLLGTGREFCLK

LC Tank PLL

Each GTM Quad contains two LC-based PLLs, one per HSCLK block. The LCPLL input reference clock selection is described in [Reference Clock Selection and Distribution](#). The LCPLL outputs feed the TX and RX clock dividers of each serial transceiver channel, which then controls the generator of serial and parallel clocks used by the PMA and PCS blocks.

The following figure illustrates a conceptual view of the LCPLL architecture. The input clock is divided by a factor of M before it is fed into the phase frequency detector. The feedback divider N determines the VCO multiplication ratio. The fractional-N divider is supported where the effective ratio is a combination of the N factor plus a fractional part. The LCPLL output frequency depends on the setting of LCPLLCLKOUT_RATE. When LCPLLCLKOUT_RATE is set to HALF, the output frequency is half of the VCO frequency. A lock indicator block compares the frequency of the reference clock and the VCO feedback clock to determine if a frequency lock has been achieved.

Figure 9: LCPLL Block Diagram



The LCPLL in the GTM transceiver has a nominal operating range between 9.5 GHz to 15 GHz. For additional information regarding the exact LCPLL operating range for different device speed grades, refer to the specific [Versal device data sheet](#). The Versal Adaptive SoC Transceivers Wizard chooses the appropriate LCPLL settings based on application requirements.

The following equation shows how to determine the LCPLL output frequency (GHz).

Equation 1: Determine the LCPLL Output Frequency

$$f_{PLLCLKOUT} = f_{PLLCLKIN} \times \frac{N.FractionalPart}{M}$$

The following equation shows how to determine the line rate (Gb/s).

Equation 2: Determine the Line Rate

$$f_{\text{LineRate}} = \frac{\text{Modulation} \times f_{\text{PLLCLKOUT}}}{D}$$

The following equation shows how to determine the fractional part of the feedback divider presented in [Equation 1](#).

Equation 3: Determine the Fractional Part of the Feedback Divider

$$\text{FractionalPart} = \frac{\text{SDM DATA}}{2^{\text{SDMWIDTH}}}$$

Table 16: LCPLL Divider Settings

Factor	Attribute/Port	Valid Settings
M	HSCLK[0/1]_LCPLL_CFG0	1, 2, 3, 4
N	A_HS[0/1]_LCPLLFBDIV	The valid divider range is 16–160
D	TXOUT_DIV RXOUT_DIV	1, 2, 4
Modulation	See TX Configurable Driver	4 (NRZ), 8 (PAM4)
SDMWIDTH	HSCLK[0/1]_LCPLL_LGC_CFG1	12, 16, 20, 24
SDM DATA	HSCLK[0/1]_LCPLLSDM DATA	[SDMWIDTH + 1:SDMWIDTH]: two's complement integer in range [–2, 1] [SDMWIDTH – 1:0]: fractional part in range [0, (2 ^{SDMWIDTH} – 1)]

Notes:

1. If Frac-N is enabled, line rates between 9.5 Gb/s and 58 Gb/s should have N ≤ 80 and FractionalPart between 0.1 and 0.9. Also, line rates between 76 Gb/s and 112 Gb/s should have N ≤ 21 and FractionalPart between 0.1 and 0.9.

Ports and Attributes

The following tables define the pins and attributes for the LCPLL.

Table 17: LCPLL Ports

Port	Direction	Clock Domain	Description
HSCLK[0/1]_LCPLLGTREFCLK0	Input	CLOCK	Local GTREFCLK0 to LCPLL in HSCLK0/1.
HSCLK[0/1]_LCPLLGTREFCLK1	Input	CLOCK	Local GTREFCLK1 to LCPLL in HSCLK0/1.
HSCLK[0/1]_LCPLLLOCK	Output	ASYNC	Active-High LCPLL frequency lock signal indicates that the LCPLL frequency is within a predetermined tolerance. The GTM transceiver and its clock outputs are not reliable until this condition is met.
HSCLK[0/1]_LCPLLNORTHREFCLK0	Input	CLOCK	North-bound GTREFCLK0 clock from the Quad below.
HSCLK[0/1]_LCPLLNORTHREFCLK1	Input	CLOCK	North-bound GTREFCLK1 clock from the Quad below.

Table 17: LCPLL Ports (cont'd)

Port	Direction	Clock Domain	Description
HSCLK[0/1]_LCPLLPD	Input	ASYNC	Reserved. Do not use. Use A_HS[0/1]_LCPLLPD instead to power down the LCPLL.
HSCLK[0/1]_LCPLLREFCLKSEL[2:0]	Input	ASYNC	Input to dynamically select the input reference clock to the LCPLL. Set this input to 3'b001 when only one clock source is connected to the LCPLL reference clock selection multiplexer. 3'b001: HSCLK0_LCPLLGTREFCLK0/ HSCLK1_LCPLLGTREFCLK1 3'b010: HSCLK0_LCPLLGTREFCLK1/ HSCLK1_LCPLLGTREFCLK0 3'b011: HSCLK[0/1]_LCPLLNORTHREFCLK0 3'b100: HSCLK[0/1]_LCPLLNORTHREFCLK1 3'b101: HSCLK[0/1]_LCPLLSOUTHREFCLK0 3'b110: HSCLK[0/1]_LCPLLSOUTHREFCLK1 3'b111: HSCLK[0/1]_LCPLLGTREFCLK Reset must be applied to the LCPLL after changing the reference clock input.
HSCLK[0/1]_LCPLLRESET	Input	ASYNC	Active-High signal that resets the LCPLL.
HSCLK[0/1]_LCPLLSDMDATA[25:0]	Input	ASYNC	Input to set the numerator of the fractional part of the LCPLL feedback divider. [SDM_WIDTHSEL+1:SDM_WIDTHSEL]: two's complement integer in range [-2, 1] [SDM_WIDTHSEL-1:0]: fractional part in range [0, (2^SDM_WIDTHSEL - 1)] Alternatively, this Port can be controlled through Attribute A_HS[0/1]_LCPLLSDMDATA. To enable Port control, set HS[0/1]_LCPLL_SDM_PIN_EN to 1'b1.
HSCLK[0/1]_LCPLLSDMTOGGLE	Input	ASYNC	Reserved. Set to 1'b0.
HSCLK[0/1]_LCPLLSOUTHREFCLK0	Input	CLOCK	South-bound GTREFCLK0 clock from the Quad above.
HSCLK[0/1]_LCPLLSOUTHREFCLK1	Input	CLOCK	South-bound GTREFCLK1 clock from the Quad above.

Table 18: LCPLL Attributes

LCPLL Attributes		
Attribute	Address	
A_CFG1	0x0E12	
Label	Bit Field	Description
A_HS1_LCPLLFBDIV	[23:16]	Feedback divider N setting for LCPLL in HSCLK1. Actual feedback divider value is A_HS1_LCPLLFBDIV + 2. Valid values are 14-158 (Actual divider values are 16-160).
A_HS0_LCPLLFBDIV	[7:0]	Feedback divider N setting for LCPLL in HSCLK0. Actual feedback divider value is A_HS0_LCPLLFBDIV + 2. Valid values are 14-158 (Actual divider values are 16-160).

Table 18: LCPLL Attributes (cont'd)

LCPLL Attributes		
Attribute	Address	
A_CFG2	0x0E13	
Label	Bit Field	Description
A_HS0_LCPLLDPD	[31:31]	Attribute control to power down LCPLL. Set HS0_LCPLLDPD_PIN_EN to 1'b0 to power down using this attribute. 1'b0: Power up 1'b1: Power down
A_HS0_LCPLLSDMDATA	[25:0]	Input to set the numerator of the fractional part of the LCPLL in HSCLK0 feedback divider. [SDM_WIDTHSEL+1:SDM_WIDTHSEL]: two's complement integer in range [-2, 1] [SDM_WIDTHSEL-1:0]: fractional part in range [0, (2^SDM_WIDTHSEL - 1)] To enable Attribute control, set HS0_LCPLL_SDM_PIN_EN to 1'b0 (default). To enable Port control, set HS0_LCPLL_SDM_PIN_EN to 1'b1.
Attribute	Address	
A_CFG4	0x0F11	
Label	Bit Field	Description
A_HS1_LCPLLDPD	[31:31]	Attribute control to power down LCPLL. Set HS1_LCPLLDPD_PIN_EN to 1'b0 to power down using this attribute. 1'b0: Power up 1'b1: Power down
A_HS1_LCPLLSDMDATA	[25:0]	Input to set the numerator of the fractional part of the LCPLL in HSCLK1 feedback divider. [SDM_WIDTHSEL+1:SDM_WIDTHSEL]: two's complement integer in range [-2, 1] [SDM_WIDTHSEL-1:0]: fractional part in range [0, (2^SDM_WIDTHSEL - 1)] To enable Attribute control, set HS1_LCPLL_SDM_PIN_EN to 1'b0 (default). To enable Port control, set HS1_LCPLL_SDM_PIN_EN to 1'b1.
Attribute	Address	
CH0_CHCLK_CFG3	0x0C2F	
CH1_CHCLK_CFG3	0x0D2F	
CH2_CHCLK_CFG3	0x0E2F	
CH3_CHCLK_CFG3	0x0F2F	
Label	Bit Field	Description
RXOUT_DIV	[28:26]	This attribute selects the setting for the RX serial clock divider D. 3'b000: /1 3'b001: /2

Table 18: LCPLL Attributes (cont'd)

LCPLL Attributes		
Attribute	Address	
HSCLK0_HSDIST_CFG	0x0D3F	
HSCLK1_HSDIST_CFG	0x0F3F	
Label	Bit Field	Description
HSDIST_DIV2SEL	[16:16]	Sets the LCPLLCLKOUT_RATE factor either to provide full LCPLL VCO frequency, or half of LCPLL VCO frequency at the output: 0: Full rate 1: Half rate
Attribute	Address	
HSCLK0_LCPLL_CFG0	0x0D3C	
HSCLK1_LCPLL_CFG0	0x0F3C	
Label	Bit Field	Description
LCPLL_PREDIV	[24:20]	LCPLL reference clock divider M settings. Valid settings are 1, 2, 3, and 4. 00000: 2 00001: 3 00010: 4 00100: 1
Attribute	Address	
HSCLK0_LCPLL_LGC_CFG0	0x0C3C	
HSCLK1_LCPLL_LGC_CFG0	0x0E3C	
Label	Bit Field	Description
LCPLLLOCKEN	[11:11]	Active-High signal enables the LCPLL lock detector.
Attribute	Address	
HSCLK0_LCPLL_LGC_CFG1	0x0C3D	
HSCLK1_LCPLL_LGC_CFG1	0x0E3D	
Label	Bit Field	Description
SDM_WIDTHSEL	[10:9]	This attribute sets the denominator of the fractional part of the feedback divider. 00: 24 01: 20 10: 16 11: 12
BYPASS	[7:7]	Active-Low signal enables the fractional-N divider.
Attribute	Address	
PIN_CFG0	0x0E10	
Label	Bit Field	Description
HS1_LCPLLPD_PIN_EN	[17:17]	Enable Attribute control using A_HS1_LCPLLPD. Use the default setting 1'b0.

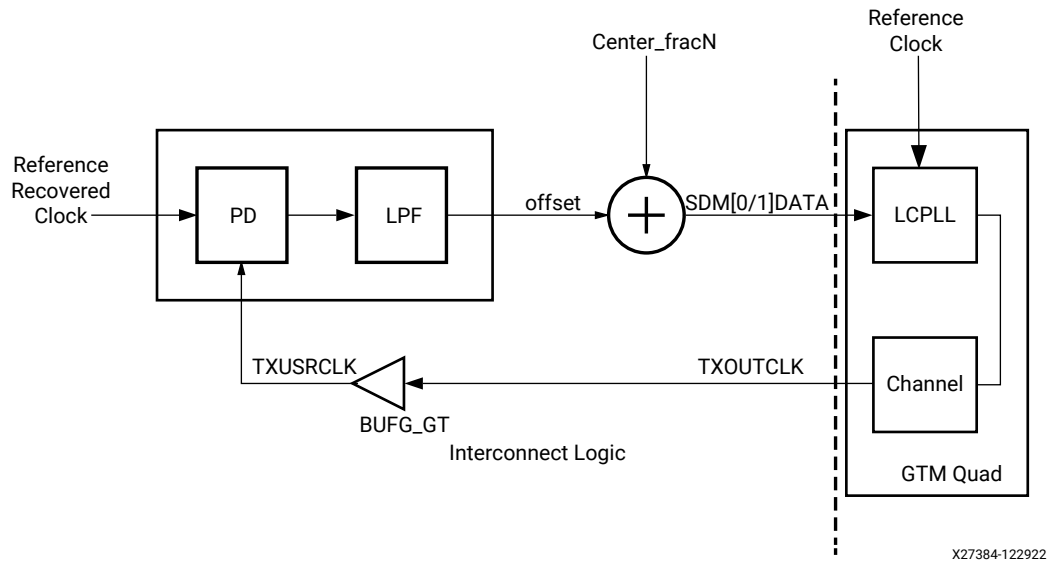
Table 18: LCPLL Attributes (cont'd)

LCPLL Attributes		
HS1_LCPLL_SDM_PIN_EN	[16:16]	1'b0: Enable Attribute control using A_HS1_LCPLLSDMDATA (default). 1'b1: Enable Port control using HSCLK1_LCPLLSDMDATA.
HS0_LCPLLPD_PIN_EN	[7:7]	Enable Attribute control using A_HS0_LCPLLPD. Use the default setting 1'b0.
HS0_LCPLL_SDM_PIN_EN	[6:6]	1'b0: Enable Attribute control using A_HS0_LCPLLSDMDATA (default). 1'b1: Enable Port control using HSCLK0_LCPLLSDMDATA.
Attribute	Address	
CH0_TX_CTRL_CFG3	0x0C37	
CH1_TX_CTRL_CFG3	0x0D37	
CH2_TX_CTRL_CFG3	0x0E37	
CH3_TX_CTRL_CFG3	0x0F37	
Label	Bit Field	Description
TXOUT_DIV	[14:13]	This attribute selects the setting for the TX serial clock divider D. 3'b000: /1 3'b001: /2 3'b010: /4

Dynamic Frac-N

Typically, SDM*DATA is statically set to create a static fractional-N divider, but it is possible to continually update the value of SDM*DATA. The following figure shows how to use the GTM transceiver when dynamic Frac-N is required. Because dynamic Frac-N operates using LCPLLs, there is no independent PPM control for the lanes connected to a particular LCPLL.

Figure 10: Dynamic Frac-N Example



The operation of the dynamic fractional divider has a user-operated strobe (**SDM*TOGGLE**) pulse that controls the **SDM*DATA** transfer from the fabric to the transceiver. The assumption here is the logic that drives **SDM*TOGGLE** and **SDM*DATA** is clocked by **SYSTEM** clock.

The following figures show the same operation with different relationships between **LCPLL** **FBCLK** and **SYSTEM** clock frequencies.

Figure 11: FBCLK is Faster than SYSTEM Clock

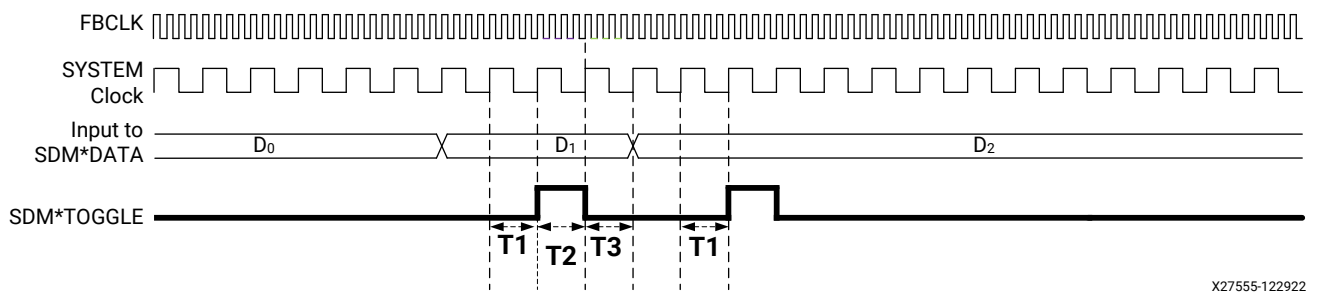
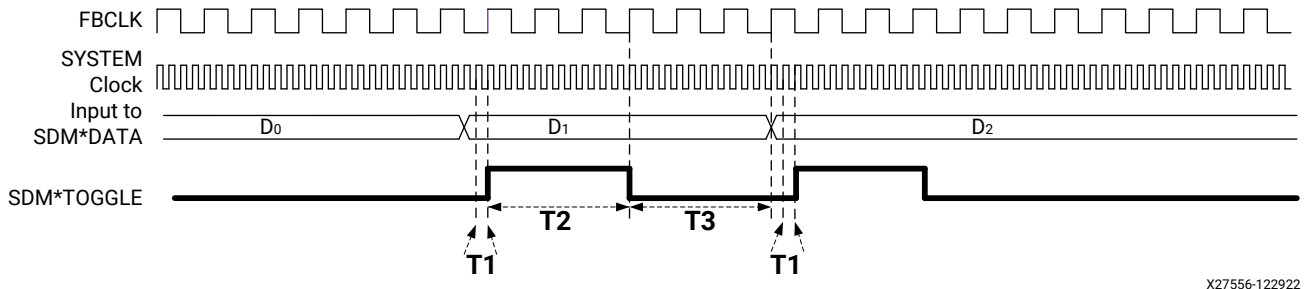


Figure 12: FBCLK is Slower than SYSTEM Clock



There are three major timing requirements marked as T1, T2, and T3 in the two figures above. These timing requirements are described in the following table.

Table 19: Timing Requirements

Time Period	Requirements	Comments
T1	* \geq * One system clock cycle	This is the least amount of time SDM*DATA must be stable and valid before SDM*TOGGLE can be asserted. Period T1 should be at least one SYSTEM clock cycle. During this period, SDM*TOGGLE must be Low.
T2	* \geq * Three FBCLK cycles	This is the least amount of time SDM*TOGGLE must be held High. Period T2 should be at least three FBCLK cycles. SDM*DATA must not change during T2.
T3	* \geq * Three FBCLK cycles	This is the least amount of time SDM*TOGGLE must be held Low. Period T3 should be at least three FBCLK cycles. SDM*DATA must not change during T3. T3 does not include T1.

Note:

1. The * symbol indicates either 0 or 1. For example, SDM*DATA can be SDM0DATA or SDM1DATA.
2. USRCLK or ABPCLK can be used as the SYSTEM clock.
3. Line rates should be below 29.0 Gb/s (NRZ) and 112.0 Gb/s (PAM4), or the frequency derivation should remain within a maximum of ± 100 ppm in both directions or 200 ppm in a single direction.

The following table shows the requirements for FBDIV when using the fractional-N divider.

Table 20: FBDIV Settings in Frac-N Mode

Frac-N Mode	Line Rate (Gb/s)	Integer Value	Fractional Value
PAM4	19 – 58	$N \leq 80$	$0.1 \leq \text{FractionalPart} \leq 0.9$
	76 – 112	$N \leq 21$	$0.1 \leq \text{FractionalPart} \leq 0.9$
NRZ	9.5 – 29	$N \leq 80$	$0.1 \leq \text{FractionalPart} \leq 0.9$

Reset and initialization

The transceiver must be initialized after device power-up and configuration before it can be used. The transmitter (TX) and receiver (RX) can be initialized independently and in parallel as shown in the following figure. Transceiver TX and RX initialization comprises three steps:

1. Initializing the associated PLL driving TX/RX
2. Initializing the associated ILO
3. Initializing the TX and RX datapaths (PMA, PCS, PROGDIV)



IMPORTANT! *The RX datapath reset signals must be executed through the Versal Adaptive SoC Transceivers Wizard. See Versal Adaptive SoC Transceivers Wizard LogiCORE IP Product Guide ([PG331](#)).*

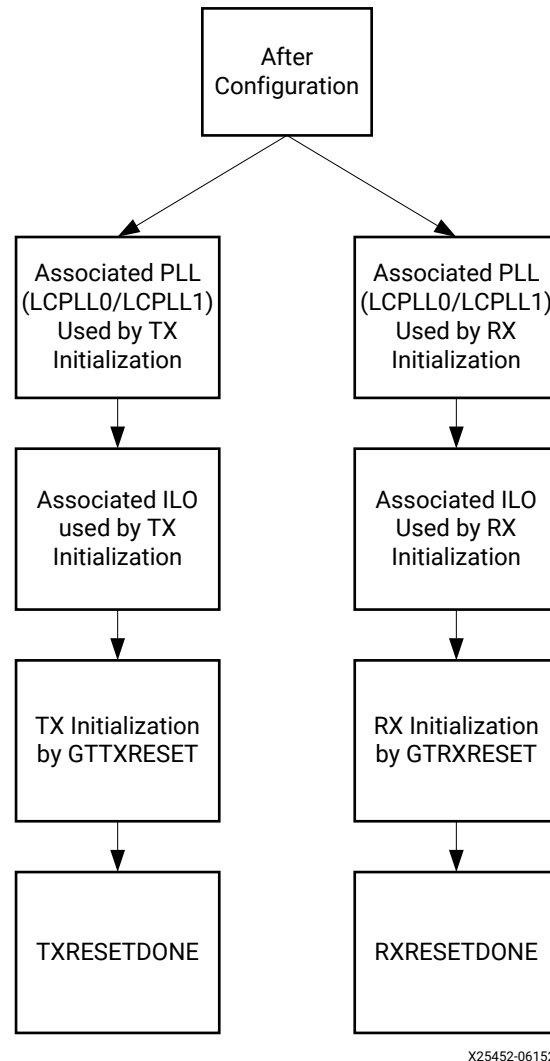
Transceiver TX and RX can receive a clock from either the nearest LCPLL or the LCPLL from the adjacent dual. The associated PLL (LCPLL0/LCPLL1) used by the TX and RX must be initialized first before ILO, TX and RX initialization. The ILO must be initialized after the associated PLL is locked. Any PLL and ILO used by the TX and RX is reset individually and its reset operation is completely independent from all TX and RX resets. The TX and RX datapaths must be initialized only after the associated PLL and ILO are locked.

For the GTM transceivers, there is a master reset controller available that can run through all reset steps described above automatically. When using the master reset, the associated PLL, ILO, and TX and RX datapaths are reset in the proper sequence by toggling the respective TX and RX master reset signal. For details on the master reset controller, refer to [Transceiver Master Reset](#).



IMPORTANT! *The reference clock to the transceiver must be available and stable prior to start of device programming or start of any manual reset sequence.*

Figure 13: Transceiver Initialization Overview



The transceiver TX and RX use a state machine to control the initialization process. They are partitioned into a few reset regions. The partition allows the reset state machine to control the reset process in a sequence that the PMA can be reset first and the PCS can be reset after the assertion of the TXUSERRDY or RXUSERRDY. It also allows the PMA, the PCS, and functional blocks inside them to be reset individually when needed during normal operation.

The transceiver offers two types of reset: initialization and component.

- **Initialization Reset:** This reset is used for complete transceiver initialization. It must be used after device power-up and configuration. During normal operation, when necessary, GTTXRESET and GTRXRESET can also be used to reinitialize the transceiver TX and RX. GTTXRESET is the initialization reset port for the transceiver TX. GTRXRESET is the initialization reset port for the transceiver RX. During initialization reset, TXRESETMODE and RXRESETMODE should be set to sequential mode. All TX PMA, TX PCS, RX PMA, and RX PCS component resets should be enabled by setting all required component bits of TXPMARESETMASK, TXPCSRESETMASK, RXPMARESETMASK, and RXPCSRESETMASK to High.
- **Component Reset:** This reset is used for special cases and specific subsection resets while the transceiver is in normal operation. The component that is required to be reset is selected by setting the associated bit within TXPMARESETMASK, TXPCSRESETMASK, RXPMARESETMASK, or RXPCSRESETMASK to High. A TX component reset is triggered by toggling the GTTXRESET port. An RX component reset is triggered by toggling the GTRXRESET port. Separate component reset ports are available. These include EYESCANRESET, RXPROGDIVRESET, RXCDRRESET, and RXPRBSCNTRESET.

All reset ports described in this section initiate the internal reset state machine when driven High. The internal reset state machines are held in the reset state until these same reset ports are driven Low. These resets are all asynchronous. The guideline for the pulse width of these asynchronous resets is one period of the reference clock, unless otherwise noted.

Note: Do not use reset ports for the purpose of power down. For details on proper power-down usage, refer to [Power Down](#).

Resetting Multiple Lanes and Quads

Resetting multiple lanes in a Quad or multiple Quads affects the power supply regulation circuit. See [Power Up/Down and Reset on Multiple Lanes](#).

Reset Modes

The transceiver resets can operate in two different modes: sequential mode and single mode.

- **Sequential mode:** The reset state machine starts with an initialization or component reset input driven High and proceeds through all states after the requested reset states in the reset state machine, as shown in [Figure 15](#) for transceiver TX or [Figure 20](#) for transceiver RX until completion. The completion of sequential mode reset flow is signaled when (TX/RX) RESETDONE transitions from Low to High.
- **Single mode:** The reset state machine only executes the requested component reset independently for a predetermined time set by its attribute. It does not process any state after the requested state, as shown in [Figure 15](#) for transceiver TX or [Figure 20](#) for transceiver RX. The requested reset can be any component reset to reset the PMA, the PCS, or functional blocks inside them. The completion of a single mode reset is signaled when (TX/RX) RESETDONE transitions from Low to High.

The GTM transceiver initialization reset must use sequential mode. All component resets can be operated in either sequential mode or single mode. The GTM transceiver uses (TX/RX) RESETMODE to select between sequential reset mode and single reset mode. The following table provides configuration details that apply to both the transceiver TX and transceiver RX. Reset modes have no impact on LCPLL resets. During normal operation, the transceiver TX or transceiver RX can be reset by applications in either sequential mode or single mode, which provides flexibility to reset a portion of the GTM transceiver. When using either sequential mode or single mode, (TX/RX) RESETMODE must be set to the desired value 50 ns before the assertions of any reset.

Table 21: Transceiver Reset Modes Operation

Operation Mode	(TX/RX) RESETMODE
Sequential Mode	2'b00
Single Mode	2'b11

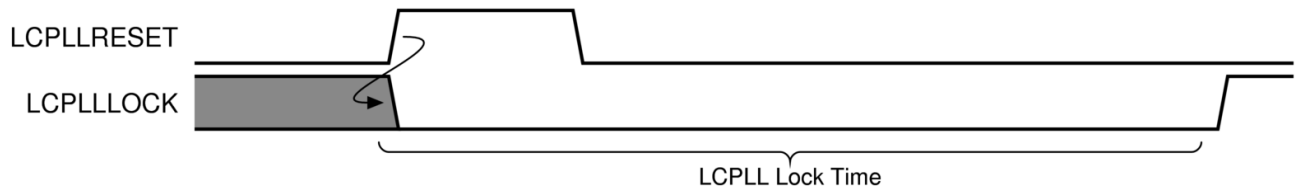
Table 22: Transceiver Reset Modes Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_RXRESETMODE[1:0]	In	Async	Reset mode port for RX. 2'b00: Sequential mode (recommended). 2'b01: Reserved. 2'b10: Reserved. 2'b11: Single mode.
CH[0/1/2/3]_TXRESETMODE[1:0]	In	Async	Reset mode port for TX. 2'b00: Sequential mode (recommended). 2'b01: Reserved. 2'b10: Reserved. 2'b11: Single mode.

LCPLL Reset

The LCPLL must be reset before it can be used. Each transceiver Quad has dedicated reset ports for each of the two respective LCPLLs. As shown in the figure, LCPLLRESET is an input that resets LCPLL. LCPLLLOCK is an output that indicates the reset process is done. The guideline for this asynchronous LCPLLRESET pulse width is one period of the reference clock. After an LCPLLRESET pulse, the internal reset controller generates an internal LCPLL reset followed by an internal SDM reset. The time required for LCPLL to lock is affected by a few factors, such as bandwidth setting and clock frequency.

Figure 14: LCPLL Reset Timing Diagram



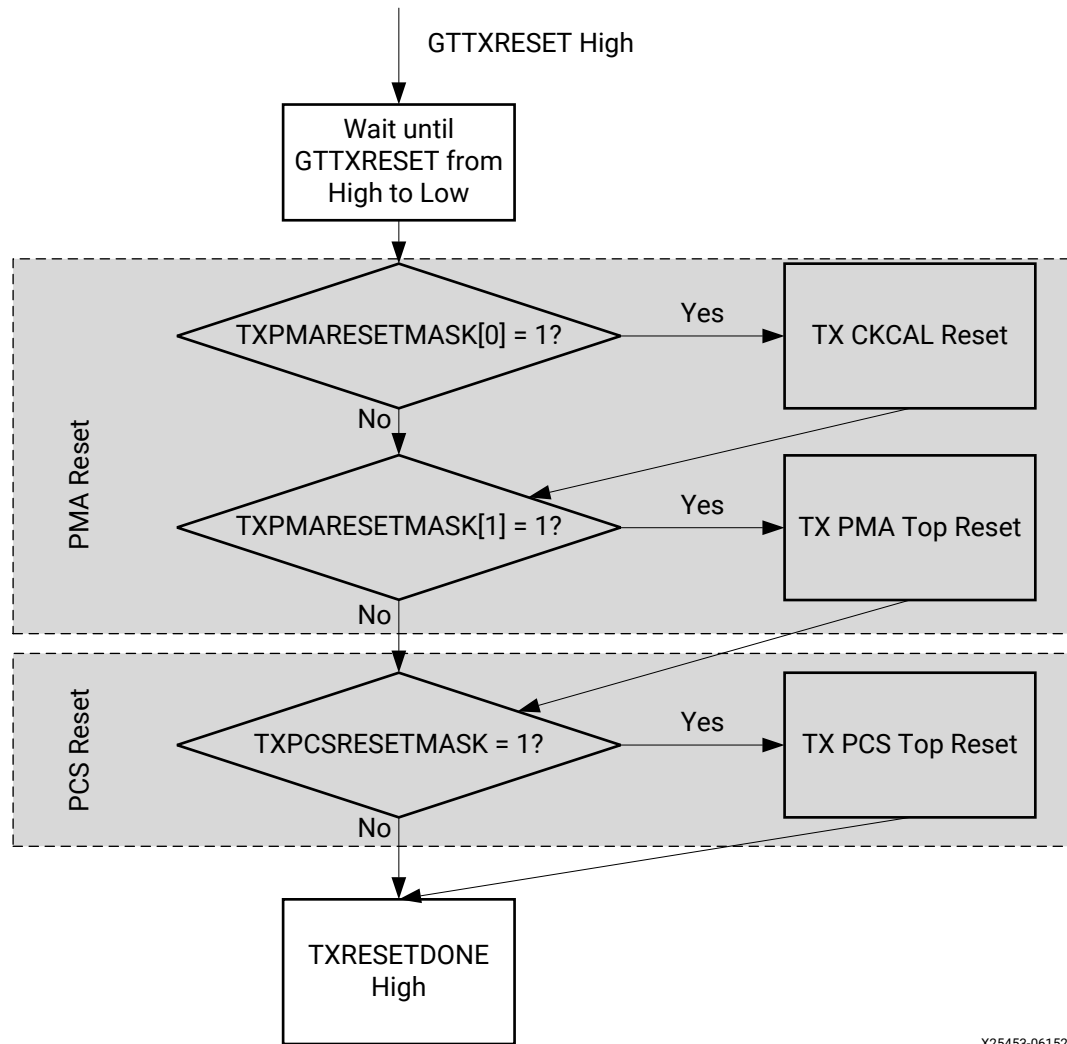
TX Initialization and Reset

The transceiver TX uses a reset state machine to control the reset process. The transceiver TX is partitioned into two reset regions, TX PMA and TX PCS. The partition allows TX initialization and reset to be operated only in sequential mode, as shown in the figure below.

The initializing TX must use GTTXRESET in sequential mode. Activating the GTTXRESET input can automatically trigger a full asynchronous TX reset. The reset state machine executes the reset sequence, as shown in the figure below, covering the whole TX PMA, and TX PCS. During normal operation, when needed, sequential mode allows you to reset the TX from activating TXPMARESET and continue the reset state machine until TXRESETDONE transitions from Low to High.

The TX reset state machine does not reset the PCS until TXUSERRDY is detected High. Drive TXUSERRDY High after all clocks used by the application including TXUSRCLK are shown as stable.

Figure 15: Transceiver TX Reset State Machine Sequence



X25453-061521

Transceiver TX Reset in Response to Completion of Configuration

The TX reset sequence shown in [TX Initialization and Reset](#) is not automatically started to follow global GSR. It must meet these conditions:

1. TXRESETMODE must be set to sequential mode.
2. GTTXRESET must be used.
3. All TXPMARESETMASK, and TXPCSRESETMASK bits should be set to High.
4. GTTXRESET cannot be driven Low until the associated PLL and ILO are locked.

5. Ensure that GTPOWERGOOD is High before releasing LCPLLRESET, ILORESET, and GTTXRESET.

If the reset mode is defaulted to single mode, then you must:

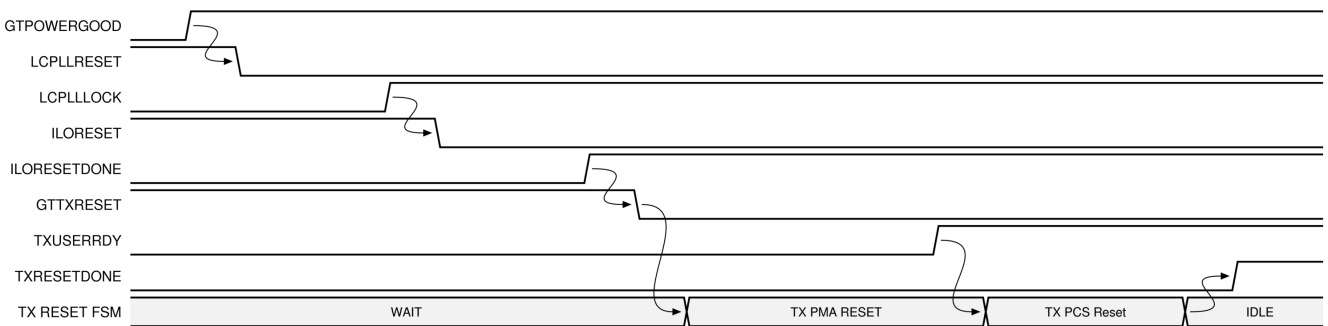
1. Wait another 300–500 ns.
2. Assert LCPLLRESET, ILORESET, and GTTXRESET following the reset sequence described in the following figure.

Alternatively, the master reset controller can be used to drive the PLL and TX reset in sequence automatically. Details can be found in [Transceiver Master Reset](#).



RECOMMENDED: Use the associated PLLLOCK from the LCPLL to release GTTXRESET from High to Low as shown in the figure. The TX reset state machine waits when GTTXRESET is detected High and starts the reset sequence until GTTXRESET is released Low.

Figure 16: Transmitter Initialization after Configuration

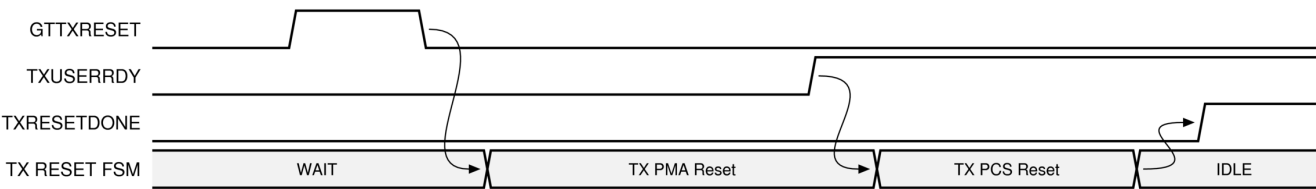


Transceiver TX Reset in Response to GTTXRESET Pulse in Full Sequential Reset

The GTM transceiver allows you to reset the entire TX completely at any time by sending GTTXRESET an active-High pulse. These conditions must be met when using GTTXRESET:

1. TXRESETMODE must be set to sequential mode.
2. All TXPMARESETMASK, and TXPCSRESETMASK bits should be held High during the reset sequence before TXRESETDONE is detected High.
3. The associated PLL must indicate locked.
4. The guideline for this asynchronous GTTXRESET pulse width is one period of the reference clock.

Figure 17: Transmitter Reset after GTTXRESET Pulse in Full Sequential Reset



Transceiver TX Component Reset

TX PMA and TX PCS can be reset individually. Component reset is enabled by setting the appropriate TXPMARESETMASK and TXPCSRESETMASK bits along with TXRESETMODE and then toggling GTTXRESET.

Driving GTTXRESET from High to Low starts the component reset process. All TXPMARESETMASK and TXPCSRESETMASK bits along with TXRESETMODE must be held constant during the reset process.

When TXRESETMODE is set to sequential mode, the internal resets are toggled in sequence depending on TXPMARESETMASK and TXPCSRESETMASK selection. When TXRESETMODE is set to single mode, the internal resets are toggled simultaneously depending on TXPMARESETMASK and TXPCSRESETMASK selection.

In sequential mode, if the TX PCS is to be reset, TXUSERRDY must assert to High prior to the internal PCS reset signal being released allowing TX reset to be completed.

Figure 18: Transmitter Reset after GTTXRESET Pulse in Component Sequential Mode

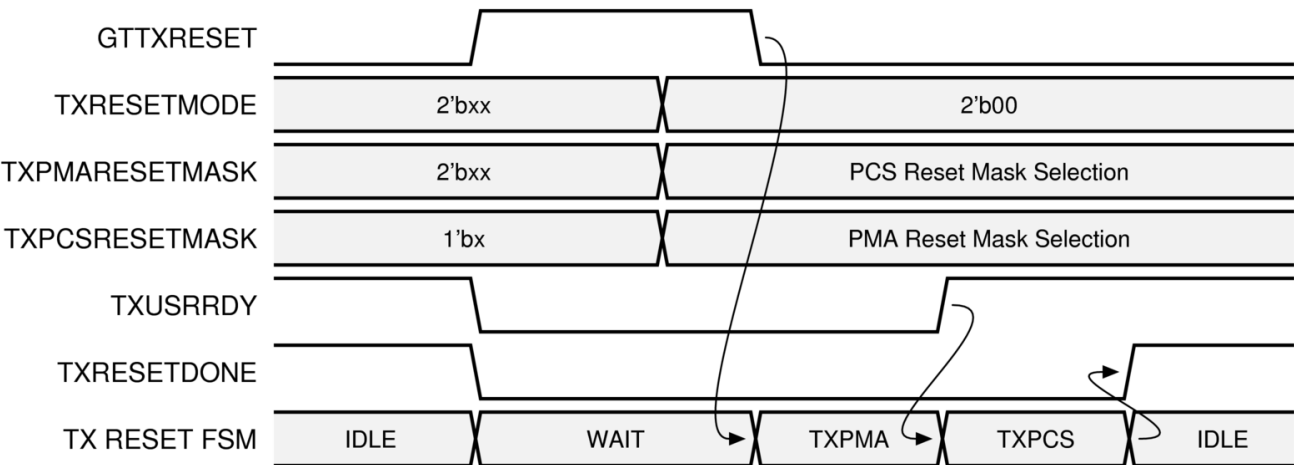
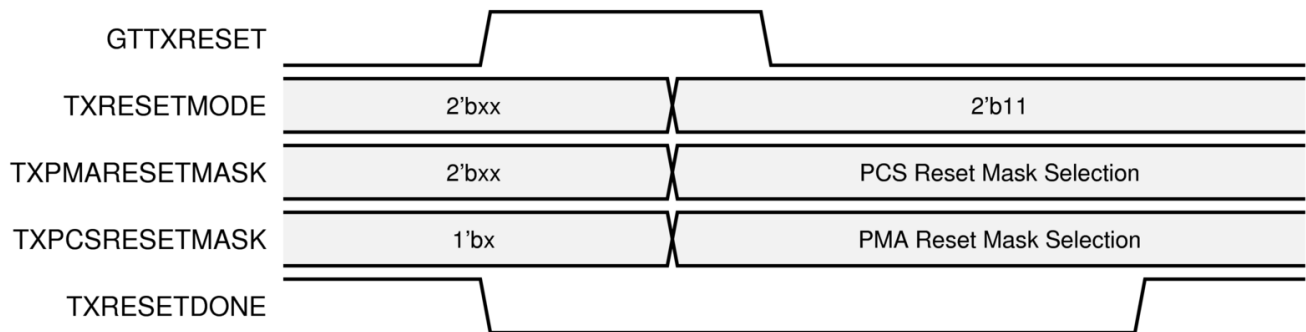


Figure 19: Transmitter Reset after GTTXRESET Pulse in Component Single Mode



The following table lists the recommended resets for common situations.

Table 23: Recommended Transmitter Resets for Common Situations

Situation	Components to be Reset	Recommended TX Reset Setting		
		TXRESETMODE	TXPMARESETMASK	TXPCSRESETMASK
After power up and configuration	LCPLL, Entire TX	2'b00	2'b11	1'b1
After turning on a reference clock to the LCPLL being used	LCPLL, Entire TX	2'b00	2'b11	1'b1
After changing the reference clock to the LCPLL being used	LCPLL, Entire TX	2'b00	2'b11	1'b1
After assertion/deassertion of LCPLLPD for the PLL being used	LCPLL, Entire TX	2'b00	2'b11	1'b1
After assertion/deassertion of TXPD[1:0]	Entire TX	2'b00	2'b11	1'b1
TX rate change	TX PMA and TX PCS	2'b00	2'b10	1'b1
TX parallel clock source reset	TX PCS	2'b00/2'b11	2'b00	1'b1

After Power-up and Configuration

The PLL being used and the entire TX require a reset after configuration. See [Transceiver TX Reset in Response to Completion of Configuration](#).

After Turning on a Reference Clock to the LCPLL Being Used

If the reference clock(s) changes or the transceiver(s) are powered up after configuration, perform a full TX sequential reset after the PLL fully completes its reset procedure.

After Changing the Reference Clock to the LCPLL Being Used

Whenever the reference clock input to the PLL is changed, the PLL must be reset afterward to ensure that it locks to the new frequency. Perform a full TX sequential reset after the PLL fully completes its reset procedure.

After Assertion/Deassertion of LCPLLPD for the PLL Being Used

When the LCPLL being used goes back to normal operation after power down, the PLL must be reset. Perform a full TX sequential reset after the PLL fully completes its reset procedure.

After Assertion/Deassertion of TXPD[1:0]

After the TXPD signal is deasserted, perform a full TX sequential reset.

TX Rate Change

When a rate change is performed using the TXRATE port, the required reset sequence is performed automatically. When TXRESETDONE is asserted, it indicates that both a rate change and the necessary reset sequence have been applied and completed.

TX Parallel Clock Source Reset

The clocks driving TXUSRCLK must be stable for correct operation. Perform a TX PCS reset after the clock source re-locks.

RX Initialization and Reset

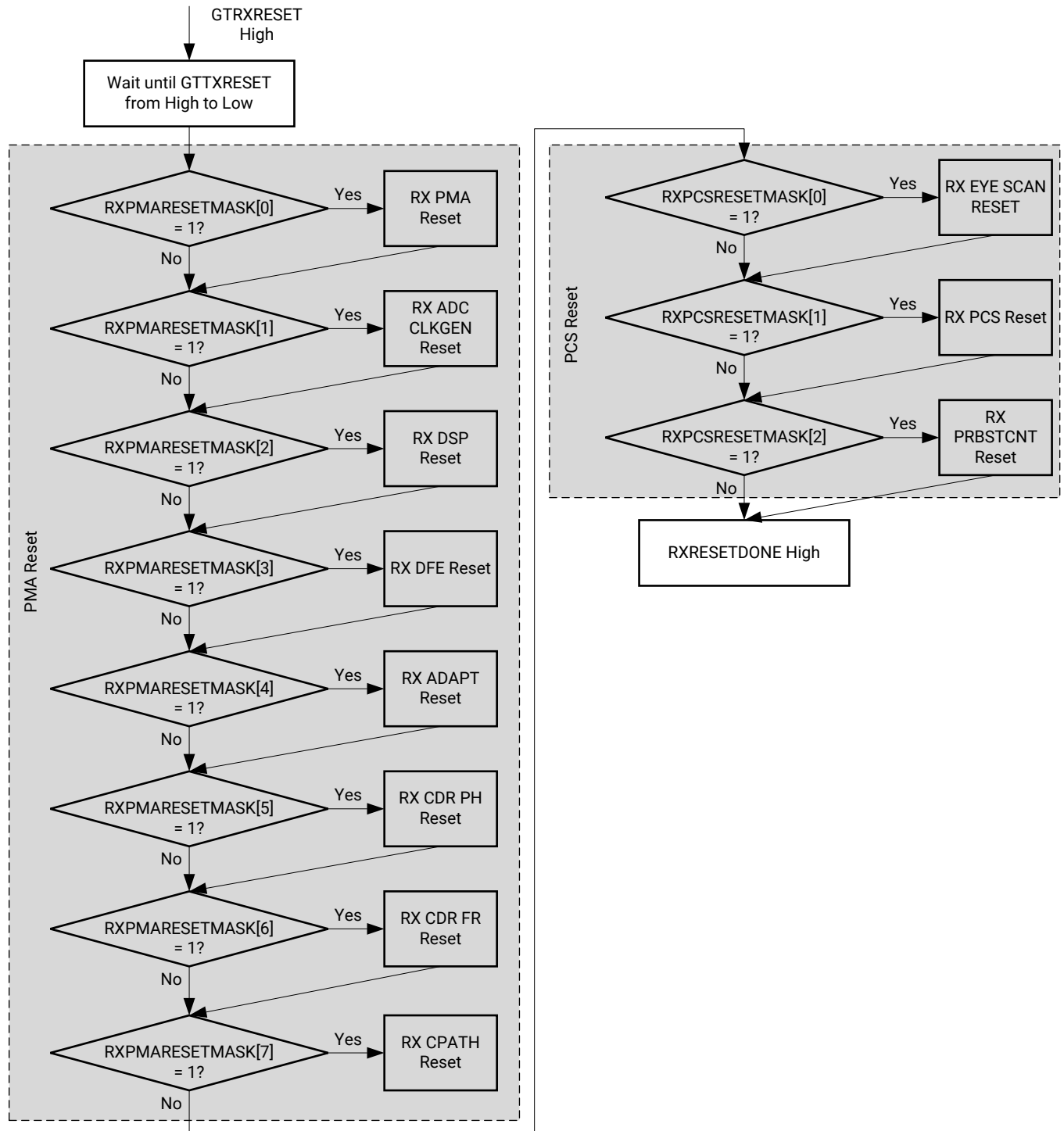
The transceiver RX uses a reset state machine to control the reset process. Due to its complexity, the transceiver RX is partitioned into more reset regions than the transceiver TX. The partition allows RX initialization and reset to be operated in either sequential mode, as shown in the following figure, or single mode:

1. **RX in Sequential Mode:** To initialize the transceiver RX, RXRESETMODE must be set to sequential mode. The RX components that are required to be reset are determined by setting the appropriate RXPMARESETMASK and RXPCSRESETMASK bits to High. The reset sequence is then triggered by toggling GTRXRESET and then internal component resets are triggered sequentially. The reset state machine executes the reset sequence as shown in the following figure, covering the entire RX PMA and RX PCS. During normal operation, the reset state machine runs until RXRESETDONE transitions from Low to High.

2. RX in Single Mode: When the transceiver RX is in single mode, RXRESETMODE must be set to single mode. The RX components that are required to be reset are determined by setting the appropriate RXPMARESETMASK and RXPCSRESETMASK bits to High. The reset sequence is then triggered by toggling GTRXRESET and the internal component resets are triggered simultaneously. In addition, EYESCANRESET, RXCDRRESET, RXPRBSCNTRESET, and RXPROGDIVRESET pins are available to reset those components directly in single mode.

In either sequential mode or single mode, the RX reset state machine does not reset the PCS until RXUSERRDY goes High. Drive RXUSERRDY High after all clocks used by the application, including RXUSRCLK, are shown to be stable.

Figure 20: Transceiver RX Reset State Machine Sequence



X25455-061521

Transceiver RX Reset in Response to Completion of Configuration

The RX reset sequence shown in [RX Initialization and Reset](#) is not automatically started to follow the global GSR.

These conditions must be met:

1. RXRESETMODE must be set to use the sequential mode.
2. GTRXRESET must be used.
3. All RXPMARESETMASK and RXPCSRESETMASK bits should be set to High.
4. GTRXRESET cannot be driven Low until the associated PLL and ILO are locked.
5. Ensure that GTPOWERGOOD is High before releasing LCPLLRESET, ILORESET, and GTRXRESET.

If the reset mode is defaulted to single mode, then you must:

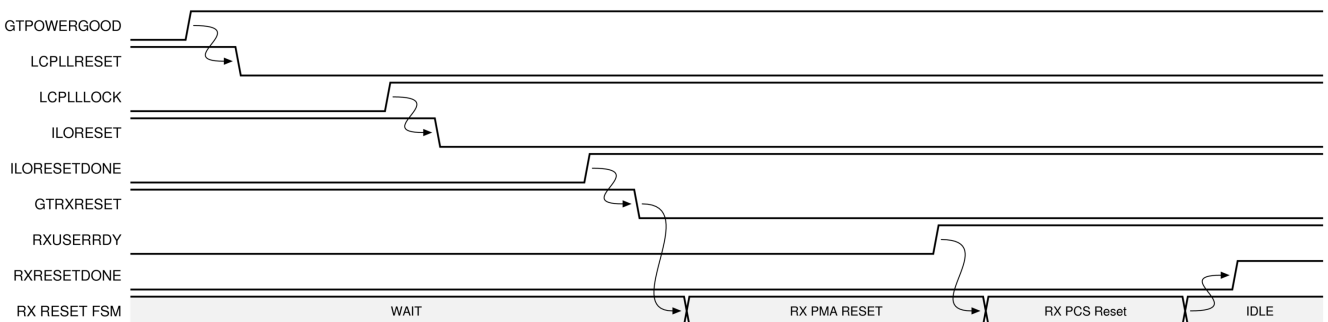
1. Change the reset mode to Sequential mode.
2. All RXPMARESETMASK and RXPCSRESETMASK bits should be changed to High.
3. Wait another 300–500 ns.
4. Assert LCPLLRESET, ILORESET, and GTRXRESET following the reset sequence described in the figure below.

Alternatively, the master reset controller can be used to drive the PLL and RX reset in sequence automatically. Details can be found in [Transceiver Master Reset](#).



RECOMMENDED: Use the associated PLLLOCK from the LCPLL to release ILORESET from High to Low as shown in the following figure, then use ILORESETDONE to release GTRXRESET from High to Low. The RX reset state machine waits when GTRXRESET is detected High and starts the reset sequence until GTRXRESET is released Low.

Figure 21: Receiver Initialization after Configuration

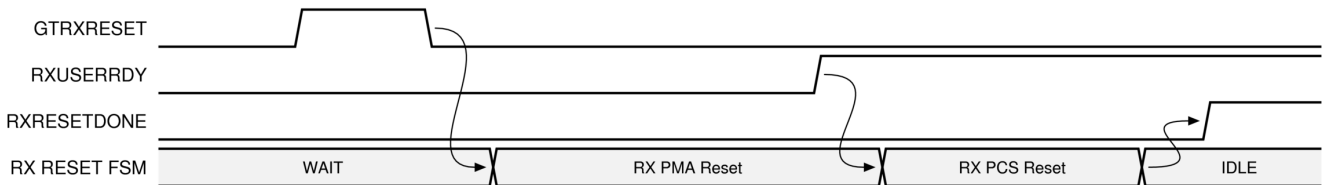


Transceiver RX Reset in Response to GTRXRESET Pulse in Full Sequential Mode

The transceiver allows you to reset the entire RX at any time by sending GTRXRESET an active-High pulse. These conditions must be met when using GTRXRESET:

1. RXRESETMODE must be set to use sequential mode.
2. All RXPMARESETMASK and RXPCSRESETMASK bits should be held to High during the reset sequence before RXRESETDONE is detected High.
3. The associated PLL must indicate locked.
4. The guideline for this asynchronous GTRXRESET pulse width is one period of the reference clock.

Figure 22: Receiver Reset after GTRXRESET Pulse in Full Sequential Reset



Transceiver RX Component Reset

Transceiver RX component resets can be reset individually in either sequential mode or single mode. They are primarily used for special cases. These resets are needed when only a specific subsection needs to be reset.

Driving GTRXRESET from High to Low starts the component reset process. All RXPMARESETMASK and RXPCSRESETMASK bits along with RXRESETMODE must be held constant during the reset process.

When RXRESETMODE is set to sequential mode, the internal resets are toggled in sequence depending on the RXPMARESETMASK and RXPCSRESETMASK selection. When RXRESETMODE is set to single mode, the internal resets are toggled simultaneously depending on the RXPMARESETMASK and RXPCSRESETMASK selection.

In sequential mode, if the RX PCS is to be reset, RXUSERRDY must toggle High prior to the internal PCS reset signal being released, allowing RX reset to be completed.

Direct single reset ports EYESCANRESET, RXCDRRESET, RXPRBSCNTRESET, and RXPROGDIVRESET are available to perform single resets of the respective RX components. When direct single reset ports are toggled, a single reset is performed regardless of RXPMARESETMASK, RXPCSRESETMASK, and RXRESETMODE selection. These ports must be held Low during any sequential or single resets driven by GTRXRESET.

The following table lists the recommended receiver resets for common situations.

Table 24: Recommended Receiver Resets for Common Situations

Situation	Components to be Reset	Recommended RX Reset Setting		
		RXRESETMODE	RXPMARESETMASK	RXPCSRESETMASK
After power up and configuration	LCPLL, ILO, Entire RX	2'b00	8'b11111111	3'b111
After turning on a reference clock to the LCPLL/RPLL being used	LCPLL, ILO, Entire RX	2'b00	8'b11111111	3'b111
After changing the reference clock to the LCPLL/RPLL being used	LCPLL, ILO, Entire RX	2'b00	8'b11111111	3'b111
After assertion/deassertion of LCPLLPD or RPLLPD for the PLL being used	LCPLL, ILO, Entire RX	2'b00	8'b11111111	3'b111
After assertion/deassertion of RXPD[1:0]	Entire RX	2'b00	8'b11111111	3'b111
RX rate change	RX PMA and RX PCS	2'b00	8'b11111111	3'b111
RX parallel clock source reset	RX PCS	2'b00	8'b00000000	3'b111
After remote power up	Entire RX	2'b00	8'b11111111	3'b111
Electrical Idle	Entire RX	2'b00	8'b11111111	3'b111
After connecting RXN/RXP	Entire RX	2'b00	8'b11111111	3'b111
After recovered clock becomes stable	RX PCS	2'b11	8'b00000000	3'b100
After RX buffer status error	RX PCS	2'b11	8'b00000000	3'b100
After changing channel bonding mode in real time	RX PCS	2'b11	8'b00000000	3'b100
After PRBS error	PRBS Error Counter	2'b11	8'b00000000	3'b010
Eye Scan reset only	Eye Scan	2'b11	8'b00000000	3'b001

Notes:

1. When reset mode is 2'b00 and PCS/PMA is set to 1, assert the reset through the bridge IP. Refer to *Versal Adaptive SoC Transceivers Wizard LogiCORE IP Product Guide* (PG331).

After Power-up and Configuration

The PLL being used and the entire RX require a reset after configuration. See [Transceiver RX Reset in Response to Completion of Configuration](#).

After Turning on a Reference Clock to the LCPLL Being Used

If the reference clock(s) changes or transceiver(s) are powered up after configuration, perform a full RX sequential reset after the PLL fully completes its reset procedure.

After Changing the Reference Clock to the LCPLL Being Used

Whenever the reference clock input to the PLL is changed, the PLL must be reset afterward to ensure that it locks to the new frequency. Perform a full RX sequential reset after the PLL fully completes its reset procedure.

After Assertion/Deassertion of LCPLLPD for the PLL Being Used

When the LCPLL being used goes back to normal operation after power down, the PLL must be reset. Perform a full RX sequential reset after the PLL fully completes its reset procedure.

After Assertion/Deassertion of RXPD[1:0]

After the RXPD signal is deasserted, perform a full RX sequential reset.

RX Rate Change

When a rate change is performed using the RXRATE port, the required reset sequence is performed automatically. When RXRESETDONE is asserted, it indicates that both a rate change and then a necessary reset sequence have been applied and completed.

RX Parallel Clock Source Reset

The clocks driving RXUSRCLK must be stable for correct operation. Perform an RX PCS reset after the clock source re-locks.

After Remote Power-Up

If the source of incoming data is powered up after the transceiver that is receiving its data has begun operating, a full RX sequential reset must be performed to ensure a clean lock to the incoming data.

After Connecting RXN/RXP

When the RX data to the transceiver comes from a connector that can be plugged in and unplugged, a full RX sequential reset must be performed when the data source is plugged in to ensure that it can lock to incoming data.

After Recovered Clock Becomes Stable

Depending on the design of the clocking scheme, it is possible for the RX reset sequence to be completed before the CDR is locked to the incoming data. In this case, the recovered clock might not be stable when RXRESETDONE is asserted. When the RX PCS is used, a single mode reset targeting the RX PCS must be triggered after the recovered clock becomes stable.

Refer to the [Versal device data sheets](#) for successful CDR lock-to-data criteria.

After an RX Buffer Status Error

After an RX FIFO buffer overflow or underflow, a single mode reset targeting the RX PCS must be triggered to ensure correct behavior.

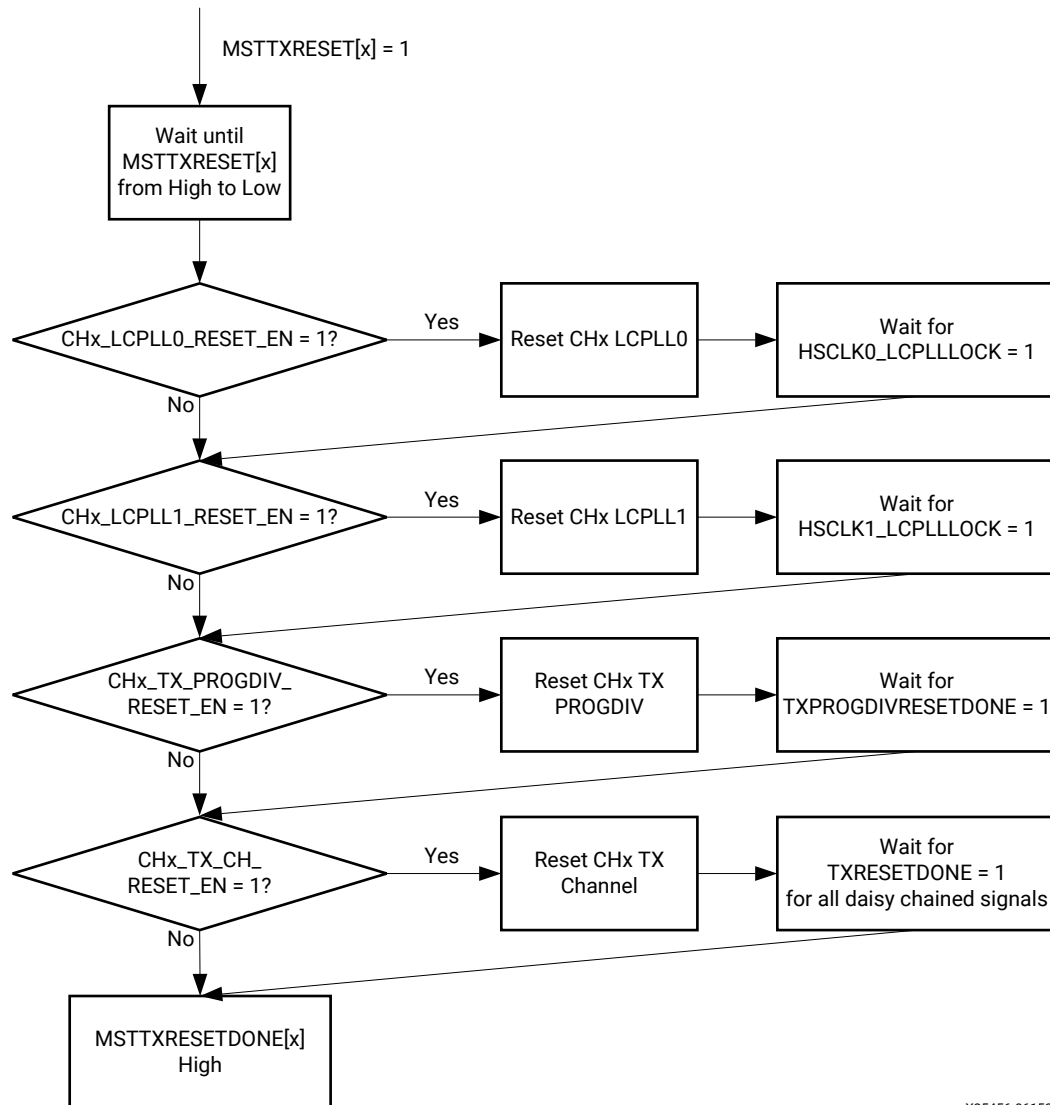
After a PRBS Error

PRBSCNTRESET is asserted to reset the PRBS error counter.

Transceiver Master Reset

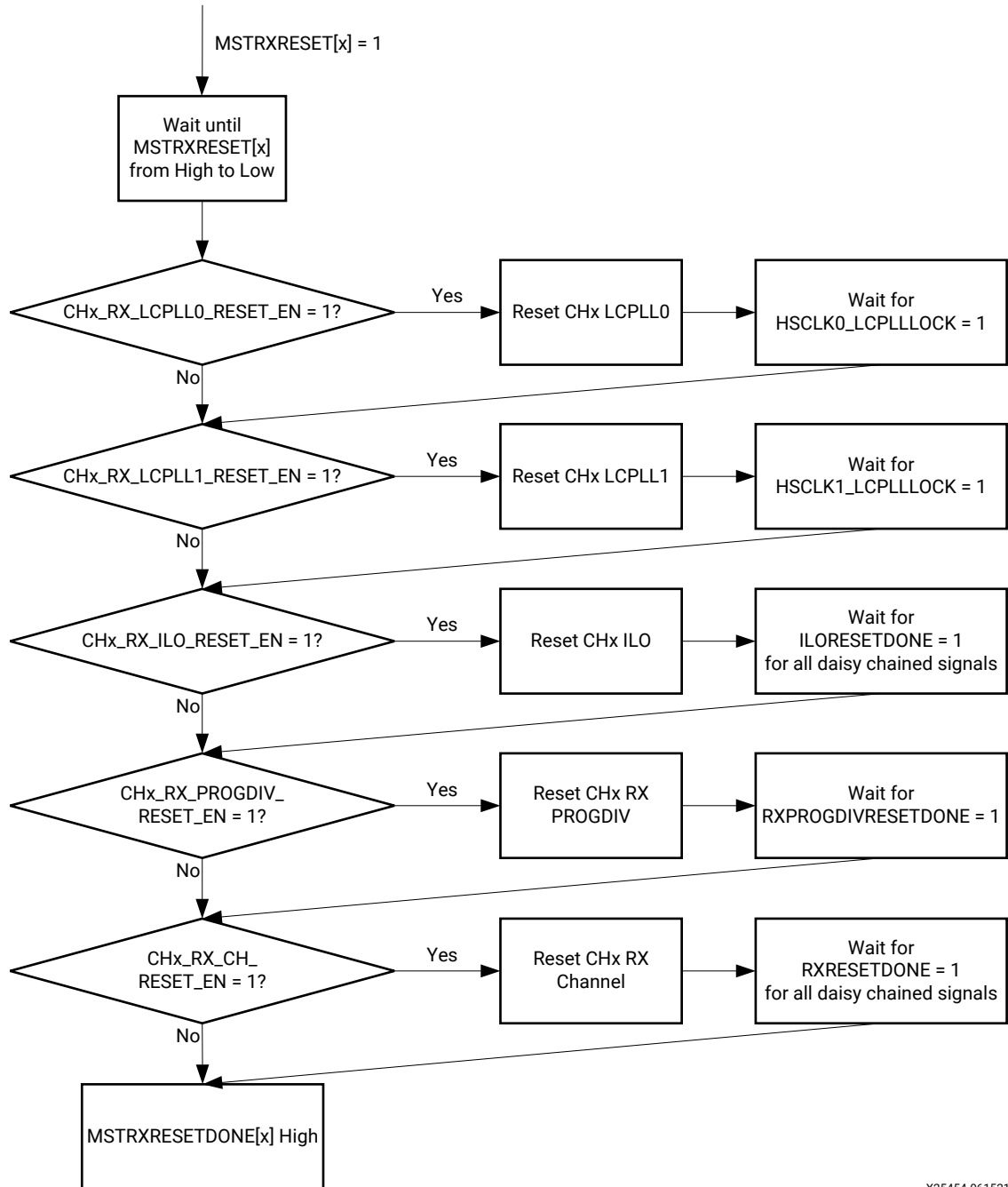
A master reset controller is available in GTM transceivers. The master reset controller automatically steps through the reset of the LCPLL, ILO, TX programmable divider, RX programmable divider, TX channel, and RX channel. The master reset controller state machine operates as shown in [Figure 23](#) for the TX and [Figure 24](#) for the RX.

Figure 23: Transceiver TX Master Reset State Machine Sequence



X25456-061521

Figure 24: Transceiver RX Master Reset State Machine Sequence



X25454-061521

Each channel contains a master reset controller port for the given channel. In multi-lane protocols, the master reset signal should be toggled individually on all used lanes. However, **ILORESETDONE** and **TX/RXRESETDONE** signals can be daisy-chained to ensure that the previous step is completed on all lanes before the master reset controller proceeds.

Transceiver Master TX Reset in Response to Completion of Configuration

The TX reset sequence shown in [Figure 15](#) is not automatically started to follow the global GSR.

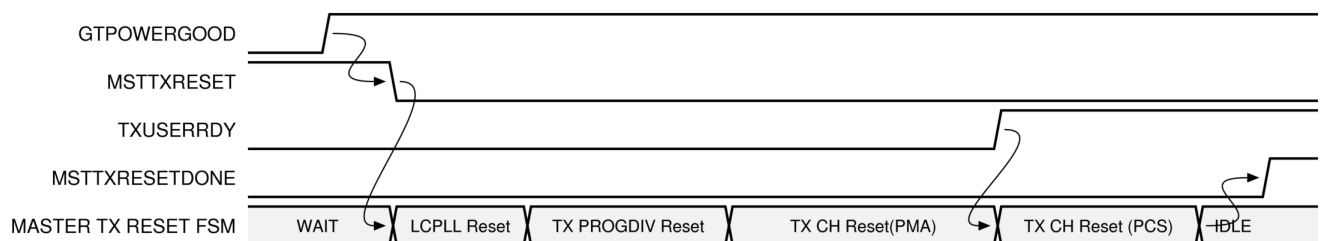
These conditions must be met:

1. TXRESETMODE must be set to use the sequential mode.
2. All TXPMARESETMASK and TXPCSRESETMASK bits should be set to High.
3. Set the appropriate MST_RESET_CFG attribute bits based on PLL selection, PROGDIV selection, and channel usage.
4. MSTTXRESET is to be used.
5. Ensure LCPLLRESET, TXPROGDIVRESET, and GTTXRESET are held Low during the master reset process.
6. Ensure that GTPOWERGOOD is High before releasing MSTTXRESET.

If the reset mode is defaulted to single mode, then you must:

1. Change reset mode to Sequential mode.
2. All TXPMARESETMASK and TXPCSRESETMASK bits should be changed to High.
3. Wait another 300–500 ns.
4. Release MSTTXRESET.

Figure 25: Transmitter Initialization after Configuration Using Master Reset Controller



Transceiver Master RX Reset in Response to Completion of Configuration

The RX reset sequence shown in the following figure is not automatically started to follow the global GSR.

These conditions must be met:

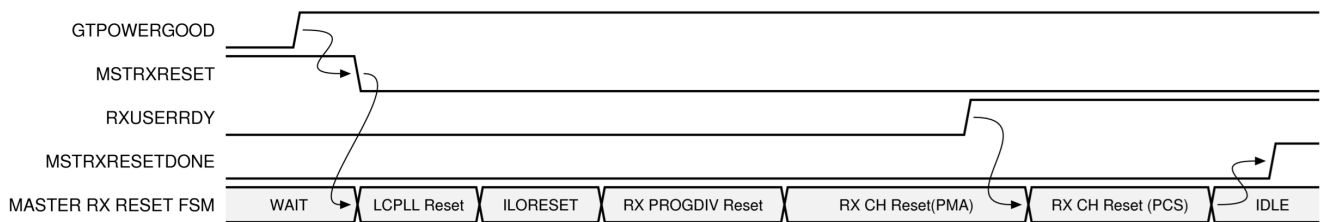
1. RXRESETMODE must be set to use the sequential mode.

2. All RXPMARESETMASK and RXPCSRESETMASK bits should be set to High.
3. Set the appropriate MST_RESET_CFG attribute bits based on PLL selection, PROGDIV selection, and channel usage.
4. MSTRXRESET is to be used.
5. Ensure LCPLLRESET, ILORESET, RXPROGDIVRESET, and GTRXRESET are held Low during the master reset process.
6. Ensure that GTPOWERGOOD is High before releasing MSTRXRESET.

If the reset mode is defaulted to single mode, then you must:

1. Change reset mode to Sequential mode.
2. All RXPMARESETMASK and RXPCSRESETMASK bits should be changed to High.
3. Wait another 300–500 ns.
4. Release MSTRXRESET.

Figure 26: Receiver Initialization after Configuration Using Master Reset Controller



Quad Sharing with Multiple IP Cores

When multiple IP cores are located within the same Quad, there are limitations on how the reset can be performed. When an IP core asserts the master reset, the corresponding PLL that it uses also goes through reset. If any other IP within the same Quad shares the same PLL, its operation will be affected. In this situation, any IP that shares the same PLL within the same Quad must perform reset at the same time.

Ports and Attributes

The following table lists ports required by the TX and RX initialization and reset process.

Table 25: Reset and Initialization Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_EYESCANRESET	Input	ASYNC	This active-High port resets the eye scan block.

Table 25: Reset and Initialization Ports (cont'd)

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_GTRXRESET	Input	ASYNC	This port is driven High and then deasserted to start a RX reset sequence. The components to be reset are determined by RXPMARESETMASK and RXPCSRESETMASK, which are controlled by the Wizard. In sequential mode, the resets are performed sequentially. In single mode, the resets are performed simultaneously.
CH[0/1/2/3]_GTTXRESET	Input	ASYNC	This port is driven High and then deasserted to start a TX reset sequence. The components to be reset are determined by TXPMARESETMASK and TXPCSRESETMASK, which are controlled by the Wizard. In sequential mode, the resets are performed sequentially. In single mode, the resets are performed simultaneously.
CH[0/1/2/3]_ILORESET	Input	ASYNC	Active-High signal that resets the ILO.
CH[0/1/2/3]_ILORESETDONE	Output	ASYNC	This active-High signal indicates the GTM transceiver ILO has finished reset and is ready for use.
CH[0/1/2/3]_ILORESETMASK	Input	ASYNC	ILO resetmask selection.
HSCLK[0/1]_LCPLLLOCK	Output	ASYNC	Active-High LCPLL frequency lock signal indicates that the LCPLL frequency is within a predetermined tolerance. The GTM transceiver and its clock outputs are not reliable until this condition is met.
HSCLK[0/1]_LCPLLRESET	Input	ASYNC	Active-High signal that resets the LCPLL.
MSTRXRESET[3:0]	Input	ASYNC	This port is driven High and then deasserted to start an RX master reset sequence. Bit 3: CH3 RX master reset Bit 2: CH2 RX master reset Bit 1: CH1 RX master reset Bit 0: CH0 RX master reset
MSTRXRESETDONE[3:0]	Output	ASYNC	This active-High signal indicates the transceiver RX master has finished reset and is ready for use. Bit 3: CH3 RX master reset done Bit 2: CH2 RX master reset done Bit 1: CH1 RX master reset done Bit 0: CH0 RX master reset done
MSTTXRESET[3:0]	Input	ASYNC	This port is driven High and then deasserted to start a TX master reset sequence. Bit 3: CH3 TX master reset Bit 2: CH2 TX master reset Bit 1: CH1 TX master reset Bit 0: CH0 TX master reset

Table 25: Reset and Initialization Ports (cont'd)

Port	Direction	Clock Domain	Description
MSTTXRESETDONE[3:0]	Output	ASYNC	This active-High signal indicates the transceiver TX master has finished reset and is ready for use. Bit 3: CH3 TX master reset done Bit 2: CH2 TX master reset done Bit 1: CH1 TX master reset done Bit 0: CH0 TX master reset done
CH[0/1/2/3]_RESETEXCEPTION	Output	ASYNC	Indicates there is an issue with the channel reset sequence.
CH[0/1/2/3]_RXCDRRESET	Input	ASYNC	CDR phase detector reset.
CH[0/1/2/3]_RXPCSRESETMASK[2:0]	Input	ASYNC	RX PCS reset mask selection. Use the recommended values from the Wizard.
CH[0/1/2/3]_RXPMARESETDONE	Output	ASYNC	This active-High signal indicates RX PMA reset is complete.
CH[0/1/2/3]_RXPMARESETMASK[7:0]	Input	ASYNC	RX PMA reset mask selection. Use the recommended values from the Wizard.
CH[0/1/2/3]_RXPRBSCNTRESET	Input	ASYNC	Assert this port High to reset the PRBS error counter and RXPHYCLK counter.
CH[0/1/2/3]_RXPROGDIVRESET	Input	ASYNC	This active-High port resets the dividers as well as the CH*_RXPROGDIVRESETDONE indicator. A reset must be performed whenever the input clock source is interrupted.
CH[0/1/2/3]_RXPROGDIVRESETDONE	Output	ASYNC	When the input clock is stable and reset is performed, this active-High signal indicates the reset is completed and the output clock is stable.
CH[0/1/2/3]_RXRESETDONE	Output	ASYNC	This active-High signal indicates the GTM transceiver RX has finished reset and is ready for use.
CH[0/1/2/3]_RXRESETMODE[1:0]	Input	ASYNC	Reset mode port for RX.
CH[0/1/2/3]_RXUSERRDY	Input	ASYNC	This port is driven High from your application when RXUSRCLK and RXUSRCLK2 are stable.
CH[0/1/2/3]_TXPCSRESETMASK	Input	ASYNC	TX PCS reset mask selection. Use the recommended values from the Wizard.
CH[0/1/2/3]_TXPMARESETDONE	Output	ASYNC	This active-High signal indicates TX PMA reset is complete.
CH[0/1/2/3]_TXPMARESETMASK[1:0]	Input	ASYNC	TX PMA reset mask selection. Use the recommended values from the Wizard.
CH[0/1/2/3]_TXPROGDIVRESET	Input	ASYNC	This active-High port resets the dividers as well as the CH*_TXPROGDIVRESETDONE indicator. A reset must be performed whenever the input clock source is interrupted.
CH[0/1/2/3]_TXPROGDIVRESETDONE	Output	ASYNC	When the input clock is stable and reset is performed, this active-High signal indicates the reset is completed and the output clock is stable.

Table 25: Reset and Initialization Ports (cont'd)

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_TXRESETDONE	Output	ASYNC	This active-High signal indicates the GTM transceiver TX has finished reset and is ready for use.
CH[0/1/2/3]_TXRESETMODE[1:0]	Input	ASYNC	This active-High signal indicates the GTM transceiver TX has finished reset and is ready for use.
CH[0/1/2/3]_TXUSERRDY	Input	ASYNC	This port is driven High from your application when TXUSRCLK and TXUSRCLK2 are stable.

The following table lists attributes required by GTM transceiver TX and RX initialization. In general cases, the reset time required by the TX PMA or the TX PCS varies depending on line rate. The factors affecting PMA reset time and PCS reset time are the user-configurable attributes TX_PMA_RESET_TIME and TX_PCS_RESET_TIME.

Table 26: Reset and Initialization Attributes

Reset and Initialization Attributes		
Attribute	Address	
CTRL_RSV_CFG0	0x0C03	
Label	Bit Field	Description
CH3_RX_CH_RESET_EN	[31:31]	Reset Mask for Master Reset Looper CH3_RX_CHANNEL
CH3_RX_PROGDIV_RESET_EN	[30:30]	Reset Mask for Master Reset Looper CH3_RX_PROGDIV
CH3_RX_ILO_RESET_EN	[29:29]	Reset Mask for Master Reset Looper CH3_RX_ILO
CH3_RX_LCPLL1_RESET_EN	[28:28]	Reset Mask for Master Reset Looper CH3_RX_LCPLL1
CH3_RX_LCPLL0_RESET_EN	[27:27]	Reset Mask for Master Reset Looper CH3_RX_LCPLL0.
Attribute	Address	
HSCLK0_LCPLL_LGC_CFG0	0x0C3C	
HSCLK1_LCPLL_LGC_CFG0	0x0E3C	
Label	Bit Field	Description
LCPLLLOCKEN	[11:11]	Active-High signal enables the LCPLL lock detector.
Attribute	Address	
MST_RESET_CFG	0x0D12	
Label	Bit Field	Description
CH2_RX_CH_RESET_EN	[30:30]	Reset Mask for Master Reset Looper RX CH2 Channel
CH2_RX_PROGDIV_RESET_EN	[29:29]	Reset Mask for Master Reset Looper RX CH2 PROGDIV
CH2_RX_ILO_RESET_EN	[28:28]	Reset Mask for Master Reset Looper RX CH2 ILO
CH2_RX_LCPLL1_RESET_EN	[27:27]	Reset Mask for Master Reset Looper RX CH2 LCPLL1
CH2_RX_LCPLL0_RESET_EN	[26:26]	Reset Mask for Master Reset Looper RX CH2 LCPLL0
CH1_RX_CH_RESET_EN	[25:25]	Reset Mask for Master Reset Looper RX CH1 Channel
CH1_RX_PROGDIV_RESET_EN	[24:24]	Reset Mask for Master Reset Looper RX CH1 PROGDIV

Table 26: Reset and Initialization Attributes (cont'd)

Reset and Initialization Attributes		
CH1_RX_ILO_RESET_EN	[23:23]	Reset Mask for Master Reset Looper RX CH1 ILO
CH1_RX_LCPLL1_RESET_EN	[22:22]	Reset Mask for Master Reset Looper RX CH1 LCPLL1
CH1_RX_LCPLL0_RESET_EN	[21:21]	Reset Mask for Master Reset Looper RX CH1 LCPLL0
CH0_RX_CH_RESET_EN	[20:20]	Reset Mask for Master Reset Looper RX CH0 Channel
CH0_RX_PROGDIV_RESET_EN	[19:19]	Reset Mask for Master Reset Looper RX CH0 PROGDIV
CH0_RX_ILO_RESET_EN	[18:18]	Reset Mask for Master Reset Looper RX CH0 ILO
CH0_RX_LCPLL1_RESET_EN	[17:17]	Reset Mask for Master Reset Looper RX CH0 LCPLL1
CH0_RX_LCPLL0_RESET_EN	[16:16]	Reset Mask for Master Reset Looper RX CH0 LCPLL0
CH3_TX_CH_RESET_EN	[15:15]	Reset Mask for Master Reset Looper TX CH3 Channel
CH3_TX_PROGDIV_RESET_EN	[14:14]	Reset Mask for Master Reset Looper TX CH3 PROGDIV
CH3_TX_LCPLL1_RESET_EN	[13:13]	Reset Mask for Master Reset Looper TX CH3 LCPLL1
CH3_TX_LCPLL0_RESET_EN	[12:12]	Reset Mask for Master Reset Looper TX CH3 LCPLL0
CH2_TX_CH_RESET_EN	[11:11]	Reset Mask for Master Reset Looper TX CH2 Channel
CH2_TX_PROGDIV_RESET_EN	[10:10]	Reset Mask for Master Reset Looper TX CH2 PROGDIV
CH2_TX_LCPLL1_RESET_EN	[9:9]	Reset Mask for Master Reset Looper TX CH2 LCPLL1
CH2_TX_LCPLL0_RESET_EN	[8:8]	Reset Mask for Master Reset Looper TX CH2 LCPLL0
CH1_TX_CH_RESET_EN	[7:7]	Reset Mask for Master Reset Looper TX CH1 Channel
CH1_TX_PROGDIV_RESET_EN	[6:6]	Reset Mask for Master Reset Looper TX CH1 PROGDIV
CH1_TX_LCPLL1_RESET_EN	[5:5]	Reset Mask for Master Reset Looper TX CH1 LCPLL1
CH1_TX_LCPLL0_RESET_EN	[4:4]	Reset Mask for Master Reset Looper TX CH1 LCPLL0
CH0_TX_CH_RESET_EN	[3:3]	Reset Mask for Master Reset Looper TX CH0 Channel
CH0_TX_PROGDIV_RESET_EN	[2:2]	Reset Mask for Master Reset Looper TX CH0 PROGDIV
CH0_TX_LCPLL1_RESET_EN	[1:1]	Reset Mask for Master Reset Looper TX CH0 LCPLL1
CH0_TX_LCPLL0_RESET_EN	[0:0]	Reset Mask for Master Reset Looper TX CH0 LCPLL0
Attribute	Address	
CH0_RESET_TIME_CFG0	0x0C40	
CH1_RESET_TIME_CFG0	0x0D40	
CH2_RESET_TIME_CFG0	0x0E40	
CH3_RESET_TIME_CFG0	0x0F40	
Label	Bit Field	Description
TX_PCS_RESET_TIME	[19:15]	Reserved. Use the recommended value from the Wizard.
TX_PMA_RESET_TIME	[14:10]	Reserved. Use the recommended value from the Wizard.
Attribute	Address	
CH0_RESET_TIME_CFG1	0x0C41	
CH1_RESET_TIME_CFG1	0x0D41	
CH2_RESET_TIME_CFG1	0x0E41	
CH3_RESET_TIME_CFG1	0x0F41	

Table 26: Reset and Initialization Attributes (cont'd)

Reset and Initialization Attributes		
Label	Bit Field	Description
RX_PMA_RESET_TIME	[4:0]	Reserved. Use the recommended value from the Wizard.
Attribute	Address	
CH0_RESET_TIME_CFG3	0x0C43	
CH1_RESET_TIME_CFG3	0x0D43	
CH2_RESET_TIME_CFG3	0x0E43	
CH3_RESET_TIME_CFG3	0x0F43	
Label	Bit Field	Description
RX_PCS_RESET_TIME	[10:0]	Reserved. Use the recommended value from the Wizard.
Attribute	Address	
RXRSTDONE_DIST_SEL	0x0D11	
Label	Bit Field	Description
CH3_CONSUMPT_SEL	[30:28]	RX Master reset controller RESETDONE daisy chain consumption selection for channel 3
CH3_DISTSOUTH_SEL	[27:26]	RX Master reset controller RESETDONE daisy chain distribution south selection for channel 3
CH3_DISTNORTH_SEL	[25:24]	RX Master reset controller RESETDONE daisy chain distribution north selection for channel 3
CH2_CONSUMPT_SEL	[22:20]	RX Master reset controller RESETDONE daisy chain consumption selection for channel 2
CH2_DISTSOUTH_SEL	[19:18]	RX Master reset controller RESETDONE daisy chain distribution south selection for channel 2
CH2_DISTNORTH_SEL	[17:16]	RX Master reset controller RESETDONE daisy chain distribution north selection for channel 2
CH1_CONSUMPT_SEL	[14:12]	RX Master reset controller RESETDONE daisy chain consumption selection for channel 1
CH1_DISTSOUTH_SEL	[11:10]	RX Master reset controller RESETDONE daisy chain distribution south selection for channel 1
CH1_DISTNORTH_SEL	[9:8]	RX Master reset controller RESETDONE daisy chain distribution north selection for channel 1
CH0_CONSUMPT_SEL	[6:4]	RX Master reset controller RESETDONE daisy chain consumption selection for channel 0
CH0_DISTSOUTH_SEL	[3:2]	RX Master reset controller RESETDONE daisy chain distribution south selection for channel 0
CH0_DISTNORTH_SEL	[1:0]	RX Master reset controller RESETDONE daisy chain distribution north selection for channel 0
Attribute	Address	
TXRSTDONE_DIST_SEL	0x0D10	
Label	Bit Field	Description
CH3_CONSUMPT_SEL	[30:28]	TX Master reset controller RESETDONE daisy chain consumption selection for channel 3
CH3_DISTSOUTH_SEL	[27:26]	TX Master reset controller RESETDONE daisy chain distribution south selection for channel 3

Table 26: Reset and Initialization Attributes (cont'd)

Reset and Initialization Attributes		
CH3_DISTNORTH_SEL	[25:24]	TX Master reset controller RESETDONE daisy chain distribution north selection for channel 3
CH2_CONSUMPT_SEL	[22:20]	TX Master reset controller RESETDONE daisy chain consumption selection for channel 2
CH2_DISTSOUTH_SEL	[19:18]	TX Master reset controller RESETDONE daisy chain distribution south selection for channel 2
CH2_DISTNORTH_SEL	[17:16]	TX Master reset controller RESETDONE daisy chain distribution north selection for channel 2
CH1_CONSUMPT_SEL	[14:12]	TX Master reset controller RESETDONE daisy chain consumption selection for channel 1
CH1_DISTSOUTH_SEL	[11:10]	TX Master reset controller RESETDONE daisy chain distribution south selection for channel 1
CH1_DISTNORTH_SEL	[9:8]	TX Master reset controller RESETDONE daisy chain distribution north selection for channel 1
CH0_CONSUMPT_SEL	[6:4]	TX Master reset controller RESETDONE daisy chain consumption selection for channel 0
CH0_DISTSOUTH_SEL	[3:2]	TX Master reset controller RESETDONE daisy chain distribution south selection for channel 0
CH0_DISTNORTH_SEL	[1:0]	TX Master reset controller RESETDONE daisy chain distribution north selection for channel 0

Rate Change

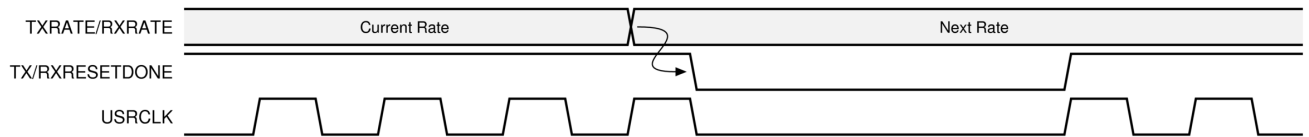
The Versal adaptive SoC GTM transceiver provides great flexibility by allowing users to dynamically change the operating line rate. To ensure all attributes are correctly configured during the rate change, AMD recommends generating wrappers from the Versal Adaptive SoC Transceivers Wizard (Wizard) that allows the designer to preconfigure a set of line rates at which the device is intended to operate.

When a rate change is performed using the CH*_TXRATE or CH*_RXRATE port to any of the preconfigured line rates set using the Wizard, the required reset sequence and necessary attribute update is performed automatically. The user should wait for the assertion of CH*_TXRESETDONE or CH*_RXRESETDONE as an indication that the current rate change process and necessary reset sequence have completed.

Rate Change Use Mode Without Reference Clock Change

If the reference clock frequency does not need to change, the user shall use the rate change port and wait for the CH*_TXRESETDONE or CH*_RXRESETDONE assertion to signal the rate change process completion. The following timing diagram shows the sequence details.

Figure 27: Transceiver Rate Change with No Changes in Reference Clock

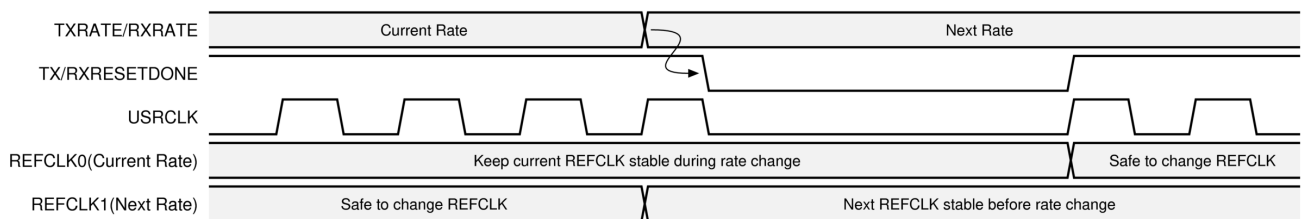


Rate Change Use Mode with Reference Clock Changes

If the user decides to use a different reference clock, whether it is an entirely different clock source or the same clock source but with only changes in the actual frequency, additional steps need to be followed when performing the rate change.

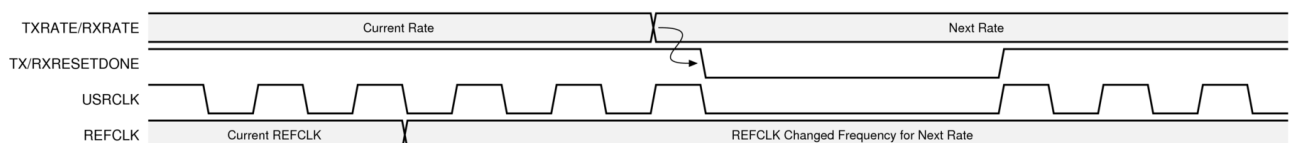
If the user decides to use a different reference clock port/source, the current reference clock source must be kept unchanged and stable during the entire rate change sequence. At the same time, the new reference clock source should be set and stable when the rate change procedure starts. The following timing diagram shows the required sequence.

Figure 28: Transceiver Rate Change with Changes in REFCLK Source



If the user decides to use the same reference clock source but the actual frequency changes, the new reference clock frequency should be set and stable prior to toggling rate change. The following timing diagram shows the proper rate change sequence under this use mode.

Figure 29: Transceiver Rate Change with Changes in REFCLK Frequency without Port Change



Rate Change Notes

In a design generated through the Wizard containing multiple rates, if the device registers were manually modified by the user, this might cause unexpected behaviors. If a register manually set by the user is part of the rate change sequence, its value will be overwritten during the rate change procedure.

The rate change algorithm does not keep a record of the original device state before any manual register overwrites. This means that if the user modifies a register that is not part of the rate change sequence, it will not be refreshed back to its default state even after each rate change procedure. This could lead to a situation where the register that was manually modified by the user might not be the most optimal and could cause performance issues in the new line rate configuration. Therefore, it is up to the user to track all manually modified register values before and after each rate change and correct them if necessary following the recommendations in this manual.

Ports and Attributes

The following table defines the rate change ports.

Table 27: Rate Change Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_RXRATE[7:0]	Input	ASYNC	This port is used to perform rate change on the transceiver RX. The Wizard will preconfigure a list of desired line rates, and this port will be used to dynamically adjust the running line rate based on the preconfigured list. Set this port to the matching preconfigured line rate option value to obtain the proper line rate.
CH[0/1/2/3]_TXRATE[7:0]	Input	ASYNC	This port is used to perform rate change on the transceiver TX. The Wizard will preconfigure a list of desired line rates, and this port will be used to dynamically adjust the running line rate based on the preconfigured list. Set this port to the matching preconfigured line rate option value to obtain the proper line rate.

The rate change attributes in Versal adaptive SoCs are only attributes that differ between the configurations given to the wizard setup. Refer to [Sub GUI Configuration Tab](#) in *Versal Adaptive SoC Transceivers Wizard LogiCORE IP Product Guide (PG331)* for information on which attributes are involved in the rate change sequence.

Power Down

The GTM transceivers support a range of power-down modes. These modes support both generic power management capabilities as well as those defined in the SATA standard.

The GTM transceivers offer different levels of power control. Each channel in each direction can be powered down separately using CH*_TXPD and CH*_RXP. The A_HSCLK*_LCPLL attribute directly affects the LCPLL.

PLL Power Down

To activate the LCPLL power-down mode, the active-High A_HSCLK*_LCPLL signal is asserted. When A_HSCLK*_LCPLL is asserted, the PLL is powered down. As a result, all clocks derived from the PLL are stopped. Recovery from this power state is indicated by the assertion of the PLL lock signal HSCLK*_LCPLLLOCK.

TX and RX Power Down

TX and RX power control signals can be used independently through CH*_TXPD and CH*_RXP. Only two power states are supported, as shown in the following table. When using this power-down mechanism, these must be true:

- CH*_TXPD[1] and CH*_TXPD[0] are connected together.
- CH*_RXP[1] and CH*_RXP[0] are connected together.

For half density mode, the channels with unused PMA must be powered down (TX/RXP[1:0] = 2'b11) to conform with the values from the Power Design Manager (PDM) tool (download at www.amd.com/power).

Table 28: TX and RX Power States

CH*_TXPD[1:0] or CH*_RXP[1:0]	Description
2'b00	Normal mode. Transceiver TX or RX is actively sending or receiving data.
2'b11	Power-down mode. Transceiver TX or RX is idle.

Ports and Attributes

The following table defines the power down ports.

Table 29: Powerdown Ports

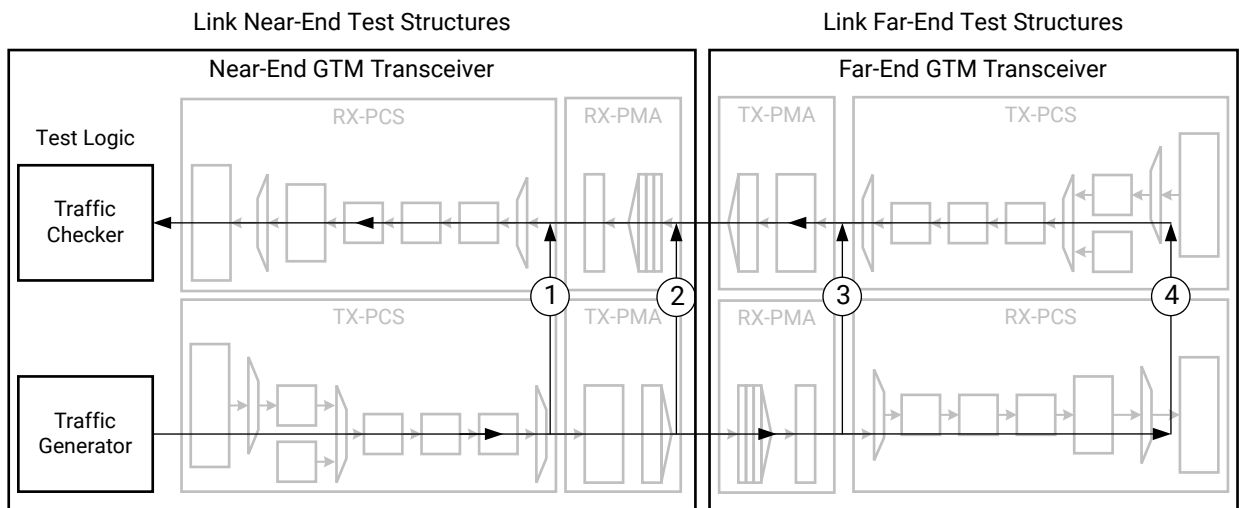
Port	Direction	Clock Domain	Description
HSCLK[0/1]_LCPLLDP	Input	ASYNC	Reserved. Do not use. Use A_HS[0/1]_LCPLLDP instead to power down the LCPLL.
CH[0/1/2/3]_RXPD[1:0]	Input	ASYNC	Powers down the RX lane. 2'b00: Power Up (Normal operation) 2'b11: Power Down
CH[0/1/2/3]_TXPD[1:0]	Input	ASYNC	Powers down the TX lane. 2'b00: Power Up (Normal operation) 2'b11: Power Down

There are no power down attributes.

Loopback

Loopback modes are specialized configurations of the transceiver datapath where the traffic stream is folded back to the source. Typically, a specific pattern is transmitted then compared to check for errors. The following figure illustrates a loopback test configuration with four different loopback modes.

Figure 30: Loopback Testing Overview



X24138-062220

Loopback test modes fall into two broad categories:

- Near-end loopback modes loop transmitted data back in the transceiver closest to the traffic generator.

- Far-end loopback modes loop received data back in the transceiver at the far end of the link.

Loopback testing can be used either during development or in deployed equipment for fault isolation. The traffic patterns used can either be application traffic patterns or specialized pseudo-random bit sequences. Each GTM transceiver has a built-in PRBS generator and checker.

Each GTM transceiver features several loopback modes to facilitate testing:

- Near-end PCS loopback (path 1 in the preceding figure). To set the GTM transceiver in near-end PCS loopback, set the following:
 - Set CH[0/1/2/3]_RXCDRHOLD = 1'b1
- Near-end PMA loopback (path 2 in the preceding figure).
- Far-end PMA loopback (path 3 in the preceding figure). The transceiver in far-end PMA loopback must use the same reference clock used by the transceiver that is the source of the loopback data because synchronous operation is required. A GTTXRESET is required after entering or exiting far-end PMA loopback.
- Far-end PCS loopback (path 4 in the preceding figure). The transceiver in far-end PCS loopback must use the same reference clock used by the transceiver that is the source of the loopback data because synchronous operation is required. Regardless of whether or not clock correction is used, the TXUSRCLK and RXUSRCLK ports must be driven by the same clocking resource (BUFG_GT).

Note: If the receiver is driven by a far-end transmitter, up to three orders of magnitude in BER degradation is expected.

Ports and Attributes

The following tables define the ports and attributes for loopback modes.

Table 30: Loopback Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_LOOPBACK[2:0]	Input	ASYNC	Controls the loopback mode when LOOPBACK_PIN_EN is set to 1'b1. 3'b000: Normal Operation 3'b001: Near-End PCS Loopback 3'b010: Near-End PMA Loopback 3'b100: Far-End PMA Loopback 3'b110: Far-End PCS Loopback

Table 31: Loopback Attributes

Loopback Attributes	
Attribute	Address
CH0_FABRIC_INTF_CFG0	0x0C4C

Table 31: Loopback Attributes (cont'd)

Loopback Attributes		
CH1_FABRIC_INTF_CFG0		0x0D4C
CH2_FABRIC_INTF_CFG0		0x0E4C
CH3_FABRIC_INTF_CFG0		0x0F4C
Label	Bit Field	Description
LOOPBACK_PIN_EN	[23:23]	This attribute dictates how the loopback mode is controlled. 1'b1: Loopback control is set using port CH*_LOOPBACK[2:0] 1'b0: Loopback control is set using attribute A_LOOPBACK[2:0]
Attribute	Address	
CH0_FABRIC_INTF_CFG1		0x0C4D
CH1_FABRIC_INTF_CFG1		0x0D4D
CH2_FABRIC_INTF_CFG1		0x0E4D
CH3_FABRIC_INTF_CFG1		0x0F4D
Label	Bit Field	Description
A_LOOPBACK	[9:7]	Controls the loopback mode when LOOPBACK_PIN_EN is set to 1'b0. 3'b000: Normal Operation 3'b001: Near-End PCS Loopback 3'b010: Near-End PMA Loopback 3'b100: Far-End PMA Loopback 3'b110: Far-End PCS Loopback

Fabric Configuration Interface

The fabric configuration interface (APB3) allows the user to dynamically update the attributes of the GTME5_QUAD primitive. The APB3 interface is a processor-friendly synchronous interface with an address bus (APB3PADDR) and separate data buses for reading (APB3PRDATA) and writing (APB3PDATA) configuration data to the primitive. An enable signal (APB3PENABLE), a read/write signal (APB3PWRITE), and a ready/valid signal (APB3PREADY) are the control signals that implement read and write operations, indicate operation completion, and indicate the availability of data. When TXOUTCLK from one lane is used to drive multiple lanes (TXUSRCLK), the corresponding PLL for each lane must share the same reference clock.

Ports and Attributes

The following table shows the APB3 ports.

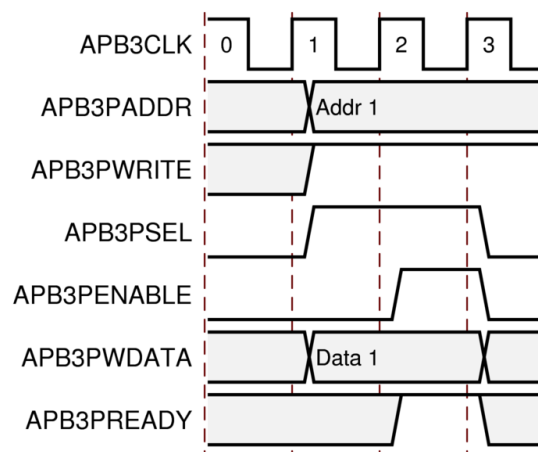
Table 32: APB3 Ports

Port	Direction	Clock Domain	Description
APB3CLK	Input	CLOCK	This port is used to provide a clock for the APB interface.
APB3PADDR[15:0]	Input	APB3CLK	Configuration data address.
APB3PENABLE	Input	APB3CLK	Set to 1'b1 to enable read/write requests.
APB3PRDATA[31:0]	Output	APB3CLK	Data bus for reading configuration data from the transceiver to the interconnect logic resources.
APB3PREADY	Output	APB3CLK	Indicates operation is completed for write operations and data is valid for read operations.
APB3PRESETN	Input	ASYNC	Active low reset. This port resets the APB3 read/write access logic.
APB3PSEL	Input	APB3CLK	The APB master generates this signal to each bus slave. It indicates that the slave device is selected and that a data transfer is required.
APB3PSLVERR	Output	APB3CLK	Indicates slave error when invalid address or when request is unable to complete.
APB3PWDATA[31:0]	Input	APB3CLK	Data bus for writing configuration data from the interconnect logic resources to the transceiver.
APB3PWRITE	Input	APB3CLK	Write enable. 0: For read request. 1: For write request.

Usage Mode

The following figure shows the write transfer timing diagram.

Figure 31: Write Transfer



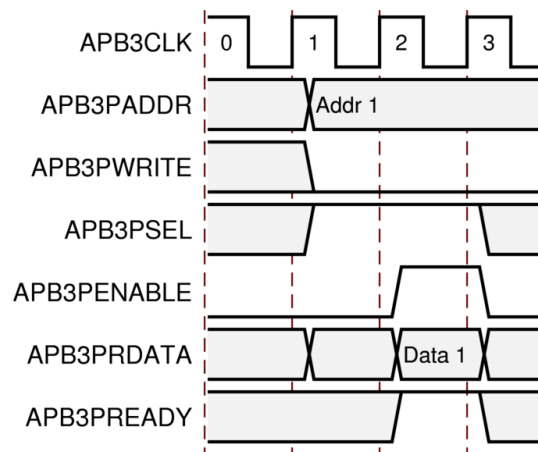
At T1, a write transfer starts with address APB3PADDR, write data APB3PWDATA, write signal APB3PWRITE, and select signal APB3PSEL, being registered at the rising edge of APB3CLK. This is called the Setup phase of the write transfer.

At T2, enable signal APB3PENABLE and ready signal APB3PREADY are registered at the rising edge of APB3CLK. When asserted, APB3PENABLE indicates the start of the Access phase of the transfer. When asserted, APB3PREADY indicates that the slave can complete the transfer at the next rising edge of APB3CLK.

The address APB3PADDR, write data APB3PWDATA, and control signals all remain valid until the transfer completes at T3, the end of the Access phase.

The enable signal APB3PENABLE is deasserted at the end of the transfer. The select signal APB3PSEL is also deasserted unless the transfer is to be followed immediately by another transfer to the same peripheral.

Figure 32: Read Transfer



The timing of the address, write, select, and enable signals are as described for write transfers. The slave must provide the data before the end of the read transfer.

APB3ADDR must not be set in the range of 0x00 through 0x0C even though the actual read or write transaction is not performed. If it happens, the data from the next subsequent read must be discarded.

Digital Monitor

The receiver uses an adaptive algorithm in optimizing a link. The digital monitor provides visibility into the current state of these adaptation loops. Digital monitor requires a clock: RXUSRCLK or any clock under 500 MHz can be used for this. Read-only registers are used to read the current codes for the adaptation loops. See Use Mode to enable the digital monitor.

Ports and Attributes

The following table shows the digital monitor ports.

Table 33: Digital Monitor Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_DMONITORCLK	Input	CLOCK	Digital monitor clock.

The following table shows the digital monitor attributes.

Table 34: Digital Monitor Attributes

Digital Monitor Attributes		
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP07	0x0887	
CH1_COESTATUS_ADAPT_LOOP07	0x0987	
CH2_COESTATUS_ADAPT_LOOP07	0x0A87	
CH3_COESTATUS_ADAPT_LOOP07	0x0B87	
Label	Bit Field	Description
H1_DAC	[12:0]	Adaptation DAC for H1 loop. H1_DAC is a signed 13-bit number with MSB as the sign bit and the last 6 LSB as the fractional bits. Range of -64 to +63.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP08	0x0888	
CH1_COESTATUS_ADAPT_LOOP08	0x0988	
CH2_COESTATUS_ADAPT_LOOP08	0x0A88	
CH3_COESTATUS_ADAPT_LOOP08	0x0B88	
Label	Bit Field	Description
HM01_DAC	[8:0]	Adaptation DAC for HM01 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP09	0x0889	
CH1_COESTATUS_ADAPT_LOOP09	0x0989	
CH2_COESTATUS_ADAPT_LOOP09	0x0A89	

Table 34: Digital Monitor Attributes (cont'd)

Digital Monitor Attributes		
CH3_COESTATUS_ADAPT_LOOP09		0x0B89
Label	Bit Field	Description
HP01_DAC	[8:0]	Adaptation DAC for HP01 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP10		0x088A
CH1_COESTATUS_ADAPT_LOOP10		0x098A
CH2_COESTATUS_ADAPT_LOOP10		0x0A8A
CH3_COESTATUS_ADAPT_LOOP10		0x0B8A
Label	Bit Field	Description
HM02_DAC	[15:8]	Adaptation DAC for HM02 loop. This value is in two's complement format.
HP02_DAC	[7:0]	Adaptation DAC for HP02 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP11		0x088B
CH1_COESTATUS_ADAPT_LOOP11		0x098B
CH2_COESTATUS_ADAPT_LOOP11		0x0A8B
CH3_COESTATUS_ADAPT_LOOP11		0x0B8B
Label	Bit Field	Description
HM03_DAC	[14:8]	Adaptation DAC for HM03 loop. This value is in two's complement format.
HP03_DAC	[7:0]	Adaptation DAC for HP03 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP12		0x088C
CH1_COESTATUS_ADAPT_LOOP12		0x098C
CH2_COESTATUS_ADAPT_LOOP12		0x0A8C
CH3_COESTATUS_ADAPT_LOOP12		0x0B8C
Label	Bit Field	Description
HM04_DAC	[14:8]	Adaptation DAC for HM04 loop. This value is in two's complement format.
HP04_DAC	[6:0]	Adaptation DAC for HP04 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP13		0x088D
CH1_COESTATUS_ADAPT_LOOP13		0x098D
CH2_COESTATUS_ADAPT_LOOP13		0x0A8D
CH3_COESTATUS_ADAPT_LOOP13		0x0B8D

Table 34: Digital Monitor Attributes (cont'd)

Digital Monitor Attributes		
Label	Bit Field	Description
HM05_DAC	[13:8]	Adaptation DAC for HM05 loop. This value is in two's complement format.
HP05_DAC	[6:0]	Adaptation DAC for HP05 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP14	0x088E	
CH1_COESTATUS_ADAPT_LOOP14	0x098E	
CH2_COESTATUS_ADAPT_LOOP14	0x0A8E	
CH3_COESTATUS_ADAPT_LOOP14	0x0B8E	
Label	Bit Field	Description
HM06_DAC	[13:8]	Adaptation DAC for HM06 loop. This value is in two's complement format.
HP06_DAC	[5:0]	Adaptation DAC for HP06 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP15	0x088F	
CH1_COESTATUS_ADAPT_LOOP15	0x098F	
CH2_COESTATUS_ADAPT_LOOP15	0x0A8F	
CH3_COESTATUS_ADAPT_LOOP15	0x0B8F	
Label	Bit Field	Description
HM07_DAC	[12:8]	Adaptation DAC for HM07 loop. This value is in two's complement format.
HP07_DAC	[5:0]	Adaptation DAC for HP07 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP16	0x0890	
CH1_COESTATUS_ADAPT_LOOP16	0x0990	
CH2_COESTATUS_ADAPT_LOOP16	0x0A90	
CH3_COESTATUS_ADAPT_LOOP16	0x0B90	
Label	Bit Field	Description
HM08_DAC	[11:8]	Adaptation DAC for HM08 loop. This value is in two's complement format.
HP08_DAC	[5:0]	Adaptation DAC for HP08 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP17	0x0891	
CH1_COESTATUS_ADAPT_LOOP17	0x0991	
CH2_COESTATUS_ADAPT_LOOP17	0x0A91	
CH3_COESTATUS_ADAPT_LOOP17	0x0B91	

Table 34: Digital Monitor Attributes (cont'd)

Digital Monitor Attributes		
Label	Bit Field	Description
HP10_DAC	[13:8]	Adaptation DAC for HP10 loop. This value is in two's complement format.
HP09_DAC	[5:0]	Adaptation DAC for HP09 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP18	0x0892	
CH1_COESTATUS_ADAPT_LOOP18	0x0992	
CH2_COESTATUS_ADAPT_LOOP18	0x0A92	
CH3_COESTATUS_ADAPT_LOOP18	0x0B92	
Label	Bit Field	Description
HP12_DAC	[13:8]	Adaptation DAC for HP12 loop. This value is in two's complement format.
HP11_DAC	[5:0]	Adaptation DAC for HP11 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP19	0x0893	
CH1_COESTATUS_ADAPT_LOOP19	0x0993	
CH2_COESTATUS_ADAPT_LOOP19	0x0A93	
CH3_COESTATUS_ADAPT_LOOP19	0x0B93	
Label	Bit Field	Description
HP14_DAC	[13:8]	Adaptation DAC for HP14 loop. This value is in two's complement format.
HP13_DAC	[5:0]	Adaptation DAC for HP13 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP20	0x0894	
CH1_COESTATUS_ADAPT_LOOP20	0x0994	
CH2_COESTATUS_ADAPT_LOOP20	0x0A94	
CH3_COESTATUS_ADAPT_LOOP20	0x0B94	
Label	Bit Field	Description
HP16_DAC	[12:8]	Adaptation DAC for HP16 loop. This value is in two's complement format.
HP15_DAC	[5:0]	Adaptation DAC for HP15 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP21	0x0895	
CH1_COESTATUS_ADAPT_LOOP21	0x0995	
CH2_COESTATUS_ADAPT_LOOP21	0x0A95	
CH3_COESTATUS_ADAPT_LOOP21	0x0B95	

Table 34: Digital Monitor Attributes (cont'd)

Digital Monitor Attributes		
Label	Bit Field	Description
HP18_DAC	[12:8]	Adaptation DAC for HP18 loop. This value is in two's complement format.
HP17_DAC	[4:0]	Adaptation DAC for HP17 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP22	0x0896	
CH1_COESTATUS_ADAPT_LOOP22	0x0996	
CH2_COESTATUS_ADAPT_LOOP22	0x0A96	
CH3_COESTATUS_ADAPT_LOOP22	0x0B96	
Label	Bit Field	Description
HP20_DAC	[11:8]	Adaptation DAC for HP20 loop. This value is in two's complement format.
HP19_DAC	[4:0]	Adaptation DAC for HP19 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP23	0x0897	
CH1_COESTATUS_ADAPT_LOOP23	0x0997	
CH2_COESTATUS_ADAPT_LOOP23	0x0A97	
CH3_COESTATUS_ADAPT_LOOP23	0x0B97	
Label	Bit Field	Description
HP23_DAC	[15:12]	Adaptation DAC for HP23 loop. This value is in two's complement format.
HP22_DAC	[11:8]	Adaptation DAC for HP22 loop. This value is in two's complement format.
HP21_DAC	[3:0]	Adaptation DAC for HP21 loop. This value is in two's complement format.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP24	0x0898	
CH1_COESTATUS_ADAPT_LOOP24	0x0998	
CH2_COESTATUS_ADAPT_LOOP24	0x0A98	
CH3_COESTATUS_ADAPT_LOOP24	0x0B98	
Label	Bit Field	Description
OS_DACP	[13:8]	Adaptation DAC for OSP loop. This value is unsigned.
OS_DACN	[5:0]	Adaptation DAC for OSN loop. This value is unsigned.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP25	0x0899	
CH1_COESTATUS_ADAPT_LOOP25	0x0999	
CH2_COESTATUS_ADAPT_LOOP25	0x0A99	
CH3_COESTATUS_ADAPT_LOOP25	0x0B99	

Table 34: Digital Monitor Attributes (cont'd)

Digital Monitor Attributes		
Label	Bit Field	Description
AGC_DAC1	[12:8]	Adaptation DAC for AGC1 loop. This value is unsigned.
AGC_DAC2	[4:0]	Adaptation DAC for AGC2 loop. This value is unsigned.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP26	0x089A	
CH1_COESTATUS_ADAPT_LOOP26	0x099A	
CH2_COESTATUS_ADAPT_LOOP26	0x0A9A	
CH3_COESTATUS_ADAPT_LOOP26	0x0B9A	
Label	Bit Field	Description
KH_DAC1	[12:8]	Adaptation DAC for KH1 loop. This value is unsigned.
KH_DAC2	[4:0]	Adaptation DAC for KH2 loop. This value is unsigned.
Attribute	Address	
CH0_COESTATUS_ADAPT_LOOP27	0x089B	
CH1_COESTATUS_ADAPT_LOOP27	0x099B	
CH2_COESTATUS_ADAPT_LOOP27	0x0A9B	
CH3_COESTATUS_ADAPT_LOOP27	0x0B9B	
Label	Bit Field	Description
KL_DAC1	[11:8]	Adaptation DAC for KL1 loop. This value is unsigned.
KL_DAC2	[3:0]	Adaptation DAC for KL2 loop. This value is unsigned.
Attribute	Address	
CH0_RX_APT_CFG5	0x0C65	
CH1_RX_APT_CFG5	0x0D65	
CH2_RX_APT_CFG5	0x0E65	
CH3_RX_APT_CFG5	0x0F65	
Label	Bit Field	Description
TESTCON	[20:17]	Set to 4b'0000 when monitoring adaptation loops.
COE_EN	[16:16]	Set to 1b'1 when monitoring adaptation loops.
Attribute	Address	
CH0_RX_APT_CFG53	0x0C95	
CH1_RX_APT_CFG53	0x0D95	
CH2_RX_APT_CFG53	0x0E95	
CH3_RX_APT_CFG53	0x0F95	
Label	Bit Field	Description
SELMINI	[28:28]	Set to 1b'0 when monitoring adaptation loops.
Attribute	Address	
CH0_RX_APT_CFG56	0x0C98	
CH1_RX_APT_CFG56	0x0D98	

Table 34: Digital Monitor Attributes (cont'd)

Digital Monitor Attributes		
CH2_RX_APT_CFG56		0x0E98
CH3_RX_APT_CFG56		0x0F98
Label	Bit Field	Description
TESTEN	[17:17]	DMON enable. Set to 1b'1 when monitoring adaptation loops.
TESTC	[16:15]	Set to 2b'00 when monitoring adaptation loops.

Use Mode

Reading loop values from digital monitor requires a clock on port CH[1/2/3/4]_DMONITORCLK along with setting attributes TESTCON, COE_EN, SELMINI, TESTEN, and TESTC. In GTM transceivers, each adaptation loop can be monitored in parallel through dedicated read-only registers.

For example, to read the values for adaptation loops KH, KL, and AGC for channel 0, set the following attributes:

- 0x0C65[20:17] = 4'b0000
- 0x0C65[16] = 1'b1
- 0x0C95[28] = 1'b0
- 0x0C98[17] = 1'b1
- 0x0C98[16:15] = 2'b00

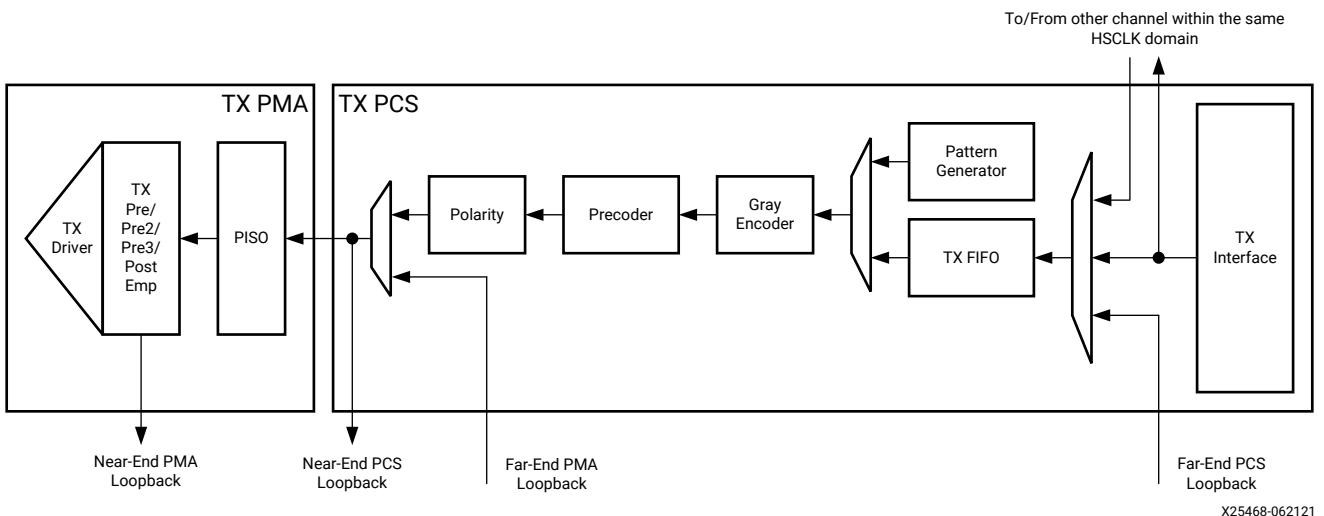
Use the following read-only registers to monitor KH, KL, and AGC for channel 0:

- 0x0899[15:8]
- 0x0899[7:0]
- 0x089A[15:8]
- 0x089A[7:0]
- 0x089B[15:8]
- 0x089B[7:0]

Transmitter

This chapter shows how to configure and use each of the functional blocks inside the transmitter (TX). Each transceiver includes an independent transmitter, which consists of a PCS and a PMA. The following figure shows the functional blocks of the transmitter. Parallel data flows from the device logic into the TX interface, through the PCS and PMA, and then out the TX driver as high-speed serial data.

Figure 33: GTM Transmitter Block Diagram



The key elements within the transceiver TX are:

- [TX Interface](#)
- [TX FIFO](#)
- [TX Pattern Generator](#)
- [TX Gray Encoder](#)
- [TX Pre-Coder](#)
- [TX Polarity Control](#)
- [TX Fabric Clock Output Control](#)
- [TX Configurable Driver](#)

TX Interface

The TX interface is the gateway to the TX datapath of the GTM transceiver. Applications transmit data through the GTM transceiver by writing data to the TXDATA port on the positive edge of TXUSRCLK. Port widths can be 32, 40, 64, 80, 128, 160, and 256 bits for NRZ mode, or 64, 80, 128, 160, 256, 320, and 512 bits for PAM4 mode. The rate of the parallel clock (TXUSRCLK) at the interface is determined by the TX line rate and the width of the TXDATA port.

Each TX fabric interface natively supports up to 256 bits from the fabric design input. To extend fabric interface support to 320 and 512 bits, two TX channel interfaces should be combined—this is referred to as half density mode. In this mode, only one TX front end channel is active. Refer to [TX Interface Width Configuration](#) for more details.

TX Interface Width Configuration

The GTM transmitter contains a 64-bit internal datapath in NRZ mode and a 128-bit internal datapath for PAM4 mode that is configurable by setting the TX_PMA_DATA_WIDTH attribute. The interface width is configurable by setting the TX_DATA_WIDTH attribute. In NRZ mode, the interface can be configured to 32, 40, 64, 80, 128, 160, and 256 bits. In PAM4 mode, the interface can be configured to 64, 80, 128, 160, 256, 320, and 512 bits.

The following table shows how the interface width for the TX datapath is selected.

Table 35: TX Interface Datapath Configuration

Encoding	Density	TX_PMA_DATA_WIDTH Encoding	TX_DATA_WIDTH Encoding	TX Internal Width	TX Interface Width
NRZ	Full	0	4	64	32
			5		40
			6		64
			7		80
			8		128
			9		160
			10		256
PAM4	Full	1	6	128	64
			7		80
			8		128
			9		160
			10		256
	Half		11		320
			12		512

When the interface width is configured to either 320 or 512 bits, this results in a half density configuration in which two channel interfaces are combined to act as one. In this mode, only a single PCS/PMA front end channel is in operation.

When either channel CH0 or CH1 is operating in half density mode, the fabric interface combines the data from adjacent channels to group as a set of 320 or 512 bits before forwarding to the TX FIFO. Designating a channel to be used in half density mode can be done by setting the adjacent channel's TX_USRCLK_SEL attribute High. An additional limitation is that the half density configuration must use either the channel 0/1 pair or channel 2/3 pair in the same Quad. Mixing and matching with different channels is not allowed. The following table shows possible channel combinations.

Table 36: TX Half Density Modes

CH0 TX_USRCLK_SEL	CH1 TX_USRCLK_SEL	Fabric Interface Operating Mode
0	0	Both channels are in full density mode.
0	1	CH0 is in half density mode.
1	0	CH1 is in half density mode.
1	1	Invalid.

The following table shows the port connections for different data widths.

Table 37: TX Fabric Port Connections for Different Data Widths

TX Fabric Data Width	Internal PCS Data Path Bit Ranges	Fabric Width Mode	Non-Active Channel (MSB) Last Bits to TX	Active Channel (LSB) First Bits to TX
512	[511:0]	Half Density	TXDATA[255:0]	TXDATA[255:0]
320	[415:256, 159:0]	Half Density	TXDATA[159:0]	TXDATA[159:0]
256	[255:0]	Full Density	-	TXDATA[255:0]
160	[159:0]	Full Density	-	TXDATA[159:0]
128	[127:0]	Full Density	-	TXDATA[127:0]
80	[79:0]	Full Density	-	TXDATA[79:0]
64	[63:0]	Full Density	-	TXDATA[63:0]
40	[39:0]	Full Density	-	TXDATA[39:0]
32	[31:0]	Full Density	-	TXDATA[31:0]

Notes:

1. In half-density mode, the least significant bits are in the active channel, and the most significant bits are in the non-active channel.
2. In 320-bit mode, the data widths are split between the two channels to have 160 in each.

Ports and Attributes

The following table shows the TX interface ports.

Table 38: TX Interface Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_TXDATA[255:0]	Input	TXUSRCLK	The bus for transmitting data.
CH[0/1/2/3]_TXUSRCLK	Input	CLOCK	This port is used to provide a clock for the internal PCS datapath.

The following table shows the TX interface attributes.

Table 39: TX Interface Attributes

TX Interface Attributes		
Attribute	Address	
CH0_TX_PCS_CFG0	0x0C16	
CH1_TX_PCS_CFG0	0x0D16	
CH2_TX_PCS_CFG0	0x0E16	
CH3_TX_PCS_CFG0	0x0F16	
Label	Bit Field	Description
TX_USRCLK_SEL	[15:14]	Clock selection for data usage at fabric interface. 2'b00: USRCLK for single channel data fabric interface usage (full density) 2'b01: Adjacent channel user clock for dual channel data fabric interface usage (half density) 2'b10: Reserved.
TX_PMA_DATA_WIDTH	[4:4]	Controls the width of the TX internal PCS datapath. 0: 64-bit internal datapath width (NRZ) 1: 128-bit internal datapath width (PAM4)
TX_DATA_WIDTH	[3:0]	Controls the TX interface width. 4'b0100: 32-bit 4'b0101: 40-bit 4'b0110: 64-bit 4'b0111: 80-bit 4'b1000: 128-bit 4'b1001: 160-bit 4'b1010: 256-bit 4'b1011: 320-bit 4'b1100: 512-bit

TXUSRCLK Generation

The TX Interface includes the parallel clock TXUSRCLK. The required rate for TXUSRCLK depends on the interface width of the GTME5_QUAD primitive and the TX line rate of the GTM transmitter. The following equation shows how to calculate the required rate for TXUSRCLK for all cases.

Equation 4: **TXUSRCLK**

$$TXUSRCLK = \frac{\text{Line Rate}}{\text{Interface Width}}$$

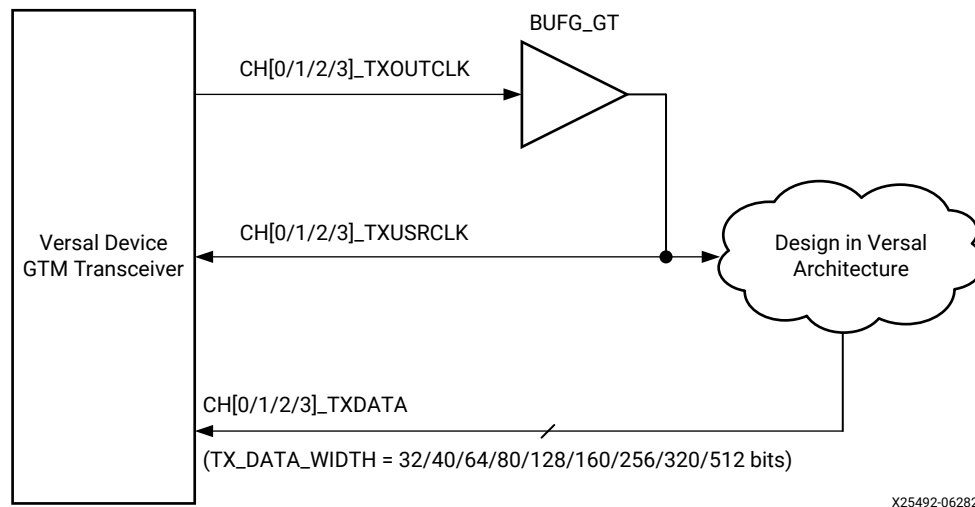
TXUSRCLK is the internal synchronization clock for all signals into the TX side of the GTM transceiver. Most signals into the TX side of the GTM transceiver are sampled on the positive edge of TXUSRCLK.

Driving the TX Interface

Depending on the TXUSRCLK frequency, there are different ways in which the Versal architecture clock resources can be used to drive the parallel clock for the TX interface. The following figure shows TXOUTCLK being used to drive TXUSRCLK in varying data widths.

- Depending on the input reference clock frequency and the required line rate, a BUFG_GT with the appropriate TXOUTCLKCTL setting is required. The Versal adaptive SoC Transceivers Wizard creates a sample design based on different design requirements for most cases.
- When TXOUTCLK from one lane is used to drive multiple lanes (TXUSRCLK), the corresponding PLL for each lane must share the same reference clock.

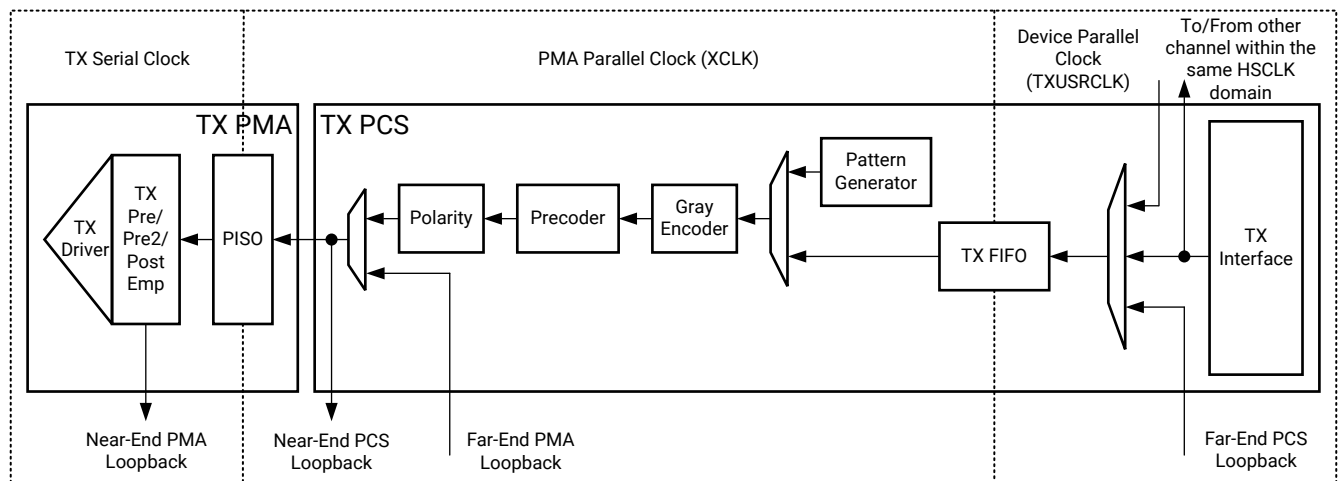
Figure 34: **TXOUTCLK Drives TXUSRCLK**



TX FIFO

In the transceiver TX datapath, the TX FIFO acts as a buffer between the two clock domains: the fabric (TXUSRCLK), and the PMA parallel clock (XCLK). To transmit data, the TX FIFO provides data width conversion between these clock domains when necessary, depending on the operating data width and encoding mode. The following figure shows the TX datapath clock domains.

Figure 35: Transmitter Datapath Clock Domains



X25493-062321

The GTM transmitter includes a TX FIFO to support data width conversion when data crosses from TXUSRCLK to XCLK domain. The buffer does not tolerate ppm differences, and only provides phase compensation between the two clocks. The TX FIFO inside the GTM transceiver must always be used and cannot be bypassed.

Using the TX FIFO

Reset the TX FIFO whenever CH*_TXBUFSTATUS indicates an overflow or underflow condition. The TX FIFO can be reset by using the TX reset procedure or PCS component reset described in [TX Initialization and Reset](#).

Reading TX FIFO Latency

The datapath latency through the TX FIFO is calculated statistically using TXLATCLK, which is asynchronous to XCLK. TX_SAMPLE_PERIOD in CH*_TX_PCS_CFG1 determines the number of TXLATCLK cycles over which averaging takes place. TXLATCLK needs to be asynchronous in both phase and frequency to both the write clock and the read clock. The TXLATCLK has the same range as the TXUSRCLK — greater than zero frequency and less than the maximum USRCLK frequency. The measured latency value in TX_FIFO_LATENCY is updated once per sampling period, and located in COE_STATUS_TX_GB_DBG6 for the 512x128 FIFO and COE_STATUS_TX_GB_DBG8 for the 320x128 FIFO.

These settings are used to read the latency:

- Set CH*_TX_PCS_CFG1[17:15] (TX_SAMPLE_PERIOD)
 - A higher averaging period gives a more accurate latency value.
- For the 512x128 FIFO, read CH*_COE_STATUS_TX_GB_DBG6[17:0] (TX_FIFO_LATENCY):
The value is in units of 1/8 UI.
- For the 320x128 FIFO, read CH*_COE_STATUS_TX_GB_DBG8[17:0] (TX_FIFO_LATENCY):
The value is in units of 1/8 UI.
- The actual latency is TX_FIFO_LATENCY plus a fixed value.

Note: Use the 512x128 FIFO when data widths are 512/256/128/64/32 and the 320x128 FIFO when data widths are 320/160/80/40.

Ports and Attributes

The following table shows the TX FIFO ports.

Table 40: TX FIFO Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_TXBUFSTATUS[2:0]	Output	TXUSRCLK	TX buffer status: Bit [2]: Reserved. Tied High. Bit [1]: FIFO overflow status. A value of 1 indicates FIFO overflow. Bit [0]: FIFO underflow status. A value of 1 indicates FIFO underflow.
CH[0/1/2/3]_TXLATCLK	Input	CLOCK	Input port used to provide a clock for the TX FIFO latency calculation.

The following table shows the TX FIFO attributes.

Table 41: TX FIFO Attributes

TX FIFO Attributes		
Attribute	Address	
CH0_COE_STATUS_TX_GB_DBG6	0x0816	
CH1_COE_STATUS_TX_GB_DBG6	0x0916	
CH2_COE_STATUS_TX_GB_DBG6	0x0A16	
CH3_COE_STATUS_TX_GB_DBG6	0x0B16	
Label	Bit Field	Description
TX_FIFO_LATENCY	[17:0]	Measured latency of the 512x128 TX FIFO averaged over TX_SAMPLE_PERIOD (CH*_TX_PCS_CFG1[17:15]) cycles. Use the 512x128 FIFO when data widths are 512/256/128/64/32. The reported latency is in units of 1/8 UI. The COE_STATUS_TX_GB_DBG6 is read-only.
Attribute	Address	
CH0_COE_STATUS_TX_GB_DBG8	0x0818	
CH1_COE_STATUS_TX_GB_DBG8	0x0918	
CH2_COE_STATUS_TX_GB_DBG8	0x0A18	
CH3_COE_STATUS_TX_GB_DBG8	0x0B18	
Label	Bit Field	Description
TX_FIFO_LATENCY	[17:0]	Measured latency of the 320x128 TX FIFO averaged over TX_SAMPLE_PERIOD (CH*_TX_PCS_CFG1[17:15]) cycles. The reported latency is in units of 1/8 UI. Use the 320x128 FIFO when data widths are 320/160/80/40. The COE_STATUS_TX_GB_DBG8 is read-only.
Attribute	Address	
CH0_TX_PCS_CFG1	0x0C17	
CH1_TX_PCS_CFG1	0x0D17	
CH2_TX_PCS_CFG1	0x0E17	
CH3_TX_PCS_CFG1	0x0F17	
Label	Bit Field	Description
TX_SAMPLE_PERIOD	[17:15]	Number of CH*_TXLATCLK cycles over which averaging take place for latency calculation: 3'b000: 256 3'b001: 512 3'b010: 1024 3'b011: 2048 3'b100: 4096 3'b101: 8192 3'b110: 16384 3'b111: 32768

TX Pattern Generator

Pseudo-random sequences (PRBS) are commonly used to test the signal integrity of high-speed links. These sequences appear random but have specific properties that can be used to measure the quality of a link. The GTM transceiver pattern generator block can generate several industry-standard PRBS patterns listed in the following table.

Table 42: Supported PRBS Patterns

Name	Polynomial	Length of Sequence	Description
PRBS-7	$1 + x^6 + x^7$	$2^7 - 1$ bits	Used to test channels with 8B/10B.
PRBS-9	$1 + x^5 + x^9$	$2^9 - 1$ bits	ITU-T Recommendation O.150, Section 5.1. PRBS-9 is one of the recommended test patterns for SFP+.
PRBS-13	$1 + x + x^2 + x^{12} + x^{13}$	$2^9 - 1$ bits	IEEE Std P802.3bs 120.5.11.2.1 test requires PRBS13Q test pattern. OIF2014.230 CEI-56G-VSR-PAM4 specification requires QPRBS13-CEI test pattern.
PRBS-15	$1 + x^{14} + x^{15}$	$2^{15} - 1$ bits	PRBS-15 is often used for jitter measurement because it is the longest pattern the Agilent DCA-J sampling scope can handle.
PRBS-23	$1 + x^{18} + x^{23}$	$2^{23} - 1$ bits	PRBS-23 is often used for non-8B/10B encoding schemes. It is one of the recommended test patterns in the SONET specification.
PRBS-31	$1 + x^{28} + x^{31}$	$2^{31} - 1$ bits	PRBS-31 is often used for non-8B/10B encoding schemes. It is a recommended PRBS test pattern for 10 Gigabit Ethernet. The pattern generator output does not include the inverter specified by IEEE802.3-2018 for PRBS31.



IMPORTANT! For PAM4 modulation, QPRBS/PRBSQ patterns are supported by sending a conventional PRBS pattern with PAM4 and Gray Coding based on the OIF2014.230 CEI-56G-VSR-PAM4 specification and IEEE Std P802.3bs. The PRBSQ pattern, as defined in IEEE 802.3bs-2017 120.5.11.2.1, is a Gray-coded PRBS pattern repeated twice. It is equivalent to the QPRBS-CEI pattern as defined in OIF-CEI-04.0 CEI-56G-VSR-PAM4 Clause 16. QPRBS, as defined in IEEE 802.3-2015 94.2.9.3, is PRBS followed by inverted PRBS. The A_TX/RXQPRBSEN is used to enable QPRBS.

The PRBS31 pattern is not IEEE802.3-2018 compliant due to non-inverted output. This means to generate an OIF/IEEE Std 802.3 compliant PRBS31Q pattern, a soft generator that can provide an IEEE Std 802.3-2018 compliant PRBS31 must be used prior to Gray encoding. One such PRBS generator and checker is demonstrated in *An Attribute-Programmable PRBS Generator and Checker* ([XAPP884](#)).

The following table describes the relationship between QPRBSEN, Gray Coding bypass, and the associated data pattern.

Table 43: PRBS Patterns for PAM4

TXQPRBSEN	TX_GRAY_BYP_EN	PRBS Pattern
0	0	PRBSQ

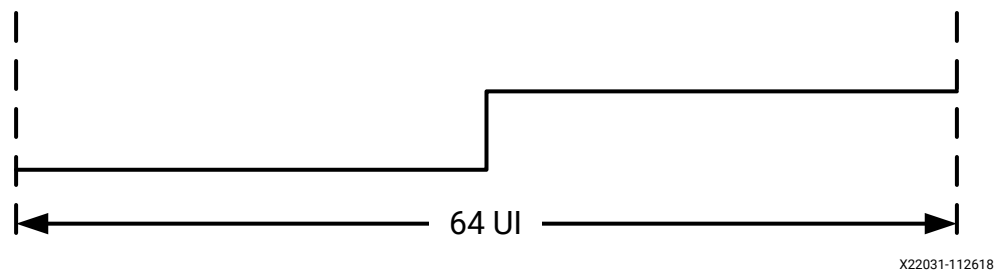
Table 43: PRBS Patterns for PAM4 (cont'd)

TXQPRBSEN	TX_GRAY_BYP_EN	PRBS Pattern
0	1	PRBS
1	1	QPRBS

In addition to PRBS patterns, the GTM transceiver supports a 64 UI square wave test pattern as shown in the following figure, an alternating 1'b0 and 1'b1 (NRZ clock) test pattern. Clocking patterns are usually used to check PLL random jitter often done with a spectrum analyzer.

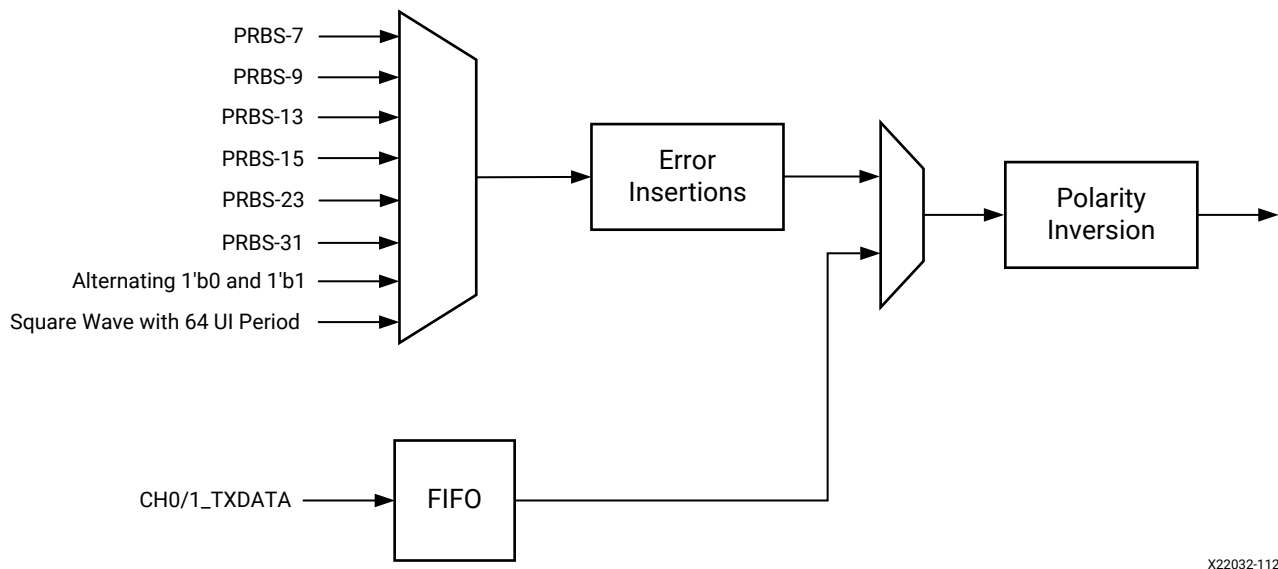
Note: For PAM4 modulation, an alternating 1'b0 and 1'b1 test pattern will not be a square wave due to the amplitude modulation mapping.

Figure 36: 64 UI Square Wave



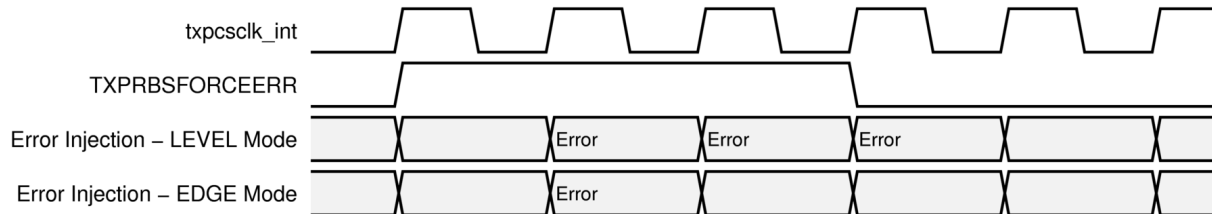
The error insertion function is also supported to verify link connection for jitter tolerance tests. When an inverted PRBS pattern is necessary, the CH[0/1]_TXPOLARITY signal is used to control polarity.

Figure 37: TX Pattern Generator Block



The pattern generator might control when an injection of an error in the generated pattern occurs by choosing between two modes. In LEVEL mode, the generator injects an error after any cycle in which CH*_TXPRBSFORCEERR is High. In EDGE mode, an error is injected only after CH*_TXPRBSFORCEERR has just gone High after being Low in the previous cycle. The mode selection can be done using the TX_PRBS_FORCE_MODE attribute. The following figure shows the error injection in both operating modes.

Figure 38: LEVEL and EDGE Mode Error Injection



Ports and Attributes

The following table shows the TX pattern generator ports.

Table 44: TX Pattern Generator Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_TXPRBSFORCEERR	Input	ASYNC	When this port is driven High, a single error is forced in the PRBS transmitter for every TXUSRCLK clock cycle that the port is asserted. While this port is asserted, the output data pattern contains one error for every TXUSRCLK clock cycle. When TXPRBSSEL is set to 4'b0000, this port does not affect TXDATA.
CH[0/1/2/3]_TXPRBSSEL[3:0]	Input	ASYNC	Transmitter PRBS generator test pattern control. 4'b0000: Standard operation mode (test pattern generation is off) 4'b0001: PRBS-7 4'b0010: PRBS-9 4'b0011: PRBS-15 4'b0100: PRBS-23 4'b0101: PRBS-31 4'b0110: PRBS-13 4'b1000: User configurable data pattern. 4'b1001: Square wave with 2 UI (alternating 0s/1s) 4'b1010: Square wave with 16 UI, 20 UI, 32 UI, 40 UI, 64 UI, or 80 UI period (based on internal data width).
CH[0/1/2/3]_TXQPRBSEN	Input	ASYNC	TX QPRBS Enable - Used along with the TXPRBSSEL. Not supported for PRBS7

The following table shows the pattern generator attributes.

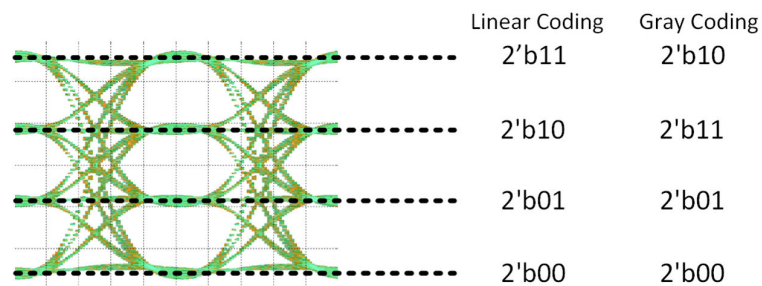
Table 45: TX Pattern Generator Attributes

TX Pattern Generator Attributes		
Attribute	Address	
CH0_TX_PCS_CFG0	0x0C16	
CH1_TX_PCS_CFG0	0x0D16	
CH2_TX_PCS_CFG0	0x0E16	
CH3_TX_PCS_CFG0	0x0F16	
Label	Bit Field	Description
TX_PRBS_FORCE_MODE	[29:29]	Pattern Generator error operation mode 0: Level mode 1: Edge mode
RXPRBSERR_LOOPBACK	[23:23]	Enable RXPRBSERR error injection: 0: Enable TXPRBSFORCEERR forces onto the TX PRBS 1: Enable RXPRBSERR bit to be internally looped back to TXPRBSFORCEERR

TX Gray Encoder

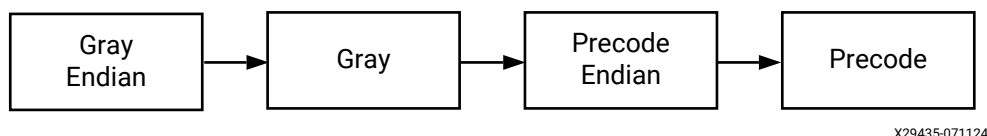
The GTM transmitter supports two types of binary encoding options, linear coding and Gray coding. By using gray coding, only one bit error per symbol is made for incorrect decisions, thus reducing the bit-error rate by more than 33%. The following figure illustrates the differences between linear coding and Gray coding.

Figure 39: Transmitted PAM4 Signal Bit Encoding



The TX gray encoder has independentendianness control blocks. The following figure shows the sequence in which bits are flipped.

Figure 40: TX Gray Encoder Bit Flip Sequence



Ports and Attributes

The following table shows the TX gray encoder attributes.

Table 46: TX Gray Encoder Attributes

TX Gray Encoder Attributes		
Attribute	Address	
CH0_TX_PCS_CFG0	0x0C16	
CH1_TX_PCS_CFG0	0x0D16	
CH2_TX_PCS_CFG0	0x0E16	
CH3_TX_PCS_CFG0	0x0F16	
Label	Bit Field	Description
TX_GRAY_ENDIAN	[13:13]	Controls TX Gray Code Endianness: 0: Big endian 1: Little endian
TX_GRAY_BYP_EN	[12:12]	Controls TX Gray Code: 0: Enable TX Gray Code 1: Bypass TX Gray Code

TX Pre-Coder

The GTM transmitter supports pre-coding. Pre-coding can be used to reduce receiver decision feedback equalization (DFE) error propagation by reducing one-tap burst error runs into two errors for every error event.

Ports and Attributes

The following table shows the TX Pre-Coder attributes.

Table 47: TX Pre-Coder Attributes

TX Pre-Coder Attributes	
Attribute	Address
CH0_TX_PCS_CFG0	0x0C16

Table 47: TX Pre-Coder Attributes (cont'd)

TX Pre-Coder Attributes		
CH1_TX_PCS_CFG0		0x0D16
CH2_TX_PCS_CFG0		0x0E16
CH3_TX_PCS_CFG0		0x0F16
Label	Bit Field	Description
TX_PRECODE_ENDIAN	[11:11]	Controls TX Pre-Coder Endianness: 0: Big endian 1: Little endian
TX_PRECODE_BYP_EN	[10:10]	Controls TX Pre-Code: 0: Enable TX Pre-Code 1: Bypass TX Pre-Code

TX Polarity Control

The GTM transceiver includes a TX polarity control function to invert outgoing data from the PCS before serialization and transmission. If TXP and TXN differential traces are accidentally swapped on the PCB, this block can be used to reverse the GTM output polarity. The TX polarity control can be accessed through the CH*_TXPOLARITY input from the interconnect logic interface. It is driven High to invert the polarity of the outgoing data.

Ports and Attributes

The following table shows the TX polarity control ports.

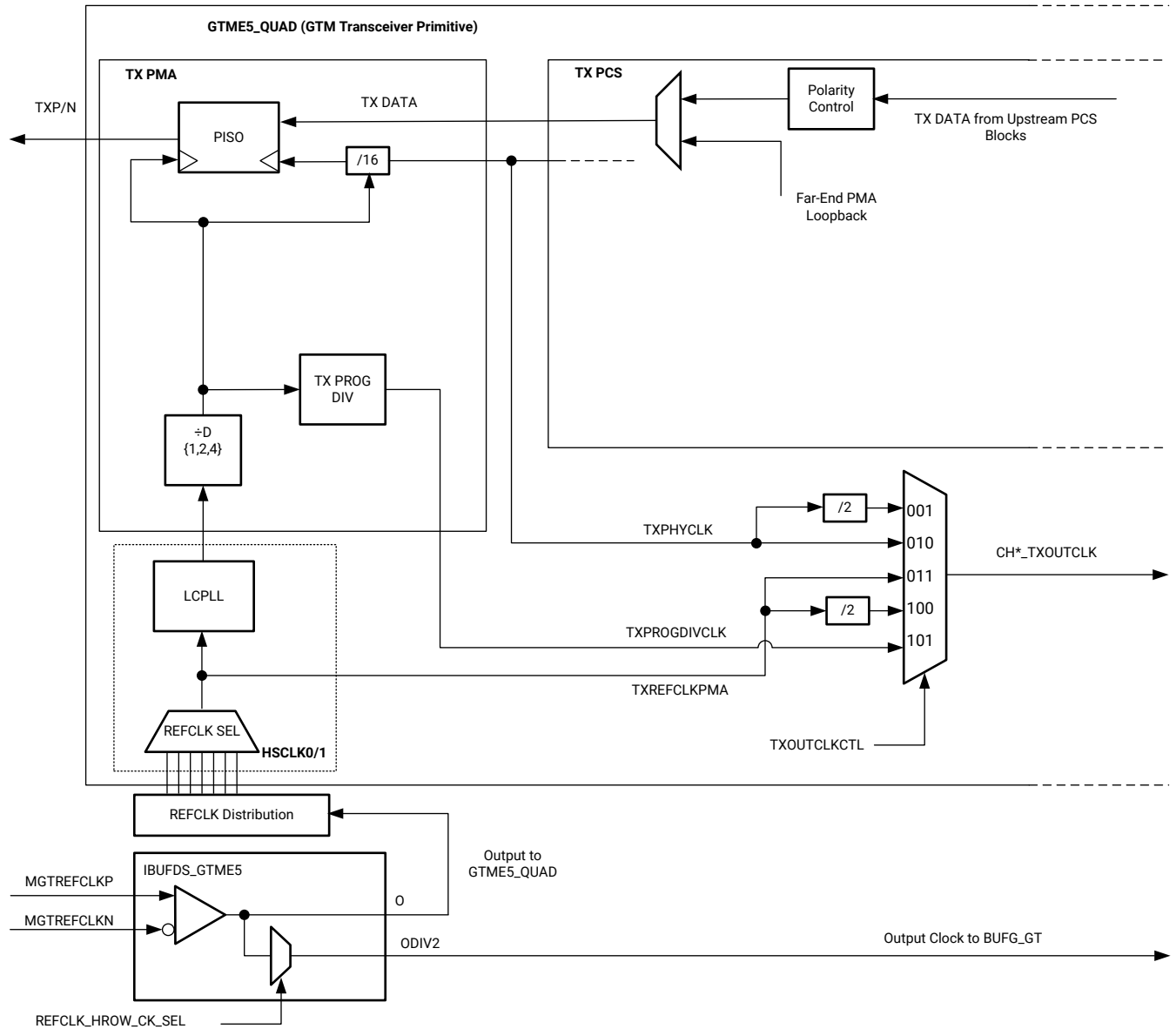
Table 48: TX Polarity Control Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_TXPOLARITY	Input	ASYNC	This port is used to invert the polarity of outgoing data. 0: Not inverted. TXP is positive, and TXN is negative. 1: Inverted. TXP is negative, and TXN is positive.

TX Fabric Clock Output Control

The TX Clock Divider Control block has two main components: serial clock divider control and parallel clock divider and selector control. The clock divider and selector details are illustrated in the following figure.

Figure 41: TX Serial and Parallel Clock Divider



X25594-072321

Note:

1. CH*_TXOUTCLK is used as the source of the interconnect logic clock via BUFG_GT.
2. For details about placement constraints and restrictions on clocking resources (such as BUFG_GT and BUFG_GT_SYNC), refer to the *Versal Adaptive SoC Clocking Resources Architecture Manual* (AM003).
3. The clock output from IBUFDS_GTME5 should only be used after GTPowerGood asserts High.

Serial Clock Divider

Each transmitter PMA module has a D divider that divides down the clock from the PLL for lower line rate support. This serial clock divider, D, can be set statically for applications with a fixed line rate or it can be changed dynamically for protocols with multiple line rates.

To use the D divider in fixed line rate applications, TXOUT_DIV must be set to the appropriate value, and the CH*_TXRATE port should be tied to 8'b00000000.

For multiple line rate applications, the CH*_TXRATE port is used to dynamically select the line rate settings which include the appropriate divider values. See [Rate Change](#) for more details.

Parallel Clock Divider and Selector

The parallel clock outputs from the TX clock divider control block can be used as an interconnect logic clock, depending on the line rate requirement.

The recommended clock for the interconnect logic is the CH*_TXOUTCLK from one of the GTM transceivers. It is also possible to bring the MGTREFCLK directly to the interconnect logic and use it as the interconnect logic clock. CH*_TXOUTCLK is preferred for general applications.

TXOUTCLKCTL controls the input selector and allows these clocks to be output via the CH*_TXOUTCLK port:

- 3'b001: The divide-by-two version of TXPHYCLK.
- 3'b010: TXPHYCLK is the divided down PLL clock after the TX PISO and is used by the TX PCS block. This clock is interrupted when the PLL is reset by one of the related reset signals.
- 3'b011: TXREFCLKPMA is the input reference clock to the LCPLL. TXREFCLKPMA is the recommended clock for general usage.
- 3'b100: The divide-by-two version of TXREFCLKPMA.
- 3'b101: TXPROGDIVCLK is the divided down PLL clock after the TX programmable divider. See [TX Programmable Divider](#) for more details.

TX Programmable Divider

The TX programmable divider shown in [TX Fabric Clock Output Control](#) uses the PLL output clock to generate a parallel output clock. By using the transceiver PLL, TX programmable divider, and BUFG_GT, CH*_TXOUTCLK (TXOUTCLKCTL = 3'b101) can be used as a clock source for the interconnect logic. The supported divider values are 4, 5, 5.5, 8, 10, 16, 16.5, 20, 32, 33, 40, 64, and 66.

Ports and Attributes

The following table defines the ports required for TX fabric clock output control.

Table 49: TX Fabric Clock Output Control Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_TXOUTCLK	Output	CLOCK	TXOUTCLK is the recommended clock output to the interconnect logic.
CH[0/1/2/3]_TXPROGDIVRESET	Input	ASYNC	This active-High port resets the dividers as well as the CH*_TXPROGDIVRESETDONE indicator. A reset must be performed whenever the input clock source is interrupted.
CH[0/1/2/3]_TXPROGDIVRESETDONE	Output	ASYNC	When the input clock is stable and reset is performed, this active-High signal indicates the reset is completed and the output clock is stable.
CH[0/1/2/3]_TXRATE[7:0]	Input	ASYNC	This port is used to perform rate change on the transceiver TX. The Wizard will preconfigure a list of desired line rates, and this port will be used to dynamically adjust the running line rate based on the preconfigured list. Set this port to the matching preconfigured line rate option value to obtain the proper line rate.

The following table defines the attributes required for TX fabric clock output control.

Table 50: TX Fabric Clock Output Control Attributes

TX Fabric Clock Output Control Attributes		
Attribute	Address	
CH0_PMA_MISC_CFG0	0x0C27	
CH1_PMA_MISC_CFG0	0x0D27	
CH2_PMA_MISC_CFG0	0x0E27	
CH3_PMA_MISC_CFG0	0x0F27	
Label	Bit Field	Description
TXOUTCLKCTL	[24:22]	This attribute selects the source for CH*_TXOUTCLK. 3'b001: Div2 version of TXPHYCLK 3'b010: TXPHYCLK 3'b011: TXREFCLKPMA 3'b100: Div2 version of TXREFCLKPMA 3'b101: TXPROGDIVCLK
Attribute	Address	
CH0_TX_CTRL_CFG3	0x0C37	
CH1_TX_CTRL_CFG3	0x0D37	
CH2_TX_CTRL_CFG3	0x0E37	

Table 50: TX Fabric Clock Output Control Attributes (cont'd)

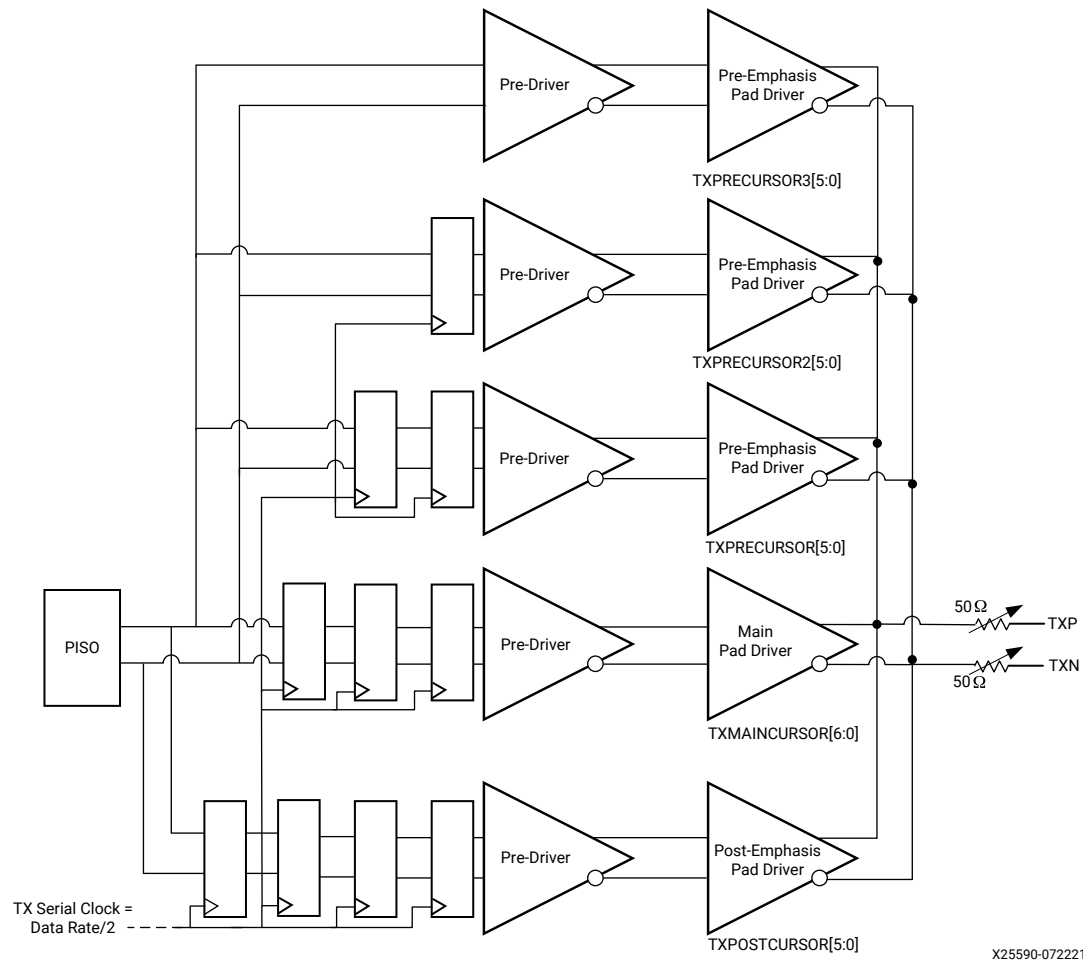
TX Fabric Clock Output Control Attributes		
CH3_TX_CTRL_CFG3	0x0F37	
Label	Bit Field	Description
TXOUT_DIV	[14:13]	This attribute selects the setting for the TX serial clock divider D. 3'b000: /1 3'b001: /2 3'b010: /4
TXPROGDIVSEL	[9:0]	TX programmable divider ratio. Valid settings are 4, 5, 5.5, 8, 10, 16, 16.5, 20, 32, 33, 40, 64, and 66. 10'b1001011000: /4 10'b1001111000: /5 10'b1110011000: /5.5 10'b0001011000: /8 10'b1001000000: /8 (Fullrate) 10'b0001111000: /10 10'b1001100000: /10 (Fullrate) 10'b0001000000: /16 10'b1001000010: /16 (Fullrate) 10'b1100011000: /16.5 10'b0001100000: /20 10'b1001100010: /20 (Fullrate) 10'b0001000010: /32 10'b1001000110: /32 (Fullrate) 10'b0100011000: /33 10'b1100000000: /33 (Fullrate) 10'b0001100010: /40 10'b1001100110: /40 (Fullrate) 10'b1001001110: /64 10'b1001001110: /64 (Fullrate) 10'b0100000000: /66 10'b1100000010: /66 (Fullrate) 10'b0001100110: /80 10'b1001101110: /80 (Fullrate)

TX Configurable Driver

The GTM transceiver TX driver is a high-speed current-mode differential output buffer. To maximize signal integrity, it includes these features:

- Differential voltage control
- Three pre-cursor and one post-cursor transmit pre-emphasis controls
- Two modulation schemes: NRZ and PAM4
- Calibrated termination resistors

Figure 42: TX Configurable Driver Block Diagram



TX Driver Swing Control

Driver swing can be controlled through CH*_TXMAINCURSOR[6:0]. The default is user specified. All values listed below are in mV_{PDD} and are typical values.

Table 51: Transmitter Swing Control

[6:0]	Swing (mV _{PDD})	Coefficient Units
7'b0101010	477	42
7'b0101011	488	43
7'b0101100	498	44
7'b0101101	508	45
7'b0101110	518	46
7'b0101111	526	47
7'b0110000	536	48
7'b0110001	545	49

Table 51: Transmitter Swing Control (cont'd)

[6:0]	Swing (mV _{PDD})	Coefficient Units
7'b0110010	554	50
7'b0110011	564	51
7'b0110100	572	52
7'b0110101	581	53
7'b0110110	590	54
7'b0110111	598	55
7'b0111000	606	56
7'b0111001	615	57
7'b0111010	624	58
7'b0111011	654	59
7'b0111100	663	60
7'b0111101	672	61
7'b0111110	681	62
7'b0111111	690	63
7'b1000000	700	64
7'b1000001	709	65
7'b1000010	719	66
7'b1000011	727	67
7'b1000100	738	68
7'b1000101	748	69
7'b1000110	756	70
7'b1000111	765	71
7'b1001000	774	72
7'b1001001	784	73
7'b1001010	793	74
7'b1001011	801	75
7'b1001100	812	76
7'b1001101	819	77
7'b1001110	828	78
7'b1001111	835	79
7'b1010000	842	80
7'b1010001	849	81
7'b1010010	857	82
7'b1010011	863	83
7'b1010100	872	84
7'b1010101	878	85
7'b1010110	887	86

Table 51: Transmitter Swing Control (cont'd)

[6:0]	Swing (mV _{PDD})	Coefficient Units
7'b1010111	902	87

Notes:

1. The peak-to-peak differential voltage is defined when CH*_TXPOSTCURSOR = 5'b000000, CH*_TXPRECUSOR = 5'b000000, CH*_TXPRECUSOR2 = 5'b000000, and CH*_TXPRECUSOR3 = 5'b000000.
2. The output swing described above is obtained using settings from the Wizard design, and the recommended values from the Wizard should not be changed.
3. TXSWING = TXMAINCUSOR + TXPOSTCURSOR + TXPRECUSOR + TXPRECUSOR2 + TXPRECUSOR3 ≤ 86. When no post-cursor or pre-cursors are set, TXMAINCUSOR is equal to TXSWING.
4. In independent cursor mode, TXMAINCUSOR is used to set the main cursor value. The IBERT tool does not operate in independent cursor mode, but uses the default non-independent cursor mode instead. In the default non-independent cursor mode, TXMAINCUSOR is used to set the cursor sum (or TXSWING in the previous note). The main cursor value is then calculated by subtracting pre-cursor and post-cursor values from the cursor sum.

TX Driver Post-Cursor Control

Driver swing can be controlled through CH*_TXPOSTCURSOR[5:0]. The default is user specified. All values listed below are in dB and are typical values.

Table 52: Transmitter Post-Cursor TX Pre-Emphasis Control

[5:0]	Emphasis (dB)	Coefficient Units
6'b0000000	0.00	0
6'b0000001	-0.23	1
6'b0000010	-0.41	2
6'b0000011	-0.62	3
6'b0000100	-0.79	4
6'b0000101	-1.01	5
6'b0000110	-1.24	6
6'b0000111	-1.50	7
6'b0001000	-1.74	8
6'b0001001	-1.98	9
6'b0001010	-2.22	10
6'b0001011	-2.50	11
6'b0001100	-2.76	12
6'b0001101	-3.01	13
6'b0001110	-3.29	14
6'b0001111	-3.56	15
6'b0010000	-3.86	16
6'b0010001	-4.17	17
6'b0010010	-4.47	18
6'b0010011	-4.82	19
6'b0010100	-5.15	20

Table 52: Transmitter Post-Cursor TX Pre-Emphasis Control (cont'd)

[5:0]	Emphasis (dB)	Coefficient Units
6'b0010101	-5.54	21
6'b0010110	-5.94	22
6'b0010111	-6.41	23
6'b0011000	-6.86	24
6'b0011001	-7.92	25
6'b0011010	-8.43	26
6'b0011011	-8.92	27
6'b0011100	-9.46	28
6'b0011101	-10.03	29
6'b0011110	-10.61	30
6'b0011111	-11.27	31
6'b0100000	-12.04	32
6'b0100001	-12.87	33
6'b0100010	-13.78	34
6'b0100011	-14.81	35
6'b0100100	-15.97	36

Notes:

- The above values are defined when CH*_TXPRECURSOR = 6'b000000, CH*_TXPRECURSOR2 = 6'b000000, and CH*_TXPRECURSOR3 = 6'b000000
- Emphasis = $20\log_{10}(V_{\text{high}}/V_{\text{low}}) = |20\log_{10}(V_{\text{low}}/V_{\text{high}})|$

TX Driver Pre-Cursor Control

Driver swing can be controlled through CH*_TXPRECURSOR[5:0]. The default is user specified. All values listed below are in dB and are typical values.

Table 53: Transmitter Pre-Cursor TX Pre-Emphasis Control

[5:0]	Emphasis (dB)	Coefficient Units
6'b0000000	0	0
6'b0000001	-0.24	1
6'b0000010	-0.44	2
6'b0000011	-0.65	3
6'b0000100	-0.86	4
6'b0000101	-1.08	5
6'b0000110	-1.31	6
6'b0000111	-1.52	7
6'b0001000	-1.75	8
6'b0001001	-2.02	9
6'b0001010	-2.25	10

Table 53: Transmitter Pre-Cursor TX Pre-Emphasis Control (cont'd)

[5:0]	Emphasis (dB)	Coefficient Units
6'b0001011	-2.49	11
6'b0001100	-2.75	12
6'b0001101	-3.03	13
6'b0001110	-3.31	14
6'b0001111	-3.55	15
6'b0010000	-3.85	16
6'b0010001	-4.16	17
6'b0010010	-4.48	18
6'b0010011	-4.81	19
6'b0010100	-5.17	20
6'b0010101	-5.55	21
6'b0010110	-5.97	22
6'b0010111	-6.43	23
6'b0011000	-6.87	24
6'b0011001	-7.94	25
6'b0011010	-8.43	26
6'b0011011	-8.93	27
6'b0011100	-9.43	28
6'b0011101	-9.98	29
6'b0011110	-10.57	30
6'b0011111	-11.30	31
6'b0100000	-12.12	32
6'b0100001	-12.84	33
6'b0100010	-13.71	34
6'b0100011	-14.75	35
6'b0100100	-15.76	36

Notes:

- The above values are defined when CH*_TXPOSTCURSOR = 6'b000000, CH*_TXPRECURSOR2 = 6'b000000, and CH*_TXPRECURSOR3 = 6'b000000.
- Emphasis = $20\log_{10}(V_{\text{high}}/V_{\text{low}}) = |20\log_{10}(V_{\text{low}}/V_{\text{high}})|$

TX Driver Pre-Cursor 2 Control

Driver swing can be controlled through CH*_TXPRECURSOR2[5:0]. The default is user specified. All values listed below are in dB and are typical values.

Table 54: Transmitter Pre-Cursor 2 TX Pre-Emphasis Control

[5:0]	Emphasis (dB)	Coefficient Units
6'b0000000	0.00	0

Table 54: Transmitter Pre-Cursor 2 TX Pre-Emphasis Control (cont'd)

[5:0]	Emphasis (dB)	Coefficient Units
6'b0000001	-0.24	1
6'b0000010	-0.44	2
6'b0000011	-0.67	3
6'b0000100	-0.89	4
6'b0000101	-1.11	5
6'b0000110	-1.35	6
6'b0000111	-1.58	7
6'b0001000	-1.81	8
6'b0001001	-2.06	9
6'b0001010	-2.30	10
6'b0001011	-2.57	11
6'b0001100	-2.80	12
6'b0001101	-3.08	13
6'b0001110	-3.32	14
6'b0001111	-3.59	15
6'b0010000	-3.86	16
6'b0010001	-4.16	17
6'b0010010	-4.47	18
6'b0010011	-4.80	19
6'b0010100	-5.14	20
6'b0010101	-5.56	21
6'b0010110	-5.99	22
6'b0010111	-6.35	23
6'b0011000	-6.78	24

Notes:

1. The above values are defined when CH*_TXPOSTCURSOR = 6'b000000, CH*_TXPRECURSOR = 6'b000000, and CH*_TXPRECURSOR3 = 6'b000000.
2. $\text{Emphasis} = 20\log_{10}(V_{\text{high}}/V_{\text{low}}) = |20\log_{10}(V_{\text{low}}/V_{\text{high}})|$
3. Pre-Cursor 2 can be used as de-emphasis by setting fir_pr2_inv = 1 in attribute control CH*_TX_CTRL_CFG0.

TX Driver Pre-Cursor 3 Control

Driver swing can be controlled through CH*_TXPRECURSOR3[5:0]. The default is user specified. All values listed below are in dB and are typical values.

Table 55: Transmitter Pre-Cursor 3 TX Pre-Emphasis Control

[5:0]	Emphasis (dB)	Coefficient Units
6'b0000000	0.00	0
6'b0000001	-0.23	1

Table 55: Transmitter Pre-Cursor 3 TX Pre-Emphasis Control (cont'd)

[5:0]	Emphasis (dB)	Coefficient Units
6'b0000010	-0.45	2
6'b0000011	-0.68	3
6'b0000100	-0.91	4
6'b0000101	-1.14	5
6'b0000110	-1.37	6
6'b0000111	-1.60	7
6'b0001000	-1.83	8
6'b0001001	-2.08	9
6'b0001010	-2.31	10
6'b0001011	-2.56	11
6'b0001100	-2.82	12
6'b0001101	-3.08	13
6'b0001110	-3.35	14
6'b0001111	-3.59	15
6'b0010000	-3.88	16
6'b0010001	-4.18	17
6'b0010010	-4.49	18
6'b0010011	-4.82	19
6'b0010100	-5.18	20
6'b0010101	-5.56	21
6'b0010110	-5.95	22
6'b0010111	-6.32	23
6'b0011000	-6.71	24

Notes:

- The above values are defined when CH*_TXPOSTCURSOR = 6'b000000, CH*_TXPRECURSOR = 6'b000000, and CH*_TXPRECURSOR2 = 6'b000000.
- Emphasis = $20\log_{10}(V_{\text{high}}/V_{\text{low}}) = |20\log_{10}(V_{\text{low}}/V_{\text{high}})|$

Ports and Attributes

The following table shows the TX configurable driver ports and attributes.

Table 56: TX Configurable Driver Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_TXINHIBIT	Input	ASYNC	When High, this signal blocks transmission of TXDATA and forces GTMTXP to 0 and GTMTXN to 1.
CH[0/1/2/3]_TXMAINCURSOR[6:0]	Input	ASYNC	Driver Swing Control. The default is user specified.

Table 56: TX Configurable Driver Ports (cont'd)

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_TXPOSTCURSOR[5:0]	Input	ASYNC	Transmitter post-cursor TX pre-emphasis control. The default is user specified.
CH[0/1/2/3]_TXPRECURSOR[5:0]	Input	ASYNC	Transmitter pre-cursor TX pre-emphasis control. The default is user specified.
CH[0/1/2/3]_TXPRECURSOR2[5:0]	Input	ASYNC	Transmitter pre-cursor2 TX pre-emphasis control. The default is user specified.
CH[0/1/2/3]_TXPRECURSOR3[5:0]	Input	ASYNC	Transmitter pre-cursor3 TX pre-emphasis control. The default is user specified.

Table 57: TX Configurable Driver Attributes

TX Configurable Driver Attributes		
Attribute	Address	
CH0_TX_CTRL_CFG0	0x0C34	
CH1_TX_CTRL_CFG0	0x0D34	
CH2_TX_CTRL_CFG0	0x0E34	
CH3_TX_CTRL_CFG0	0x0F34	
Label	Bit Field	Description
fir_pr2_inv	[22:22]	This attribute can invert the polarity of Pre-Cursor 2. 0: Not inverted (default) 1: Inverted
Attribute	Address	
CH0_FABRIC_INTF_CFG4	0x0C50	
CH1_FABRIC_INTF_CFG4	0x0D50	
CH2_FABRIC_INTF_CFG4	0x0E50	
CH3_FABRIC_INTF_CFG4	0x0F50	
Label	Bit Field	Description
A_TXEMPPRE3	[5:0]	Attribute for TXEMPPRE3.
Attribute	Address	
CH0_FABRIC_INTF_CFG5	0x0C51	
CH1_FABRIC_INTF_CFG5	0x0D51	
CH2_FABRIC_INTF_CFG5	0x0E51	
CH3_FABRIC_INTF_CFG5	0x0F51	
Label	Bit Field	Description
A_TXEMPPRE2	[29:24]	Attribute for TXEMPPRE2.
A_TXEMPPRE	[23:18]	Attribute for TXEMPPRE.
A_TXEMPPPOS	[17:12]	Attribute for TXEMPPPOS.
A_TXEMPMMAIN	[11:5]	Attribute for TXEMPMMAIN.

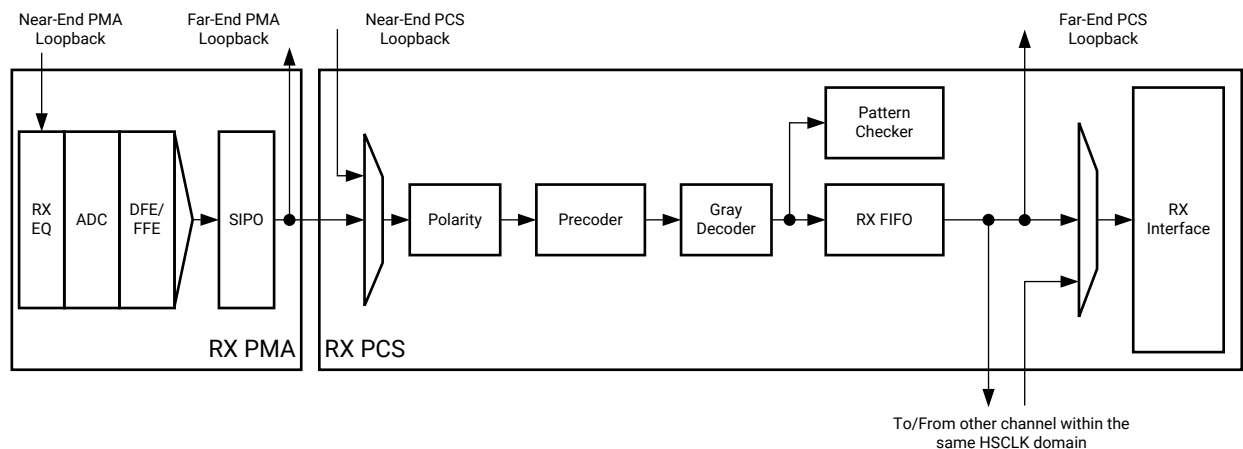
Use Modes

The GTM TX has the ability to receive serial data using two different modulation schemes: NRZ and PAM4. NRZ signals contain one bit of information per symbol while PAM4 signals contain two bits of information per symbol. Using PAM4 modulation doubles the transmitted data bandwidth while maintaining the same unit interval (UI). Use the Versal Adaptive SoC Transceivers Wizard to program the GTM TX to a desired signal modulation. See [Rate Change](#) for modulation changes in runtime.

Receiver

This section shows how to configure and use each of the functional blocks inside the receiver (RX). Each transceiver includes an independent receiver made up of a PCS and a PMA. The following figure shows the blocks of the transceiver RX. High-speed serial data flows from traces on the board into the PMA of the transceiver RX, into the PCS, and finally into the interconnect logic.

Figure 43: GTM Receiver Block Diagram



X23628-013120

The key elements within the transceiver RX are:

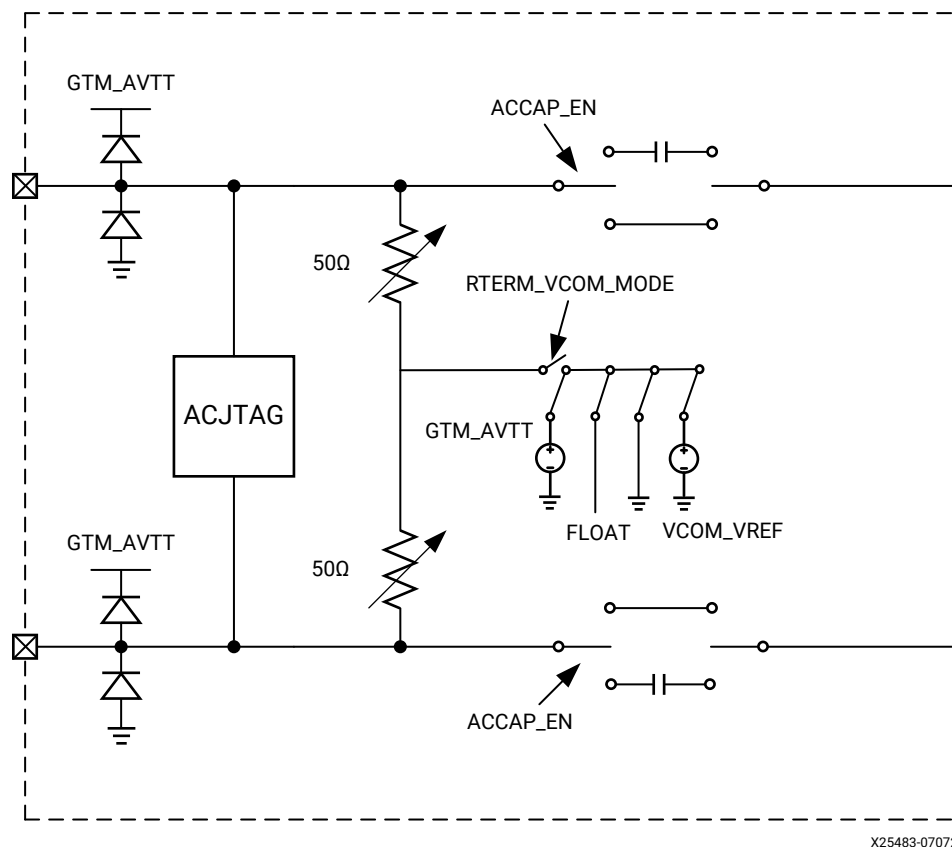
- [RX Analog Front End](#)
- [RX Equalizer](#)
- [RX CDR](#)
- [RX Fabric Clock Output Control](#)
- [RX Margin Analysis](#)
- [RX Polarity Control](#)
- [RX Pre-Coder](#)
- [RX Gray Decoder](#)
- [RX Pattern Checker](#)
- [RX FIFO](#)

- [RX Interface](#)

RX Analog Front End

The RX analog front end (AFE) is a high-speed input differential buffer with configurable RX termination voltage, on-chip coupling capacitors, and calibrated termination resistor. When the on-chip coupling capacitor is used, a level shifter circuit is also enabled to maintain DC balance when receiving unbalanced patterns. The level shifter circuit can also be enabled independently of the on-chip coupling capacitor.

Figure 44: RX Analog Front End



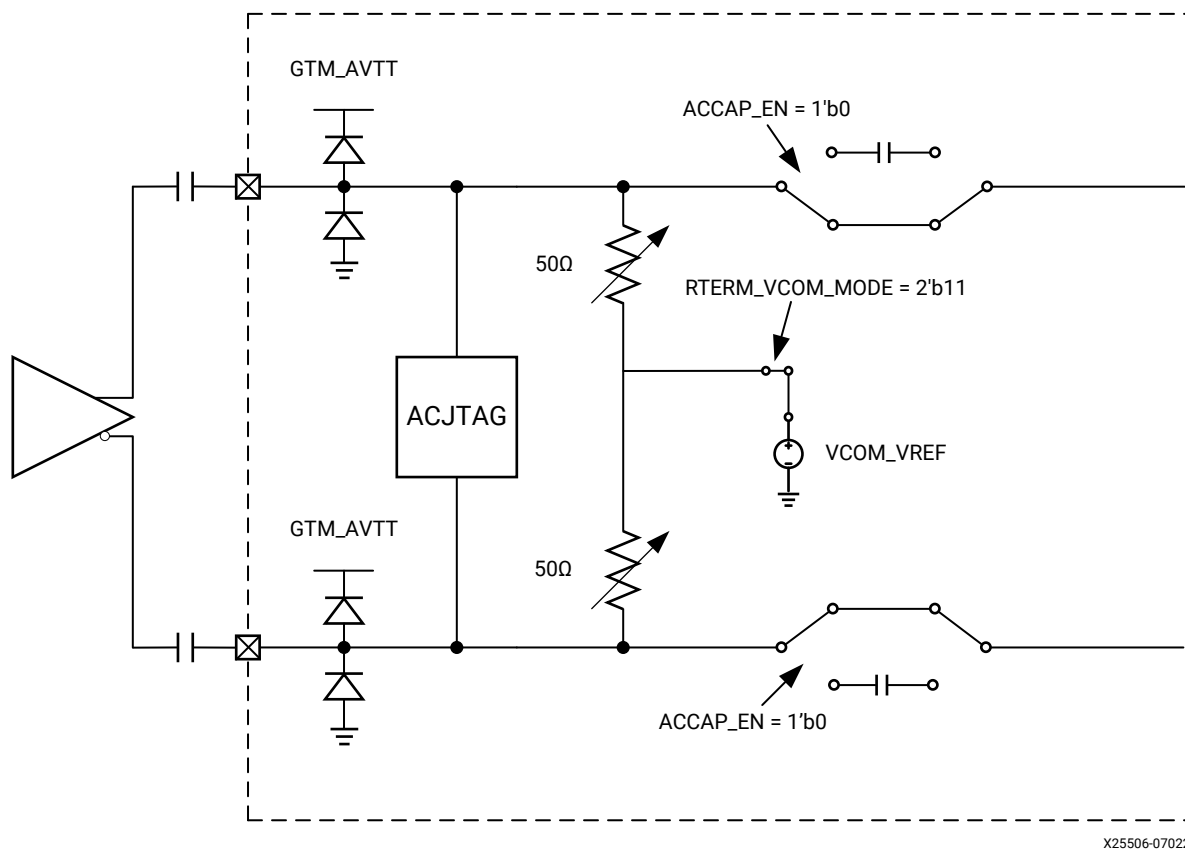
AC Coupling Mode

This is the typical and default transceiver operation mode. In this use mode, the external AC coupling capacitor must be present and should follow data sheet recommendations. The device's internal capacitor is bypassed. The termination voltage is set to **VCOM_VREF** provided by CTLE.

Table 58: AC Coupling Mode Settings

Use Mode	External AC Coupling	Termination Voltage	Setting
AC Coupling	Yes	VCOM_VREF (typically 590 mV)	<ul style="list-style-type: none"> • ACCAP_EN = 1'b0 • LSHIFT_EN = 1'b0 • RTERM_VCOM_MODE = 2'b11

Figure 45: AC Coupling



X25506-070221

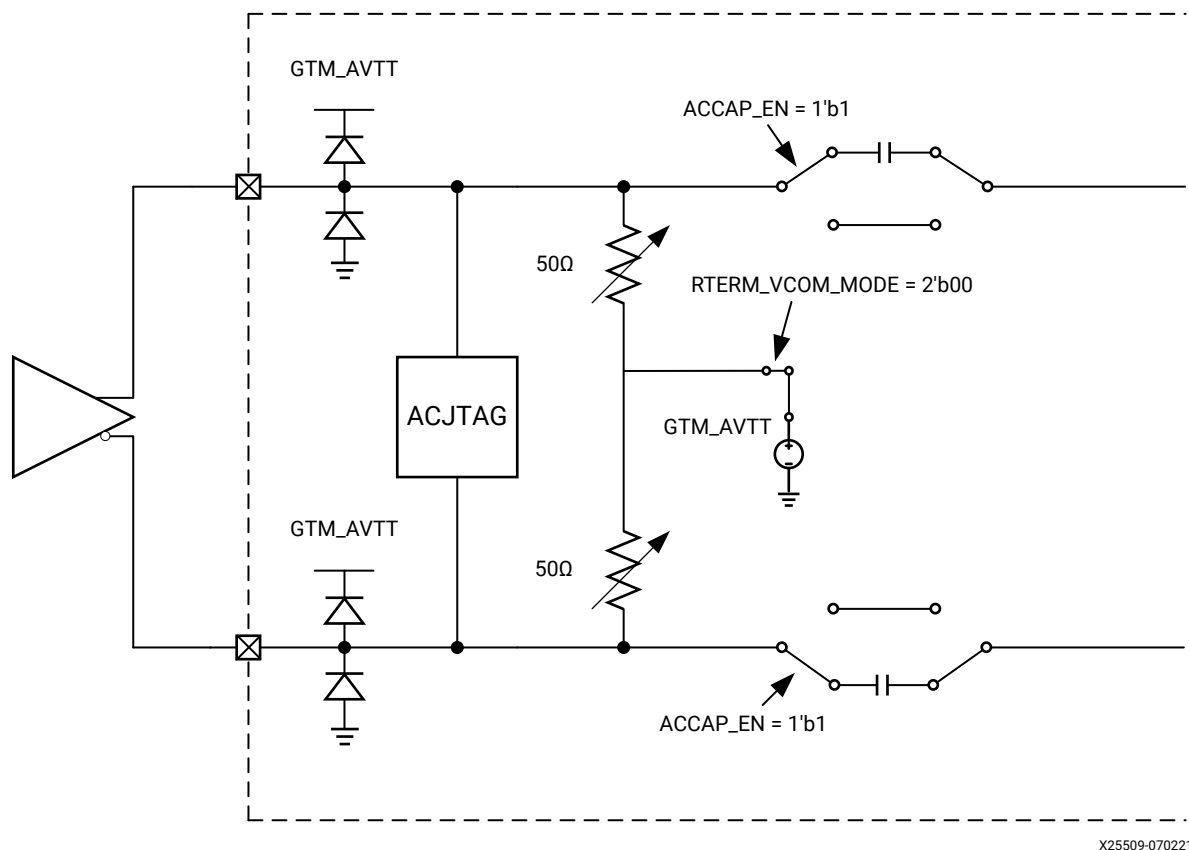
DC Coupling Mode with GTM_AVTT Termination

This mode enables an internal AC capacitor circuit that decouples the analog front end from the equalization block, allowing greater flexibility on the common mode input voltage. The TX or board circuit should ensure that the RX input common mode voltage is set to between 0.6V ~ 1.0V.

Table 59: DC Coupling Mode with GTM_AVTT Termination

Use Mode	External AC Coupling	Termination Voltage	Setting
DC Coupling	No	GTM_AVTT	<ul style="list-style-type: none"> • ACCAP_EN = 1'b1 • LSHIFT_EN = 1'b1 • RTERM_VCOM_MODE = 2'b00

Figure 46: DC Coupling with GTM_AVTT Termination



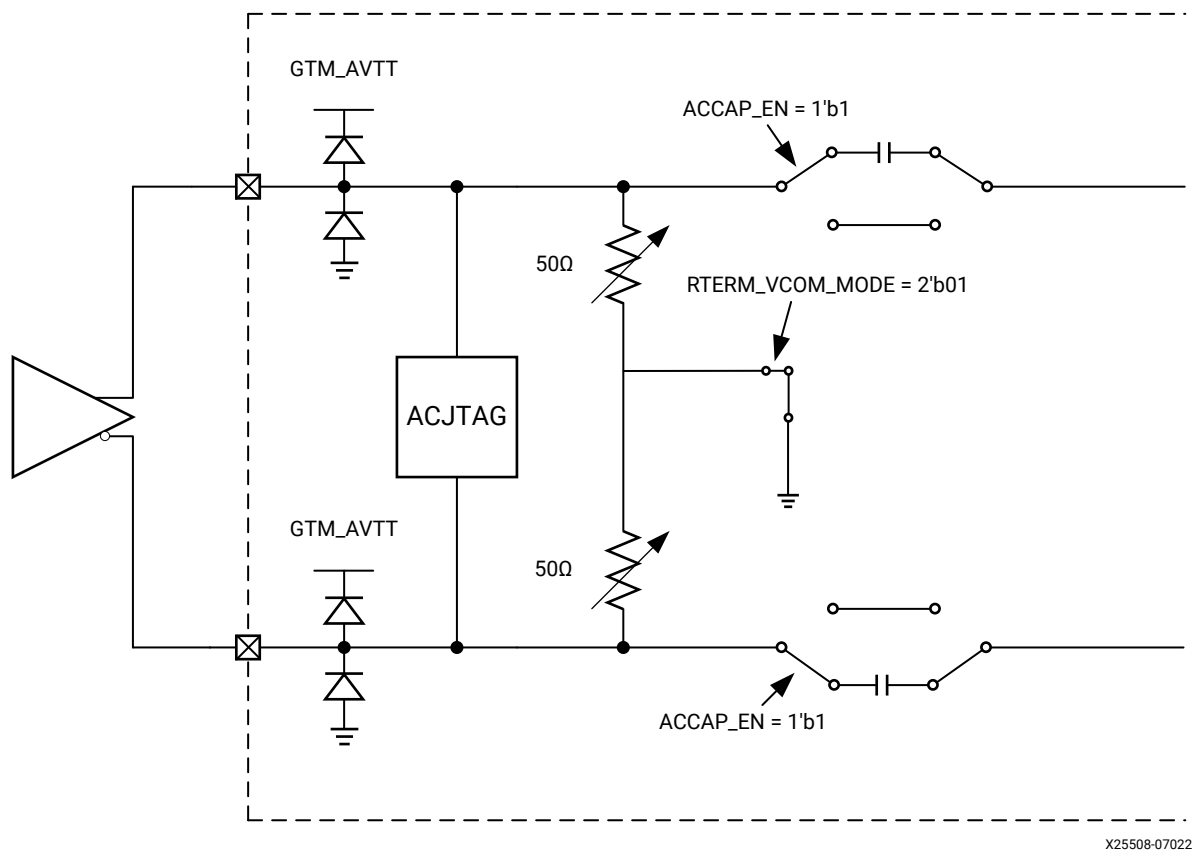
DC Coupling Mode with GND Termination

This mode enables an internal AC capacitor circuit that decouples the analog front end from the equalization block, allowing greater flexibility on the common mode input voltage. The TX or board circuit should ensure that the RX input common mode voltage is set to between 0.2V ~ 0.6V.

Table 60: DC Coupling Mode with GND Termination

Use Mode	External AC Coupling	Termination Voltage	Setting
DC Coupling	No	GND	<ul style="list-style-type: none"> • ACCAP_EN = 1'b1 • LSHIFT_EN = 1'b1 • RTERM_VCOM_MODE = 2'b01

Figure 47: DC Coupling with GND Termination



X25508-070221

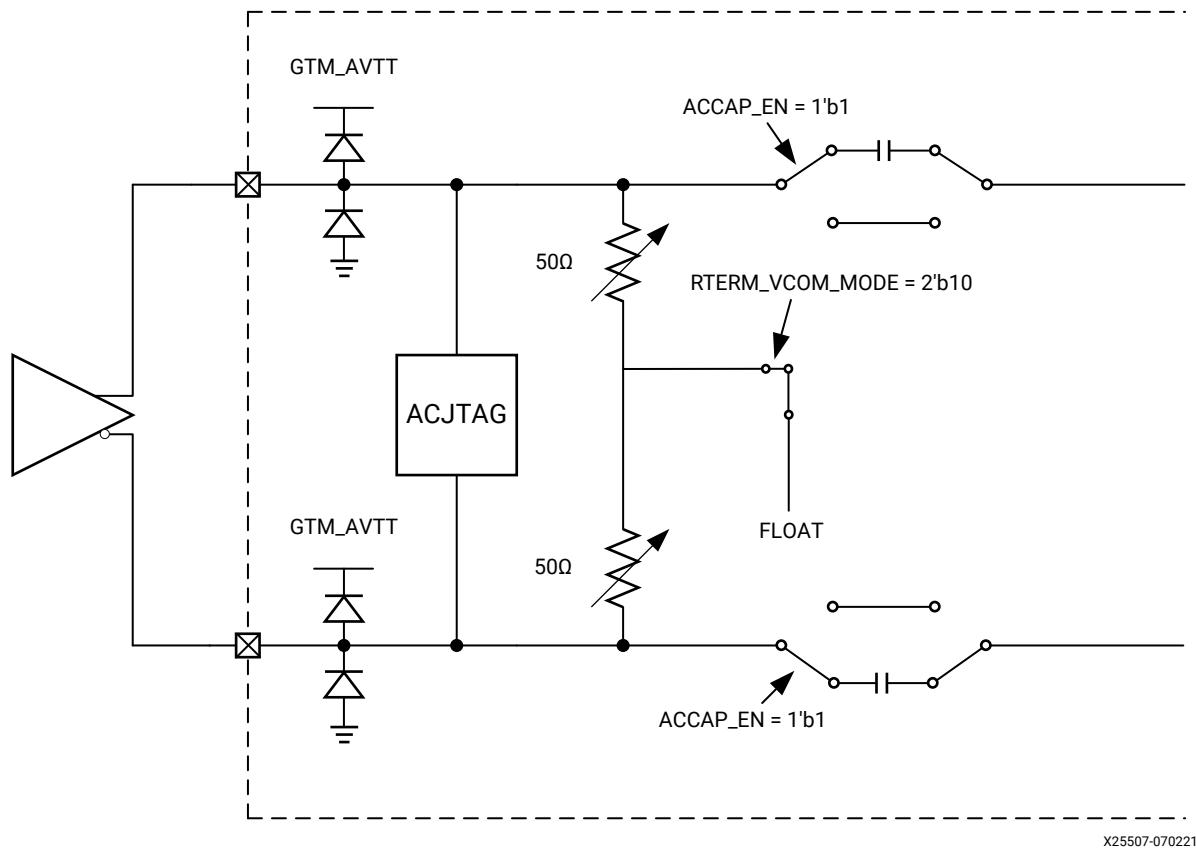
DC Coupling Mode with FLOAT Termination

This mode enables an internal AC capacitor circuit that decouples the analog front end from the equalization block, allowing greater flexibility on the common mode input voltage. The TX or board circuit should ensure that the RX input common mode voltage is set to between 0.2V ~ 1.0V.

Table 61: DC Coupling Mode with FLOAT Termination

Use Mode	External AC Coupling	Termination Voltage	Setting
DC Coupling	No	FLOAT	<ul style="list-style-type: none"> • ACCAP_EN = 1'b1 • LSHIFT_EN = 1'b1 • RTERM_VCOM_MODE = 2'b10

Figure 48: DC Coupling with FLOAT Termination



Ports and Attributes

The following tables show the RX analog front end ports and attributes.

Table 62: RX AFE Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_GTM_RXN	Input	ASYNC	Differential complement of GTMRXP forming a differential receiver input pair. Together these ports represent pads.

Table 62: RX AFE Ports (cont'd)

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_GTMRXP	Input	ASYNC	Differential complement of GTMRXN forming a differential receiver input pair. Together these ports represent pads.

Table 63: RX AFE Attributes

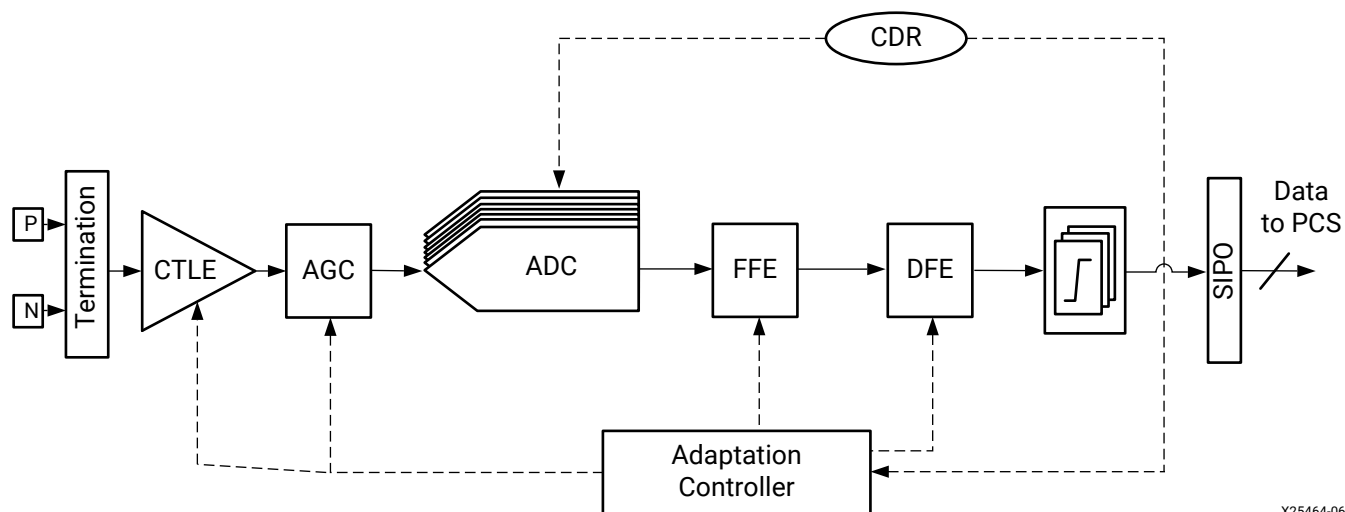
RX AFE Attributes		
Attribute	Address	
CH0_RX_PAD_CFG0	0x0CA3	
CH1_RX_PAD_CFG0	0x0DA3	
CH2_RX_PAD_CFG0	0x0EA3	
CH3_RX_PAD_CFG0	0x0FA3	
Label	Bit Field	Description
LSHIFT_EN	[12:12]	Enable Level-shifting Mode. If enabled, it will function independent of value of accap_en. 1'b0 disabled 1'b1 enabled
ACCAP_EN	[11:11]	Enables the internal AC capacitor which is required in DC-Coupling use modes. 1'b0: Allows for external AC-Coupling capacitor for AC-Coupling use mode 1'b1: Enables internal AC-Coupling capacitor for all DC-Coupling use modes
Attribute	Address	
CH0_RX_PAD_CFG1	0x0CA4	
CH1_RX_PAD_CFG1	0x0DA4	
CH2_RX_PAD_CFG1	0x0EA4	
CH3_RX_PAD_CFG1	0x0FA4	
Label	Bit Field	Description
RTERM_VCOM_MODE	[5:4]	Set the RX termination mode. Float is required when insertion loss is less than 10 dB. 2'b00: terminate to GTM_AVTT 2'b01: terminate to GND 2'b10: terminate to Float 2'b11: terminate to VCOM_REF provided by CTLE, typically 590 mV. (default)

RX Equalizer

A serial link bit error rate (BER) performance is a function of the transmitter, transmission media, and receiver. The transmission media of the channel is bandwidth-limited and the signal traveling through is subjected to attenuation and distortion.

The GTM receiver in Versal adaptive SoCs is an ADC-based buffer that breaks the equalizer into two domains, analog and digital. The incoming signal first passes through the analog stage consisting of a CTLE and AGC stage. The signal is then digitized by the ADC and passes through the feed-forward equalizer (FFE) and decision feedback equalizer (DFE) before being sampled by the CDR.

Figure 49: GTM RX Equalizer



X25464-061721

The combination of the CTLE, FFE, and DFE can compensate for both the pre-cursor and post-cursor of the transmitted symbol. All equalization loops are auto adaptive to handle a wide range of channel profiles and to compensate for any PVT variations.

Ports and Attributes

The following table shows the RX equalizer attributes.

Table 64: RX Equalizer Attributes

RX Equalizer Attributes		
Attribute	Address	
CH0_RX_APT_CFG13	0x0C6D	
CH1_RX_APT_CFG13	0x0D6D	
CH2_RX_APT_CFG13	0x0E6D	
CH3_RX_APT_CFG13	0x0F6D	
Label	Bit Field	Description
INITDAC_HM02	[31:24]	Override value for HM02 loop. This value is in two's complement format.
INITDAC_HM01	[15:7]	Override value for HM01 loop. This value is in two's complement format.

Table 64: RX Equalizer Attributes (cont'd)

RX Equalizer Attributes		
Attribute	Address	
CH0_RX_APT_CFG14	0x0C6E	
CH1_RX_APT_CFG14	0x0D6E	
CH2_RX_APT_CFG14	0x0E6E	
CH3_RX_APT_CFG14	0x0F6E	
Label	Bit Field	Description
INITDAC_HM04	[31:25]	Override value for HM04 loop. This value is in two's complement format.
INITDAC_HM03	[15:9]	Override value for HM03 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG15	0x0C6F	
CH1_RX_APT_CFG15	0x0D6F	
CH2_RX_APT_CFG15	0x0E6F	
CH3_RX_APT_CFG15	0x0F6F	
Label	Bit Field	Description
INITDAC_HM06	[31:26]	Override value for HM06 loop. This value is in two's complement format.
INITDAC_HM05	[15:10]	Override value for HM05 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG16	0x0C70	
CH1_RX_APT_CFG16	0x0D70	
CH2_RX_APT_CFG16	0x0E70	
CH3_RX_APT_CFG16	0x0F70	
Label	Bit Field	Description
INITDAC_HM08	[31:28]	Override value for HM08 loop. This value is in two's complement format.
INITDAC_HM07	[15:11]	Override value for HM07 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG17	0x0C71	
CH1_RX_APT_CFG17	0x0D71	
CH2_RX_APT_CFG17	0x0E71	
CH3_RX_APT_CFG17	0x0F71	
Label	Bit Field	Description
INITDAC_HP01	[8:0]	Override value for HP01 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG18	0x0C72	

Table 64: RX Equalizer Attributes (cont'd)

RX Equalizer Attributes		
CH1_RX_APT_CFG18		0x0D72
CH2_RX_APT_CFG18		0x0E72
CH3_RX_APT_CFG18		0x0F72
Label	Bit Field	Description
INITDAC_HP04	[31:25]	Override value for HP04 loop. This value is in two's complement format.
INITDAC_HP03	[15:8]	Override value for HP03 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG19		0x0C73
CH1_RX_APT_CFG19		0x0D73
CH2_RX_APT_CFG19		0x0E73
CH3_RX_APT_CFG19		0x0F73
Label	Bit Field	Description
INITDAC_HP06	[31:26]	Override value for HP06 loop. This value is in two's complement format.
INITDAC_HP05	[15:9]	Override value for HP05 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG20		0x0C74
CH1_RX_APT_CFG20		0x0D74
CH2_RX_APT_CFG20		0x0E74
CH3_RX_APT_CFG20		0x0F74
Label	Bit Field	Description
INITDAC_HP08	[31:26]	Override value for HP08 loop. This value is in two's complement format.
INITDAC_HP07	[15:10]	Override value for HP07 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG21		0x0C75
CH1_RX_APT_CFG21		0x0D75
CH2_RX_APT_CFG21		0x0E75
CH3_RX_APT_CFG21		0x0F75
Label	Bit Field	Description
INITDAC_HP10	[31:26]	Override value for HP10 loop. This value is in two's complement format.
INITDAC_HP09	[15:10]	Override value for HP09 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG22		0x0C76

Table 64: RX Equalizer Attributes (cont'd)

RX Equalizer Attributes		
CH1_RX_APT_CFG22		0x0D76
CH2_RX_APT_CFG22		0x0E76
CH3_RX_APT_CFG22		0x0F76
Label	Bit Field	Description
INITDAC_HP12	[31:26]	Override value for HP12 loop. This value is in two's complement format.
INITDAC_HP11	[15:10]	Override value for HP11 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG23		0x0C77
CH1_RX_APT_CFG23		0x0D77
CH2_RX_APT_CFG23		0x0E77
CH3_RX_APT_CFG23		0x0F77
Label	Bit Field	Description
INITDAC_HP14	[31:26]	Override value for HP14 loop. This value is in two's complement format.
INITDAC_HP13	[15:10]	Override value for HP13 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG24		0x0C78
CH1_RX_APT_CFG24		0x0D78
CH2_RX_APT_CFG24		0x0E78
CH3_RX_APT_CFG24		0x0F78
Label	Bit Field	Description
INITDAC_HP16	[31:27]	Override value for HP16 loop. This value is in two's complement format.
INITDAC_HP15	[15:10]	Override value for HP15 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG25		0x0C79
CH1_RX_APT_CFG25		0x0D79
CH2_RX_APT_CFG25		0x0E79
CH3_RX_APT_CFG25		0x0F79
Label	Bit Field	Description
INITDAC_HP18	[31:27]	Override value for HP18 loop. This value is in two's complement format.
INITDAC_HP17	[15:11]	Override value for HP17 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG26		0x0C7A

Table 64: RX Equalizer Attributes (cont'd)

RX Equalizer Attributes		
CH1_RX_APT_CFG26		0x0D7A
CH2_RX_APT_CFG26		0x0E7A
CH3_RX_APT_CFG26		0x0F7A
Label	Bit Field	Description
INITDAC_HP20	[31:28]	Override value for HP20 loop. This value is in two's complement format.
INITDAC_HP19	[15:11]	Override value for HP19 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG27		0x0C7B
CH1_RX_APT_CFG27		0x0D7B
CH2_RX_APT_CFG27		0x0E7B
CH3_RX_APT_CFG27		0x0F7B
Label	Bit Field	Description
INITDAC_HP22	[31:28]	Override value for HP22 loop. This value is in two's complement format.
INITDAC_HP21	[15:12]	Override value for HP21 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG28		0x0C7C
CH1_RX_APT_CFG28		0x0D7C
CH2_RX_APT_CFG28		0x0E7C
CH3_RX_APT_CFG28		0x0F7C
Label	Bit Field	Description
INITDAC_HP23	[15:12]	Override value for HP23 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG45		0x0C8D
CH1_RX_APT_CFG45		0x0D8D
CH2_RX_APT_CFG45		0x0E8D
CH3_RX_APT_CFG45		0x0F8D
Label	Bit Field	Description
INITDAC_HP02	[31:24]	Override value for HP02 loop. This value is in two's complement format.
Attribute	Address	
CH0_RX_APT_CFG50		0x0C92
CH1_RX_APT_CFG50		0x0D92
CH2_RX_APT_CFG50		0x0E92
CH3_RX_APT_CFG50		0x0F92

Table 64: RX Equalizer Attributes (cont'd)

RX Equalizer Attributes		
Label	Bit Field	Description
FFE_HM08_EXHOLD	[30:30]	Enable to freeze current FFE tap HM08 adapt value.
FFE_HM07_EXHOLD	[29:29]	Enable to freeze current FFE tap HM07 adapt value.
FFE_HM06_EXHOLD	[28:28]	Enable to freeze current FFE tap HM06 adapt value.
FFE_HM05_EXHOLD	[27:27]	Enable to freeze current FFE tap HM05 adapt value.
FFE_HM04_EXHOLD	[26:26]	Enable to freeze current FFE tap HM04 adapt value.
FFE_HM03_EXHOLD	[25:25]	Enable to freeze current FFE tap HM03 adapt value.
FFE_HM02_EXHOLD	[24:24]	Enable to freeze current FFE tap HM02 adapt value.
FFE_HM01_EXHOLD	[23:23]	Enable to freeze current FFE tap HM01 adapt value.
FFE_HP23_EXHOLD	[22:22]	Enable to freeze current FFE tap HP23 adapt value.
FFE_HP22_EXHOLD	[21:21]	Enable to freeze current FFE tap HP22 adapt value.
FFE_HP21_EXHOLD	[20:20]	Enable to freeze current FFE tap HP21 adapt value.
FFE_HP20_EXHOLD	[19:19]	Enable to freeze current FFE tap HP20 adapt value.
FFE_HP19_EXHOLD	[18:18]	Enable to freeze current FFE tap HP19 adapt value.
FFE_HP18_EXHOLD	[17:17]	Enable to freeze current FFE tap HP18 adapt value.
FFE_HP17_EXHOLD	[16:16]	Enable to freeze current FFE tap HP17 adapt value.
FFE_HP16_EXHOLD	[15:15]	Enable to freeze current FFE tap HP16 adapt value.
FFE_HP15_EXHOLD	[14:14]	Enable to freeze current FFE tap HP15 adapt value.
FFE_HP14_EXHOLD	[13:13]	Enable to freeze current FFE tap HP14 adapt value.
FFE_HP13_EXHOLD	[12:12]	Enable to freeze current FFE tap HP13 adapt value.
FFE_HP12_EXHOLD	[11:11]	Enable to freeze current FFE tap HP12 adapt value.
FFE_HP11_EXHOLD	[10:10]	Enable to freeze current FFE tap HP11 adapt value.
FFE_HP10_EXHOLD	[9:9]	Enable to freeze current FFE tap HP10 adapt value.
FFE_HP09_EXHOLD	[8:8]	Enable to freeze current FFE tap HP09 adapt value.
FFE_HP08_EXHOLD	[7:7]	Enable to freeze current FFE tap HP08 adapt value.
FFE_HP07_EXHOLD	[6:6]	Enable to freeze current FFE tap HP07 adapt value.
FFE_HP06_EXHOLD	[5:5]	Enable to freeze current FFE tap HP06 adapt value.
FFE_HP05_EXHOLD	[4:4]	Enable to freeze current FFE tap HP05 adapt value.
FFE_HP04_EXHOLD	[3:3]	Enable to freeze current FFE tap HP04 adapt value.
FFE_HP03_EXHOLD	[2:2]	Enable to freeze current FFE tap HP03 adapt value.
FFE_HP02_EXHOLD	[1:1]	Enable to freeze current FFE tap HP02 adapt value.
FFE_HP01_EXHOLD	[0:0]	Enable to freeze current FFE tap HP01 adapt value.
Attribute	Address	
CH0_RX_APT_CFG51	0x0C93	
CH1_RX_APT_CFG51	0x0D93	
CH2_RX_APT_CFG51	0x0E93	
CH3_RX_APT_CFG51	0x0F93	

Table 64: RX Equalizer Attributes (cont'd)

RX Equalizer Attributes		
Label	Bit Field	Description
FFE_HM08_OVERRD	[30:30]	Enable to override FFE tap HM08 value according to INITDAC_HM08.
FFE_HM07_OVERRD	[29:29]	Enable to override FFE tap HM07 value according to INITDAC_HM07.
FFE_HM06_OVERRD	[28:28]	Enable to override FFE tap HM06 value according to INITDAC_HM06.
FFE_HM05_OVERRD	[27:27]	Enable to override FFE tap HM05 value according to INITDAC_HM05.
FFE_HM04_OVERRD	[26:26]	Enable to override FFE tap HM04 value according to INITDAC_HM04.
FFE_HM03_OVERRD	[25:25]	Enable to override FFE tap HM03 value according to INITDAC_HM03.
FFE_HM02_OVERRD	[24:24]	Enable to override FFE tap HM02 value according to INITDAC_HM02.
FFE_HM01_OVERRD	[23:23]	Enable to override FFE tap HM01 value according to INITDAC_HM01.
FFE_HP23_OVERRD	[22:22]	Enable to override FFE tap HP23 value according to INITDAC_HP23.
FFE_HP22_OVERRD	[21:21]	Enable to override FFE tap HP22 value according to INITDAC_HP22.
FFE_HP21_OVERRD	[20:20]	Enable to override FFE tap HP21 value according to INITDAC_HP21.
FFE_HP20_OVERRD	[19:19]	Enable to override FFE tap HP20 value according to INITDAC_HP20.
FFE_HP19_OVERRD	[18:18]	Enable to override FFE tap HP19 value according to INITDAC_HP19.
FFE_HP18_OVERRD	[17:17]	Enable to override FFE tap HP18 value according to INITDAC_HP18.
FFE_HP17_OVERRD	[16:16]	Enable to override FFE tap HP17 value according to INITDAC_HP17.
FFE_HP16_OVERRD	[15:15]	Enable to override FFE tap HP16 value according to INITDAC_HP16.
FFE_HP15_OVERRD	[14:14]	Enable to override FFE tap HP15 value according to INITDAC_HP15.
FFE_HP14_OVERRD	[13:13]	Enable to override FFE tap HP14 value according to INITDAC_HP14.
FFE_HP13_OVERRD	[12:12]	Enable to override FFE tap HP13 value according to INITDAC_HP13.
FFE_HP12_OVERRD	[11:11]	Enable to override FFE tap HP12 value according to INITDAC_HP12.
FFE_HP11_OVERRD	[10:10]	Enable to override FFE tap HP11 value according to INITDAC_HP11.
FFE_HP10_OVERRD	[9:9]	Enable to override FFE tap HP10 value according to INITDAC_HP10.
FFE_HP09_OVERRD	[8:8]	Enable to override FFE tap HP09 value according to INITDAC_HP09.

Table 64: RX Equalizer Attributes (cont'd)

RX Equalizer Attributes		
FFE_HP08_OVERRD	[7:7]	Enable to override FFE tap HP08 value according to INITDAC_HP08.
FFE_HP07_OVERRD	[6:6]	Enable to override FFE tap HP07 value according to INITDAC_HP07.
FFE_HP06_OVERRD	[5:5]	Enable to override FFE tap HP06 value according to INITDAC_HP06.
FFE_HP05_OVERRD	[4:4]	Enable to override FFE tap HP05 value according to INITDAC_HP05.
FFE_HP04_OVERRD	[3:3]	Enable to override FFE tap HP04 value according to INITDAC_HP04.
FFE_HP03_OVERRD	[2:2]	Enable to override FFE tap HP03 value according to INITDAC_HP03.
FFE_HP02_OVERRD	[1:1]	Enable to override FFE tap HP02 value according to INITDAC_HP02.
FFE_HP01_OVERRD	[0:0]	Enable to override FFE tap HP01 value according to INITDAC_HP01.
Attribute	Address	
CH0_RX_APT_CFG53	0x0C95	
CH1_RX_APT_CFG53	0x0D95	
CH2_RX_APT_CFG53	0x0E95	
CH3_RX_APT_CFG53	0x0F95	
Label	Bit Field	Description
AGC_EXHOLD	[13:13]	Enable to freeze current automatic gain control (AGC) adapt value.
KH_EXHOLD	[12:12]	Enable to freeze current CTLE high frequency loop (KH) adapt value.
KL_EXHOLD	[11:11]	Enable to freeze current CTLE low frequency loop (KL) adapt value.
OS_EXHOLD	[10:10]	Enable to freeze current offset cancellation (OS) adapt value.
Attribute	Address	
CH0_RX_APT_CFG54	0x0C96	
CH1_RX_APT_CFG54	0x0D96	
CH2_RX_APT_CFG54	0x0E96	
CH3_RX_APT_CFG54	0x0F96	
Label	Bit Field	Description
MAIN_EXHOLD	[29:29]	Adaptation freeze control. This bit must be set to 1'b1 to freeze the loops selected by CH*_RX_APT_CFG50 and CH*_RX_APT_CFG53. If this bit is not enabled, all loops will be auto-adapting.
MAIN_OVRIDE	[28:28]	Adaptation override control. This bit must be set to 1'b1 to override the loops selected in CH*_APT_CFG51 and CH*_APT_CFG53. If this bit is not enabled, all loops will be auto-adapting.

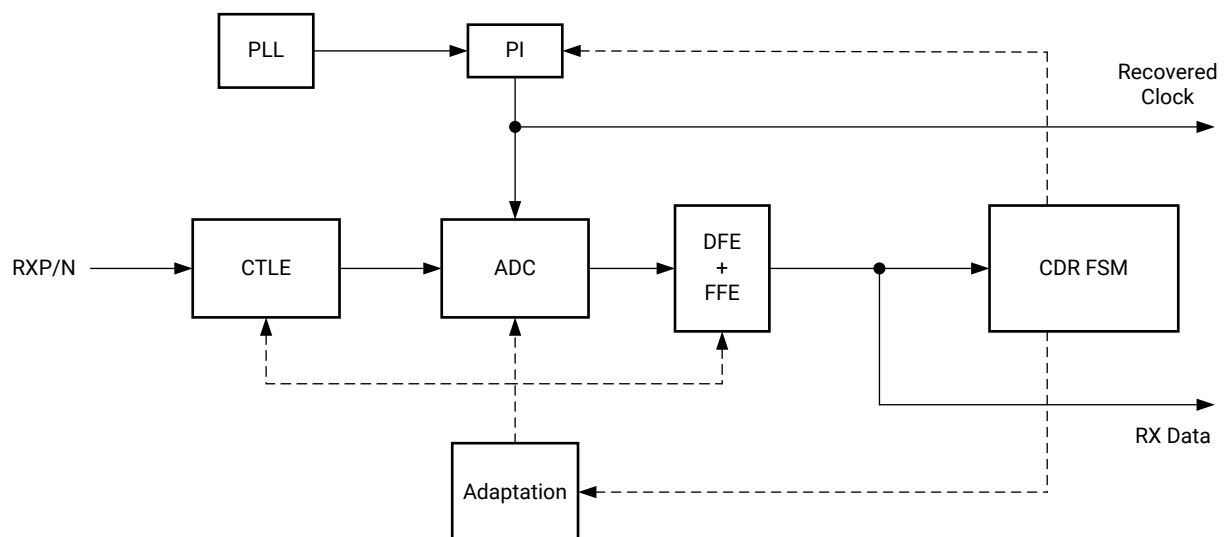
Use Modes

The GTM RX has the ability to receive serial data using two different modulation schemes: NRZ and PAM4. NRZ signals contain one bit of information per symbol while PAM4 signals contain two bits of information per symbol. Using PAM4 modulation doubles the transmitted data bandwidth while maintaining the same unit interval (UI). Use the Versal Adaptive SoC Transceivers Wizard to program the GTM RX to a desired signal modulation. See [Rate Change](#) for modulation changes in runtime.

RX CDR

The RX clock data recovery (CDR) circuit in each Versal adaptive SoC GTM transceiver channel extracts the recovered clock and data from an incoming data stream. The following figure illustrates the architecture of the CDR block. Clock paths are shown with dotted lines for clarity.

Figure 50: CDR Block Diagram



X20925-053118

The GTM transceiver employs the baud-rate phase detection CDR architecture. Incoming data first goes through receiver equalization and ADC where the data is sampled. The sampled data then moves through FFE and DFE before feeding to the CDR state machine and downstream transceiver blocks.

The LCPLL provides a base clock to the phase interpolator. The phase interpolator in turn produces fine, evenly spaced sampling phases to allow the CDR state machine to have fine phase control. The CDR state machine can track incoming data streams that can have a frequency offset from the local PLL reference clock.

Ports and Attributes

The following table shows the RX CDR ports.

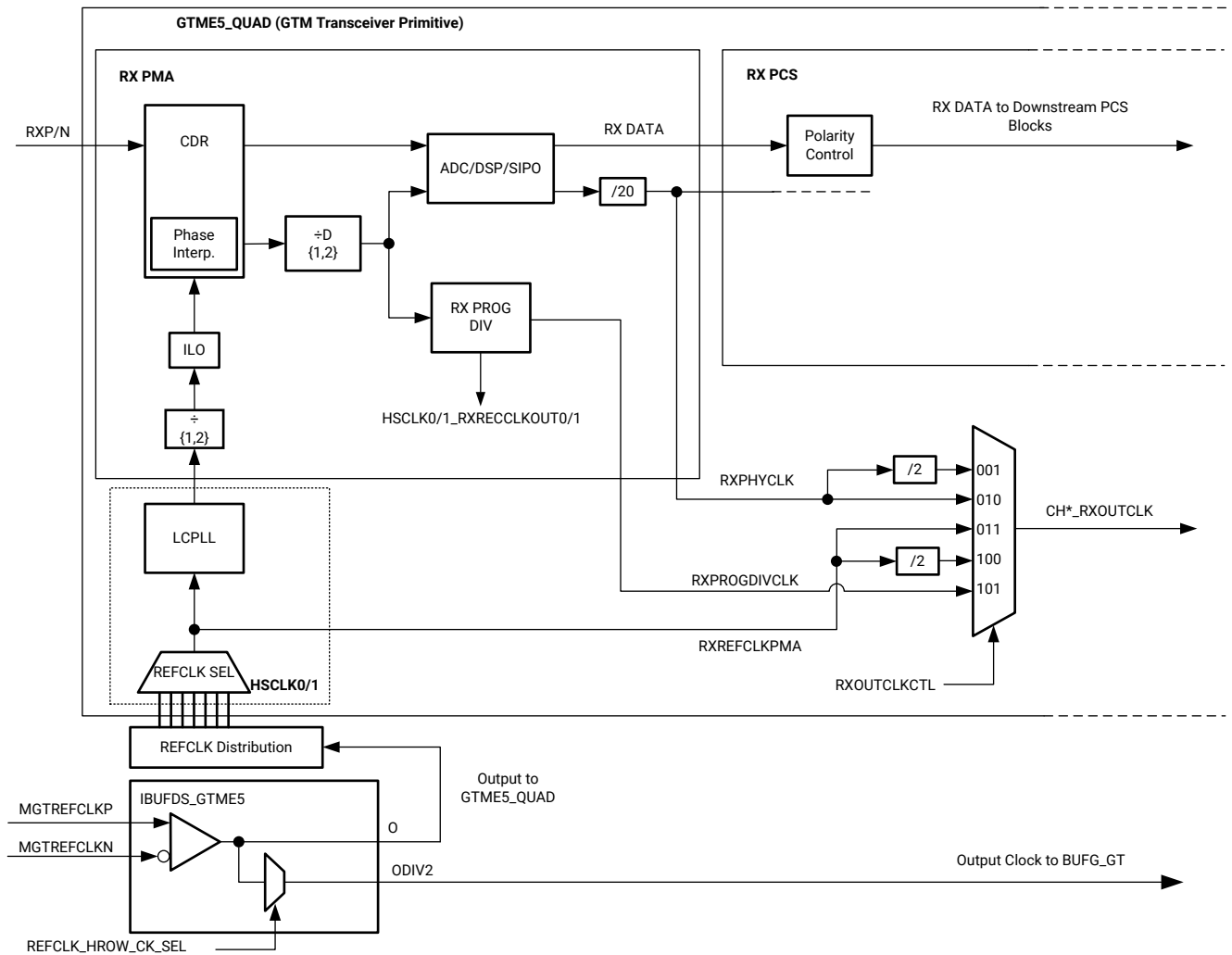
Table 65: CDR Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_RXCDRHOLD	Input	ASYNC	Hold the CDR control loop frozen.
CH[0/1/2/3]_RXCDRLOCK	Output	ASYNC	Reserved.

RX Fabric Clock Output Control

The RX clock divider control block has two main components: serial clock divider control and parallel clock divider and selector control. The clock divider and selector details are illustrated in the figure below.

Figure 51: RX Serial and Parallel Clock Divider



X25593-071024

Notes related to the figure:

1. CH*_RXOUTCLK is used as the source of the interconnect logic clock via BUFG_GT.
2. For details about placement constraints and restrictions on clocking resources (such as BUFG_GT and BUFG_GT_SYNC), refer to the *Versal Adaptive SoC Clocking Resources Architecture Manual* (AM003).
3. The clock output from IBUFDS_GTME5 should only be used after GTPowerGood asserts High.

Serial Clock Divider

Each transmitter PMA module has a D divider that divides down the clock from the PLL for lower line rate support. This serial clock divider D can be set statically for applications with a fixed line rate or changed dynamically for protocols with multiple line rates.

To use the divider D in fixed line rate applications, `RXOUT_DIV` must be set to the appropriate value, and the `CH*_RXRATE` port should be tied to `8'b00000000`.

For multiple line rate applications, the `CH*_TXRATE` port is used to dynamically select the line rate settings, which include the appropriate divider values. See [Rate Change](#) for more details.

Parallel Clock Divider and Selector

The parallel clock outputs from the RX clock divider control block can be used as an interconnect logic clock depending on the line rate and protocol requirements.

The recommended clock for the interconnect logic is the `CH*_RXOUTCLK` from one of the GTM transceivers. It is also possible to bring the `MGTREFCLK` directly to the interconnect logic and use it as the interconnect logic clock.

The `RXOUTCLKCTL` attribute controls the input selector and allows these clocks to be output via the `CH*_RXOUTCLK` port:

- `3'b001`: The divide-by-two version of `RXPHYCLK`.
- `3'b010`: `RXPHYCLK` is the recovered clock that can be brought out to the interconnect logic. The recovered clock is used by protocols that do not have a clock compensation mechanism and require to use a clock synchronous to the data (the recovered clock) to clock the downstream interconnect logic. It is also used by the RX PCS block. This clock is interrupted when the PLL or CDR is reset by one of the related reset signals.
- `3'b011`: `RXREFCLKPMA` is the input reference clock to the LCPLL. For usages that do not require outputting a recovered clock to the interconnect logic, `RXREFCLKPMA` can be used as the system clock. However, `CH*_TXOUTCLK` is usually used as a system clock.
- `3'b100`: The divide-by-two version of `RXREFCLKPMA`.
- `3'b101`: `RXPROGDIVCLK` is the divided down PLL clock after the RX programmable divider. See [RX Fabric Clock Output Control](#) for more details.

RX Programmable Divider

The RX programmable divider shown in [RX Fabric Clock Output Control](#) uses the recovered clock from the CDR to generate a parallel output clock. By using the recovered clock, RX programmable divider, and `BUFG_GT`, `CH*_RXOUTCLK` (`RXOUTCLKCTL = 3'b101`) can be used as a clock source for the interconnect logic instead of consuming PLL or MMCM resources in the interconnect logic. The output clock of the programmable divider can also be brought out to the transceiver reference clock pin configured as an output. The supported divider values are 4, 5, 5.5, 8, 10, 16, 16.5, 20, 32, 33, 40, 64, and 66.

Ports and Attributes

The following table defines the ports required for RX fabric clock output control.

Table 66: RX Fabric Clock Output Control Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_RXOUTCLK	Output	CLOCK	RXOUTCLK is the recommended clock output for the interconnect logic.
CH[0/1/2/3]_RXPROGDIVRESET	Input	ASYNC	This active-High port resets the dividers as well as the CH*_RXPROGDIVRESETDONE indicator. A reset must be performed whenever the input clock source is interrupted.
CH[0/1/2/3]_RXPROGDIVRESETDONE	Output	ASYNC	When the input clock is stable and reset is performed, this active-High signal indicates the reset is completed and the output clock is stable.
CH[0/1/2/3]_RXRATE[7:0]	Input	ASYNC	This port is used to perform rate change on the transceiver RX. The Wizard will preconfigure a list of desired line rates, and this port will be used to dynamically adjust the running line rate based on the preconfigured list. Set this port to the matching preconfigured line rate option value to obtain the proper line rate.

The following table defines the attributes required for RX fabric clock output control.

Table 67: RX Fabric Clock Output Control Attributes

RX Fabric Clock Output Control Attributes		
Attribute	Address	
CH0_CHCLK_CFG3	0x0C2F	
CH1_CHCLK_CFG3	0x0D2F	
CH2_CHCLK_CFG3	0x0E2F	
CH3_CHCLK_CFG3	0x0F2F	
Label	Bit Field	Description
RXOUT_DIV	[28:26]	This attribute selects the setting for the RX serial clock divider D. 3'b000: /1 3'b001: /2

Table 67: RX Fabric Clock Output Control Attributes (cont'd)

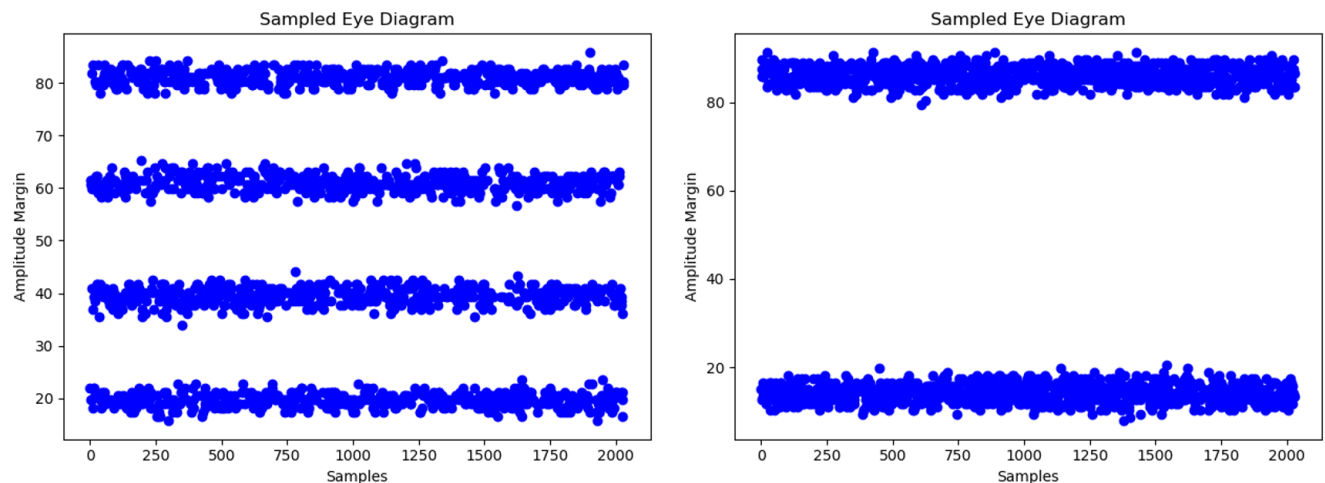
RX Fabric Clock Output Control Attributes		
RXPROGDIVSEL	[25:16]	RX programmable divider ratio. Valid settings are 4, 5, 5.5, 8, 10, 16, 16.5, 20, 32, 33, 40, 64, and 66. 10'b1001011000: /4 10'b1001111000: /5 10'b1110011000: /5.5 10'b0001011000: /8 10'b1001000000: /8 (Fullrate) 10'b0001111000: /10 10'b1001100000: /10 (Fullrate) 10'b0001000000: /16 10'b1001000010: /16 (Fullrate) 10'b1100011000: /16.5 10'b0001100000: /20 10'b1001100010: /20 (Fullrate) 10'b0001000010: /32 10'b1001000110: /32 (Fullrate) 10'b0100011000: /33 10'b1100000000: /33 (Fullrate) 10'b0001100010: /40 10'b1001100110: /40 (Fullrate) 10'b1001001110: /64 10'b0100000000: /66 10'b1100000010: /66 (Fullrate) 10'b0001100110: /80 10'b1001101110: /80 (Fullrate)
Attribute	Address	
CH0_PMA_MISC_CFG0	0x0C27	
CH1_PMA_MISC_CFG0	0x0D27	
CH2_PMA_MISC_CFG0	0x0E27	
CH3_PMA_MISC_CFG0	0x0F27	
Label	Bit Field	Description
RXOUTCLKCTL	[31:29]	This attribute selects the source for CH*_RXOUTCLK. 3'b001: Div2 version of RXPHYCLK 3'b010: RXPHYCLK 3'b011: RXREFCLKPMA 3'b100: Div2 version of RXREFCLKPMA 3'b101: RXPROGDIVCLK

RX Margin Analysis

As line rates and channel attenuation increase, the receiver equalizers are more often enabled to overcome channel attenuation. This poses a challenge to system bring-up because the quality of the link cannot be determined by measuring the far-end eye opening at the receiver pins. At high line rates, the received eye measurement on the printed circuit board can appear to be completely closed even though the internal eye after the receiver equalizer is open.

Because the GTM receiver is ADC-based, the conventional eye scan used in the previous family of transceivers (such as GTH and GTY transceivers) is not possible. The GTM RX provides a non-destructive sampled eye diagram that can be used to measure and visualize the receiver signal margin after the equalizer, as shown in the following figure.

Figure 52: Sampled Eye Diagram for (a) PAM4 and (b) NRZ Modulation Respectively



The sampled eye diagram is constructed by plotting one sample per symbol located at the CDR sampling point (after the equalizer). By plotting multiple samples, the image looks like the preceding figure. Sampled eye supports both NRZ and PAM4 modulation, with the only difference being that the NRZ eye consists of two amplitude margin levels, and PAM4 with four distinct amplitude margin levels. For more information on implementing the sampled eye diagram in your application, contact your local FAE.

RX Polarity Control

If the RXP and RXN differential traces are accidentally swapped on the PCB, the differential data received by the GTM RX can be reversed. The GTM RX allows inversion to be done on parallel bytes in the PCS after the SIPO to offset reversed polarity on the differential pair. The polarity control function uses the CH*_RXPOLARITY input, which is driven High from the interconnect logic interface to invert polarity.

Ports and Attributes

The following table shows the RX polarity control ports.

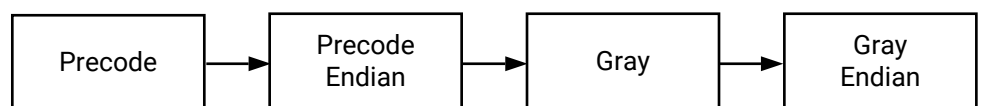
Table 68: RX Polarity Control Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_RXPOLARITY	Input	ASYNC	This port can invert the polarity of incoming data: 0: Not inverted. RXP is positive and RXN is negative. 1: Inverted. RXP is negative and RXN is positive.

RX Pre-Coder

The GTM receiver supports pre-coding. Pre-coding can be used to reduce receiver DFE error propagation by reducing 1-tap burst error runs into two errors for every error event. The RX pre-coder has independent endianness control blocks. The following figure shows the sequence in which bits are flipped.

Figure 53: RX Pre-Coder Bit Flip Sequence



X29436-071124

Ports and Attributes

The following table shows the RX pre-coder attributes.

Table 69: RX Pre-Coder Attributes

RX Pre-Coder Attributes		
Attribute	Address	
CH0_RX_PCS_CFG0	0x0C14	
CH1_RX_PCS_CFG0	0x0D14	
CH2_RX_PCS_CFG0	0x0E14	
CH3_RX_PCS_CFG0	0x0F14	
Label	Bit Field	Description
RX_PRECODE_ENDIAN	[11:11]	Controls RX Pre-Coder Endianness: 0: Big endian 1: Little endian
RX_PRECODE_BYP_EN	[10:10]	Controls RX Pre-Code: 0: Enable RX Pre-Code 1: Bypass RX Pre-Code

RX Gray Decoder

The GTM receiver supports two types of binary encoding options, linear and gray coding. By using gray coding, only one bit error per symbol is made for incorrect decisions, thus reducing the bit-error rate by more than 33%.

Ports and Attributes

The following table shows the RX gray decoder attributes.

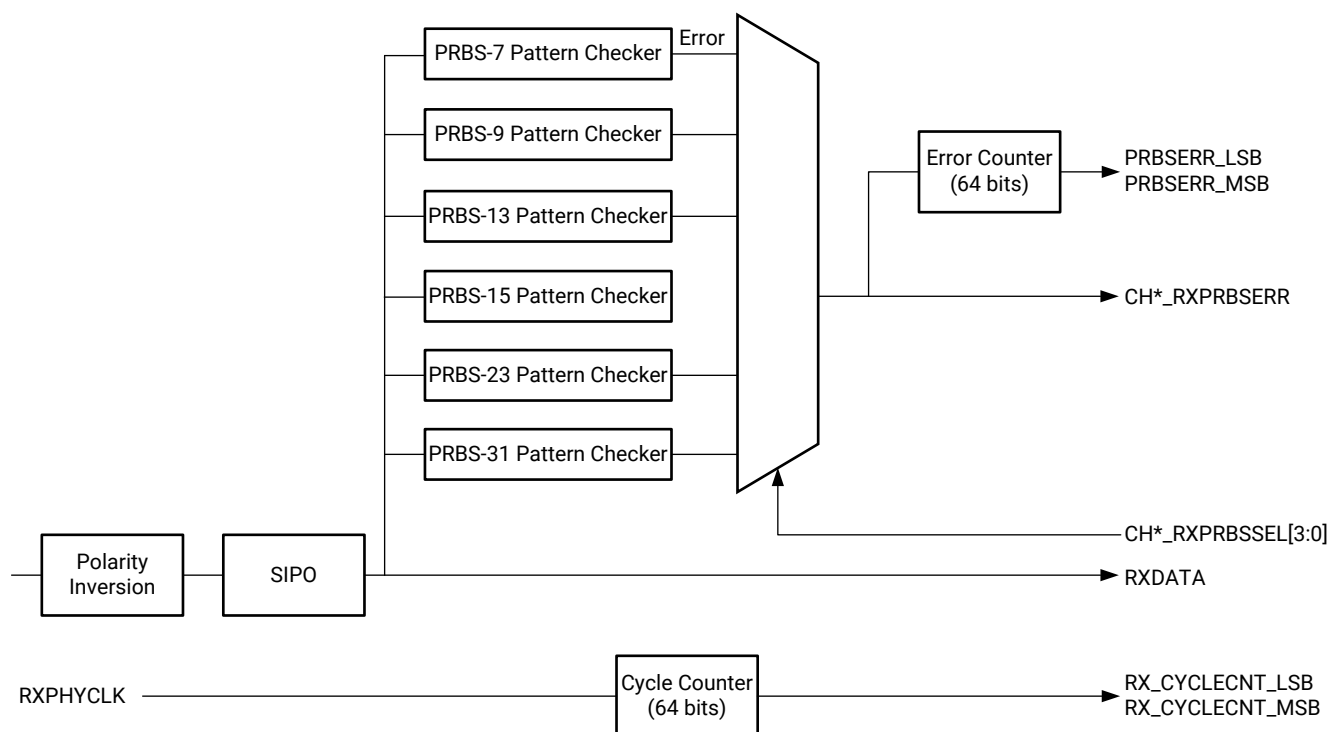
Table 70: RX Gray Decoder Attributes

RX Gray Decoder Attributes		
Attribute	Address	
CH0_RX_PCS_CFG0	0x0C14	
CH1_RX_PCS_CFG0	0x0D14	
CH2_RX_PCS_CFG0	0x0E14	
CH3_RX_PCS_CFG0	0x0F14	
Label	Bit Field	Description
RX_GRAY_ENDIAN	[13:13]	Controls RX Gray Code Endianness: 0: Big endian 1: Little endian
RX_GRAY_BYP_EN	[12:12]	Controls RX Gray Decoder: 0: Enable RX Gray Decoder 1: Bypass RX Gray Decoder

RX Pattern Checker

The GTM receiver includes a built-in PRBS checker (see the following figure). This checker can be set to check for one of the six industry-standard PRBS patterns. The checker is self-synchronizing and works on the incoming data before comma alignment or decoding. This function can be used to test the signal integrity of the channel.

Figure 54: RX Pattern Checker Block



X25484-062221

Ports and Attributes

The following tables show the RX pattern checker ports and attributes.

Table 71: RX Pattern Checker Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_RXPRBSCNTRESET	Input	ASYNC	Assert this port High to reset the PRBS error counter.
CH[0/1/2/3]_RXPRBSCNTSTOP	Input	ASYNC	Input control to stop the PRBS error counter and RXPHYCLK counter. The PRBS checker needs to restart the error counter after a PRBS stop.

Table 71: RX Pattern Checker Ports (cont'd)

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_RXPRBSERR	Output	RXUSRCLK	This non-sticky status output indicates that PRBS errors have occurred. Only use PRBSERR_LSB and PRBSERR_MSB to read the precise bit error counts.
CH[0/1/2/3]_RXPRBSLOCKED	Output	RXUSRCLK	Output to indicate that the RX PRBS checker has been error free for RXPRBS_LINKACQ_CNT PHYCLK cycles after reset. Once asserted High, RXPRBSLOCKED does not de-assert until reset of the RX pattern checker via a reset of the RX (GTRXRESET, RXPMARESET, or RXPCSRESET in sequential mode) or a reset of the PRBS error counter (RXPRBSCNTRESET).
CH[0/1/2/3]_RXPRBSSEL[3:0]	Input	ASYNC	Receiver PRBS checker test pattern control. Only these settings are valid: 4'b0000: Standard operation mode. (PRBS check is off) 4'b0001: PRBS-7 4'b0010: PRBS-9 4'b0011: PRBS-15 4'b0100: PRBS-23 4'b0101: PRBS-31 4'b0110: PRBS-13 After changing patterns, perform a reset of the RX (GTRXRESET, RXPMARESET, or RXPCSRESET) or a reset of the PRBS error counter (RXPRBSCNTRESET) such that the RX pattern checker can attempt to reestablish the link acquired. No checking is done for non-PRBS patterns.
CH[0/1/2/3]_RXQPRBSEN	Input	ASYNC	RX QPRBS Enable - Used along with the RXPRBSSEL. Not supported for PRBS7

Table 72: RX Pattern Checker Attributes

RX Pattern Checker Attributes		
Attribute	Address	
CH0_COE_STATUS_PRBSERR_CNT0	0x0840	
CH1_COE_STATUS_PRBSERR_CNT0	0x0940	
CH2_COE_STATUS_PRBSERR_CNT0	0x0A40	
CH3_COE_STATUS_PRBSERR_CNT0	0x0B40	
Label	Bit Field	Description
DO_FROM_PRBSERR_LSB	[31:0]	Sum of the bit error count of the received data pattern for a time the particular PRBS mode is enabled. LSB [31:0] bits of the PRBS error counter.
Attribute	Address	
CH0_COE_STATUS_PRBSERR_CNT1	0x0841	
CH1_COE_STATUS_PRBSERR_CNT1	0x0941	

Table 72: RX Pattern Checker Attributes (cont'd)

RX Pattern Checker Attributes		
CH2_COE_STATUS_PRBSERR_CNT1	0x0A41	
CH3_COE_STATUS_PRBSERR_CNT1	0x0B41	
Label	Bit Field	Description
DO_FROM_PRBSERR_MSB	[31:0]	Sum of the bit error count of the received data pattern for a time the particular PRBS mode is enabled. MSB [31:0] bits of the PRBS error counter. To read this value, DO_FROM_PRBSERR_LSB must be read first. This causes current value of this word to be captured at the same time for subsequent reading.
Attribute	Address	
CH0_COE_STATUS_PRBS_LINK2RESEED_CNT	0x0844	
CH1_COE_STATUS_PRBS_LINK2RESEED_CNT	0x0944	
CH2_COE_STATUS_PRBS_LINK2RESEED_CNT	0x0A44	
CH3_COE_STATUS_PRBS_LINK2RESEED_CNT	0x0B44	
Label	Bit Field	Description
RXPRBSLOCKED	[17:17]	RX PRBS Link Locked Status going to Fabric
RXPRBSERR	[16:16]	RX PRBS Link Error Status going to Fabric
Attribute	Address	
CH0_COE_STATUS_RX_CYCLECNT0	0x0842	
CH1_COE_STATUS_RX_CYCLECNT0	0x0942	
CH2_COE_STATUS_RX_CYCLECNT0	0x0A42	
CH3_COE_STATUS_RX_CYCLECNT0	0x0B42	
Label	Bit Field	Description
DO_FROM_RX_CYCLECNT_LSB	[31:0]	Lower 32 bit RXPHYCLK cycle counter value. To read this value, DO_FROM_PRBSERR_LSB must be read first. This causes current value of this word to be captured at the same time for subsequent reading. Used in conjunction with COE_STATUS_PRBSERR_CNT0/CNT1.
Attribute	Address	
CH0_COE_STATUS_RX_CYCLECNT1	0x0843	
CH1_COE_STATUS_RX_CYCLECNT1	0x0943	
CH2_COE_STATUS_RX_CYCLECNT1	0x0A43	
CH3_COE_STATUS_RX_CYCLECNT1	0x0B43	
Label	Bit Field	Description
DO_FROM_RX_CYCLECNT_MSB	[31:0]	Upper 32-bit RXPHYCLK cycle counter value. To read this value, DO_FROM_PRBSERR_LSB must be read first. This causes current value of this word to be captured at the same time for subsequent reading. Used in conjunction with COE_STATUS_PRBSERR_CNT0/CNT1.

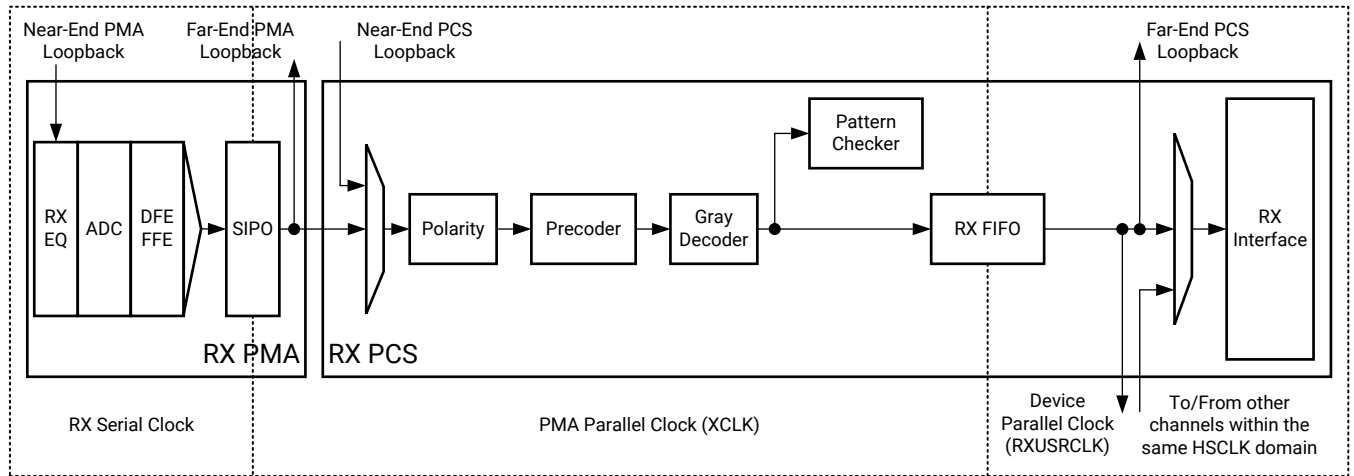
Table 72: RX Pattern Checker Attributes (cont'd)

RX Pattern Checker Attributes		
Attribute	Address	
CH0_RX_PCS_CFG0	0x0C14	
CH1_RX_PCS_CFG0	0x0D14	
CH2_RX_PCS_CFG0	0x0E14	
CH3_RX_PCS_CFG0	0x0F14	
Label	Bit Field	Description
RXPRBS_LINKACQ_CNT	[28:21]	RX pattern checker link acquire count. Used in conjunction with output port RXPRBSLOCKED. After the RX PRBS checker has seen RXPRBS_LINKACQ_CNT XCLK cycles of error-free PRBS data, RXPRBSLOCKED is asserted High. Valid range is 15-255.
Attribute	Address	
CH0_TX_PCS_CFG0	0x0C16	
CH1_TX_PCS_CFG0	0x0D16	
CH2_TX_PCS_CFG0	0x0E16	
CH3_TX_PCS_CFG0	0x0F16	
Label	Bit Field	Description
RXPRBSERR_LOOPBACK	[23:23]	Enable RXPRBSERR error injection: 0: Enable TXPRBSFORCEERR forces onto the TX PRBS 1: Enable RXPRBSERR bit to be internally looped back to TXPRBSFORCEERR

RX FIFO

The transceiver RX datapath has two internal parallel clock domains used in the PCS: the PMA parallel clock domain (XCLK) and the fabric clock domain (RXUSRCLK). To receive data, the RX buffer provides data width conversion between these clock domains when necessary, depending on the operating data width and encoding mode. The following figure shows the RX datapath clock domains.

Figure 55: RX Datapath Clock Domains



X25494-062321

The GTM receiver includes an RX FIFO to support data width conversion when data crosses from the XCLK to RXUSRCLK domain. The buffer does not tolerate ppm differences and only provides phase compensation between the two clocks. The RX FIFO inside the GTM transceiver must always be used and cannot be bypassed.

Using the RX FIFO

Reset the RX FIFO whenever CH*_RXBUFSTATUS indicates an overflow or underflow condition. The RX FIFO can be reset by using the RX reset procedure or PCS component reset described in [RX Initialization and Reset](#).

Reading RX FIFO Latency

The datapath latency through the RX FIFO is calculated statistically using RXLATCLK, which is asynchronous to XCLK. RX_SAMPLE_COUNT in CH*_RX_PCS_CFG1 determines the number of RXLATCLK cycles over which averaging takes place. RXLATCLK needs to be asynchronous in both phase and frequency to both the write clock and the read clock. The RXLATCLK has the same range as the RXUSRCLK, greater than zero frequency and less than the max USRCLK frequency. The measured latency value in RX_FIFO_LATENCY is updated once per sampling period, and located in COE_STATUS_RX_GB_DBG6 for the 160x512 FIFO and COE_STATUS_RX_GB_DBG8 for the 160x320 FIFO.

These settings are used to read the latency:

- Set CH*_RX_PCS_CFG1[24:22] (RX_SAMPLE_COUNT)
 - A higher averaging period gives a more accurate latency value.
- For the 160x512 FIFO, read CH*_COE_STATUS_RX_GB_DBG6[16:0] (RX_FIFO_LATENCY): The value is in units of 1/8 UI.

- For the 160x320 FIFO, read CH*_COE_STATUS_RX_GB_DBG8[16:0] (RX_FIFO_LATENCY):
The value is in units of 1/8 UI.
- The actual latency is RX_FIFO_LATENCY plus a fixed value.

Note: Use the 160x512 FIFO when data widths are 512/256/128/64/32 and the 160x320 FIFO when data widths are 320/160/80/40.

Ports and Attributes

The following table shows the RX FIFO ports.

Table 73: RX FIFO Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_RXBUFSTATUS[2:0]	Output	RXUSRCLK	RX buffer status: Bit [2]: Reserved. Tied High. Bit [1]: FIFO overflow status. A value of 1 indicates FIFO overflow. Bit [0]: FIFO underflow status. A value of 1 indicates FIFO underflow.
CH[0/1/2/3]_RXLATCLK	Input	CLOCK	Input port used to provide a clock for the RX FIFO latency calculation.

The following table shows the RX FIFO attributes.

Table 74: RX FIFO Attributes

RX FIFO Attributes		
Attribute	Address	
CH0_COE_STATUS_RX_GB_DBG6	0x0836	
CH1_COE_STATUS_RX_GB_DBG6	0x0936	
CH2_COE_STATUS_RX_GB_DBG6	0x0A36	
CH3_COE_STATUS_RX_GB_DBG6	0x0B36	
Label	Bit Field	Description
RX_FIFO_LATENCY	[16:0]	Measured latency of the 160x512 RX FIFO averaged over RX_SAMPLE_PERIOD (CH*_RX_PCS_CFG1[24:22]) cycles. The reported latency is in units of 1/8 UI. Use the 160x512 FIFO when data widths are 512/256/128/64/32. The COE_STATUS_RX_GB_DBG6 is read-only.
Attribute	Address	
CH0_COE_STATUS_RX_GB_DBG8	0x0838	
CH1_COE_STATUS_RX_GB_DBG8	0x0938	
CH2_COE_STATUS_RX_GB_DBG8	0x0A38	
CH3_COE_STATUS_RX_GB_DBG8	0x0B38	

Table 74: RX FIFO Attributes (cont'd)

RX FIFO Attributes		
Label	Bit Field	Description
RX_FIFO_LATENCY	[16:0]	Measured latency of the 160x320 RX FIFO averaged over RX_SAMPLE_PERIOD (CH*_RX_PCS_CFG1[24:22]) cycles. The reported latency is in units of 1/8 UI. Use the 160x320 FIFO when data widths are 320/160/80/40. The COE_STATUS_RX_GB_DBG8 is read-only.
Attribute	Address	
CH0_RX_PCS_CFG1	0x0C15	
CH1_RX_PCS_CFG1	0x0D15	
CH2_RX_PCS_CFG1	0x0E15	
CH3_RX_PCS_CFG1	0x0F15	
Label	Bit Field	Description
RX_SAMPLE_COUNT	[24:22]	Number of samples used for latency averaging for RX FIFO: 3'b000: 256 3'b001: 512 3'b010: 1024 3'b011: 2048 3'b100: 4096 3'b101: 8192 3'b110: 16384 3'b111: 32768

RX Interface

The RX interface is the gateway to the RX datapath of the GTM transceiver. The application receives data through the GTM transceiver by reading data to the RXDATA port on the positive edge of RXUSRCLK. Port widths can be 32, 40, 64, 80, 128, 160, and 256 bits for NRZ mode, or 64, 80, 128, 160, 256, 320, and 512 bits for PAM4 mode. The rate of the parallel clock (RXUSRCLK) at the interface is determined by the RX line rate and the width of the RXDATA port.

Each RX fabric interface natively supports up to 256 bits output to the fabric design. To extend fabric interface support to 320 and 512 bits, two RX channel interfaces should be combined—this is referred to as half density mode. In this mode, only one RX front end channel is active. Refer to [RX Interface Width Configuration](#) for more details.

RX Interface Width Configuration

The GTM receiver contains an 80-bit internal datapath in NRZ mode and a 160-bit internal datapath for PAM4 mode that is configurable by setting the RX_PMA_DATA_WIDTH attribute. The interface width is configurable by setting the RX_DATA_WIDTH attribute. In NRZ mode, the interface can be configured to 32, 40, 64, 80, 128, 160, and 256 bits. In PAM4 mode, the interface can be configured to 64, 80, 128, 160, 256, 320, and 512 bits.

The following table shows how the interface width for the RX datapath is selected.

Table 75: RX Interface Datapath Configuration

Encoding	Density	RX_PMA_DATA_WIDTH Encoding	RX_DATA_WIDTH Encoding	RX Internal Width	RX Interface Width
NRZ	Full	0	4	80	32
			5		40
			6		64
			7		80
			8		128
			9		160
			10		256
PAM4	Full	1	6	160	64
			7		80
			8		128
			9		160
			10		256
	Half		11		320
			12		512

When the interface width is configured to either 320 or 512 bits, this results in a half density configuration in which two channel interfaces are combined to act as one. In this mode, only a single PCS/PMA front end channel is in operation.

When either channel CH0 or CH1 is operating in half density mode, the fabric interface combines the data from adjacent channels to group as a set of 320 or 512 bits before forwarding to the RX FIFO. Designating a channel to be used in half density mode can be done by setting the adjacent channel's RX_USRCLK_SEL attribute High. An additional limitation is that the half density configuration must use either the channel 0/1 pair or channel 2/3 pair in the same Quad. Mixing and matching with different channels is not allowed. The following table shows possible channel combinations.

Table 76: RX Half Density Modes

CH0 RX_USRCLK_SEL	CH1 RX_USRCLK_SEL	Fabric Interface Operating Mode
0	0	Both channels are in full density mode.
0	1	CH0 is in half density mode.
1	0	CH1 is in half density mode.
1	1	Invalid.

The following table shows the port connections for different data widths.

Table 77: RX Fabric Port Connections for Different Data Widths

RX Fabric Data Width	Internal PCS Data Path Bit Ranges	Fabric Width Mode	Non-Active Channel (MSB) Last Bits to RX	Active Channel (LSB) First Bits to RX
512	[511:0]	Half Density	RXDATA[255:0]	RXDATA[255:0]
320	[415:256, 159:0]	Half Density	RXDATA[159:0]	RXDATA[159:0]
256	[255:0]	Full Density	-	RXDATA[255:0]
160	[159:0]	Full Density	-	RXDATA[159:0]
128	[127:0]	Full Density	-	RXDATA[127:0]
80	[79:0]	Full Density	-	RXDATA[79:0]
64	[63:0]	Full Density	-	RXDATA[63:0]
40	[39:0]	Full Density	-	RXDATA[39:0]
32	[31:0]	Full Density	-	RXDATA[31:0]

Notes:

1. In half-density mode, the least significant bits are in the active channel, and the most significant bits are in the non-active channel.
2. In 320-bit mode, the data widths are split between the two channels to have 160 in each.

Ports and Attributes

The following table shows the RX interface ports.

Table 78: RX Interface Ports

Port	Direction	Clock Domain	Description
CH[0/1/2/3]_RXDATA[255:0]	Output	RXUSRCLK	The bus for receiving data.
CH[0/1/2/3]_RXUSRCLK	Input	CLOCK	This port is used to provide a clock for the internal RX PCS datapath.

The following table shows the RX interface attributes.

Table 79: RX Interface Attributes

RX Interface Attributes		
Attribute	Address	
CH0_RX_PCS_CFG0	0x0C14	
CH1_RX_PCS_CFG0	0x0D14	
CH2_RX_PCS_CFG0	0x0E14	
CH3_RX_PCS_CFG0	0x0F14	
Label	Bit Field	Description
RX_USRCLK_SEL	[15:14]	Clock selection for data usage at fabric interface. 2'b00: USRCLK for single channel data fabric interface usage (full density) 2'b01: Adjacent channel user clock for dual channel data fabric interface usage (half density) 2'b10: Reserved. 2'b11 Reserved.
RX_PMA_DATA_WIDTH	[4:4]	Controls the width of the RX internal PCS datapath. 0: 80-bit internal datapath width (NRZ) 1: 160-bit internal datapath width (PAM4)
RX_DATA_WIDTH	[3:0]	Controls the RX interface width. 4'b0100: 32-bit 4'b0101: 40-bit 4'b0110: 64-bit 4'b0111: 80-bit 4'b1000: 128-bit 4'b1001: 160-bit 4'b1010: 256-bit 4'b1011: 320-bit 4'b1100: 512-bit

RXUSRCLK Generation

The RX Interface includes the parallel clock RXUSRCLK. RXUSRCLK is the internal clock for the PCS logic in the transmitter. The required rate for RXUSRCLK depends on the interface width of the GTME5_QUAD primitive and the RX line rate of the GTM transmitter. The following equation shows how to calculate the required rate for RXUSRCLK for all cases.

Equation 5: **RXUSRCLK**

$$RXUSRCLK = \frac{Line\ Rate}{Interface\ Width}$$

RXUSRCLK is the main synchronization clock for all signals into the RX side of the GTM transceiver. Most signals into the RX side of the GTM transceiver are sampled on the positive edge of RXUSRCLK.

Board Design Guidelines

Topics related to implementing a design on a printed circuit board that uses GTM transceivers are presented in this chapter. GTM transceivers are analog circuits that require special consideration and attention when designing and implementing them on a printed circuit board. Besides an understanding of the functionality of the device pins, a design that performs optimally requires attention to issues such as device interfacing, transmission line impedance and routing, power supply design filtering and distribution, component selection, and PCB layout and stackup design.

Pin Description and Design Guidelines

Transceiver Pin Descriptions

The following table defines the GTM transceiver Quad pins.

Table 80: Transceiver Quad Descriptions

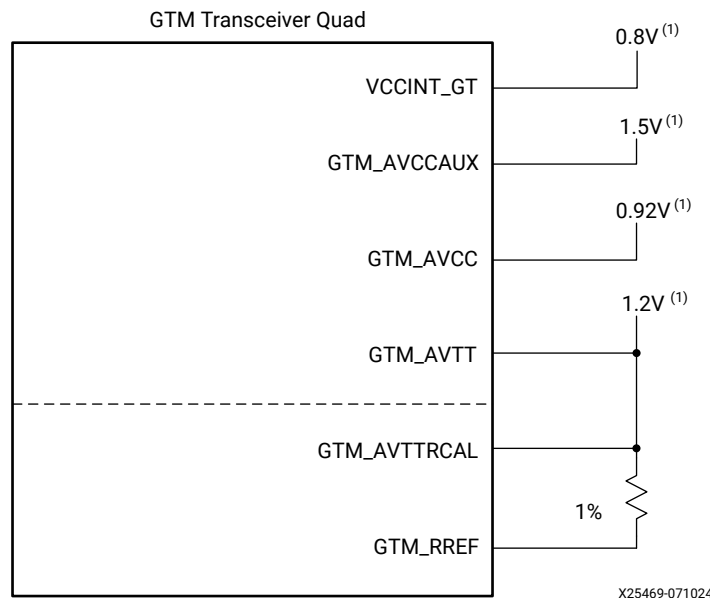
Pins	Dir	Description
GTM_REFCLK0P GTM_REFCLK0N	In/Out (Pad)	Configured as either reference clock input pins or RX recovered clock output pins for the Quad.
GTM_REFCLK1P GTM_REFCLK1N	In/Out (Pad)	Configured as either reference clock input pins or RX recovered clock output pins for the Quad.
GTM_RXP[0/1/2/3]/ GTM_RXN[0/1/2/3]	In (Pad)	RXP and RXN are the differential input pairs for each of the receivers in the transceiver Quad.
GTM_TXP[0/1/2/3]/ GTM_TXN[0/1/2/3]	Out (Pad)	TXP and TXN are the differential output pairs for each of the transmitters in the transceiver Quad.
GTM_AVTTRCAL	In (Pad)	Bias current supply for the termination resistor calibration circuit. See Termination Resistor Calibration Circuit .
GTM_RREF	In (Pad)	Calibration resistor input pin for the termination resistor calibration circuit. See Termination Resistor Calibration Circuit .
VCCINT_GT	In (Pad)	VCCINT_GT is the digital supply voltage for select modules in the GTM transceivers. Most packages have multiple groups of power supply connections in the package for VCCINT_GT. Refer to the package pin definitions to identify in which power supply group a specific GTM transceiver Quad is located. See the Versal device data sheets for the nominal voltage.

Table 80: Transceiver Quad Descriptions (cont'd)

Pins	Dir	Description
GTM_AVCC	In (Pad)	GTM_AVCC is the analog supply for the internal analog circuits of the GTM transceiver Quad tile. This includes the analog circuits for the PLLs, transmitters, and receivers. Most packages have multiple groups of power supply connections in the package for GTM_AVCC. Refer to the package pin definitions to identify in which power supply group a specific GTM transceiver Quad is located. See the Versal device data sheets for the nominal voltage.
GTM_AVTT	In (Pad)	GTM_AVTT is the analog supply for the transmitter and receiver termination circuits of the GTM transceiver Quad tile. Most packages have multiple groups of power supply connections in the package for GTM_AVTT. Refer to the package pin definitions to identify in which power supply group a specific GTM transceiver Quad is located. See the Versal device data sheets for the nominal voltage.
GTM_AVCCAUX	In (Pad)	GTM_AVCCAUX is the auxiliary analog LCPLL voltage supply for the transceivers. Most packages have multiple groups of power supply connections in the package for GTM_AVCCAUX. Refer to the package pin definitions to identify in which power supply group a specific GTM transceiver Quad is located. See the Versal device data sheets for the nominal voltage.

The following figure shows the external power supply connections with the GTM transceivers.

Figure 56: GTM Transceiver Power Supply Connections



Note: The voltage values are nominal. See the [Versal device data sheets](#) for values and tolerances.

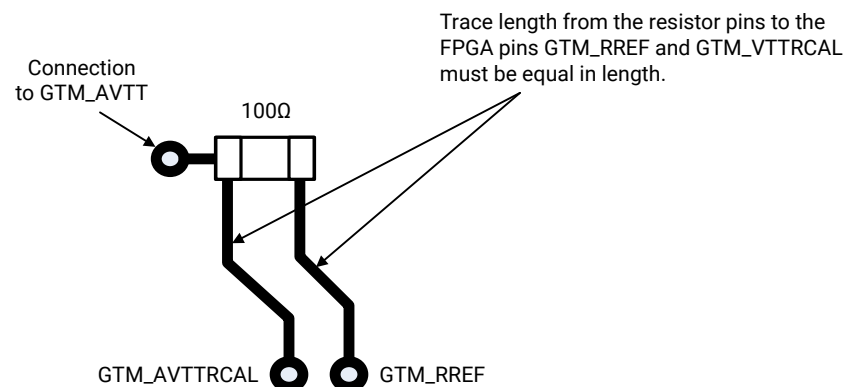
Termination Resistor Calibration Circuit

There is one resistor calibration circuit (RCAL) shared between all transceiver Quad primitives in a transceiver Quad column. The GTM_AVTTRCAL and GTM_RREF pins connect the bias circuit power and the external calibration resistor to the RCAL circuit. The RCAL circuit performs the resistor calibration only during configuration of the Versal adaptive SoC. Prior to configuration, all analog supply voltages must be present and within the proper tolerance as specified in the [Versal device data sheets](#). If an entire power supply group (PSG) is not used by any Quads, GTM_AVTTRCAL and GTM_RREF should be tied to ground. See [Analog Power Supply Pins](#) for more details regarding RCAL biasing recommendations when there are unused Quads.

The RCAL circuit is associated with the GTM transceiver Quad that is the RCAL master. The RCAL master performs the termination resistor calibration during configuration of the Versal adaptive SoC and then distributes the calibrated values to all of the GTM transceiver Quads in the column. The Quad in which the RCAL circuit is located must be powered on.

Connect the GTM_AVTTRCAL pin to the GTM_AVTT supply and to a pin on the 100Ω precision external resistor. The other pin of the resistor is connected to the GTM_RREF pin. The resistor calibration circuit provides a controlled current load to the resistor connected to the GTM_RREF pin. It then senses the voltage drop across the external calibration resistor and uses that value to adjust the internal resistor calibration setting. The quality of the resistor calibration is dependent on the accuracy of the voltage measurement at the GTM_AVTTRCAL and GTM_RREF pins. To eliminate errors due to the voltage drop across the traces that lead from the resistor and to the Versal adaptive SoC pins, the trace from the GTM_AVTTRCAL pin to the resistor should have the same length and geometry as the trace that connects the other pin of the resistor to the GTM_RREF pin. Also, the maximum DC resistance of the PCB trace must be limited to less than 0.5Ω. See the suggested layout in the following figure.

Figure 57: PCB Layout for the RCAL Resistor



X25470-062121

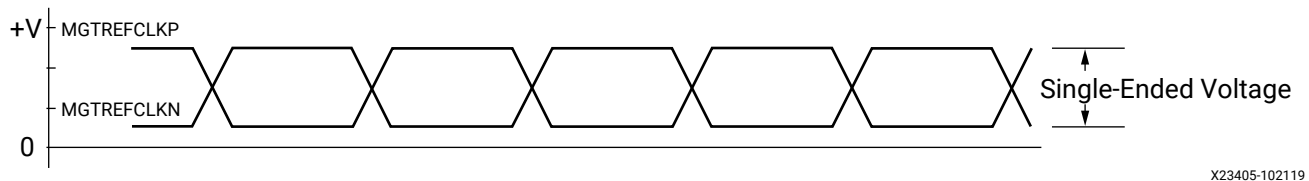
Reference Clock

This section focuses on the selection of the reference clock source or oscillator. An oscillator is characterized by:

- Frequency range
- Output voltage swing
- Jitter (deterministic, random, peak-to-peak)
- Rise and fall times
- Supply voltage and current
- Noise specification
- Duty cycle and duty-cycle tolerance
- Frequency stability

These characteristics are selection criteria when choosing an oscillator for a GTM transceiver design. [Figure 58](#) illustrates the convention for the single-ended clock input voltage swing, peak-to-peak. This figure is provided to show the contrast to the differential clock input voltage swing calculation shown in [Figure 59](#), as used in the GTM transceiver portion of the [Versal device data sheets](#).

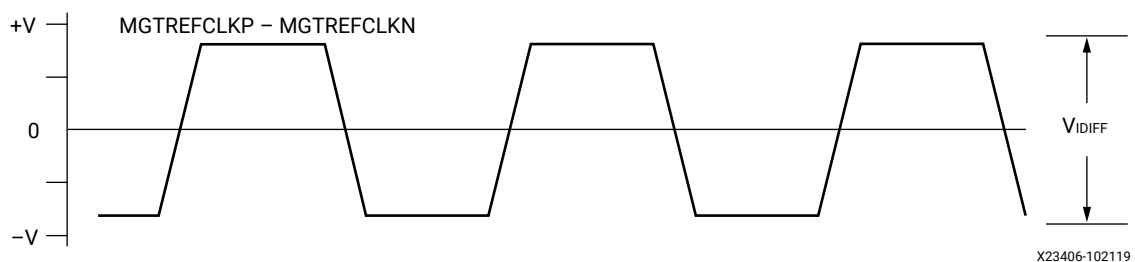
Figure 58: Single-Ended Clock Input Voltage Swing, Peak-to-Peak



X23405-102119

The following figure illustrates the differential clock input voltage swing, which is defined as $\text{GTM_REFCLKP}/\text{GTM_GTREFCLKN}$.

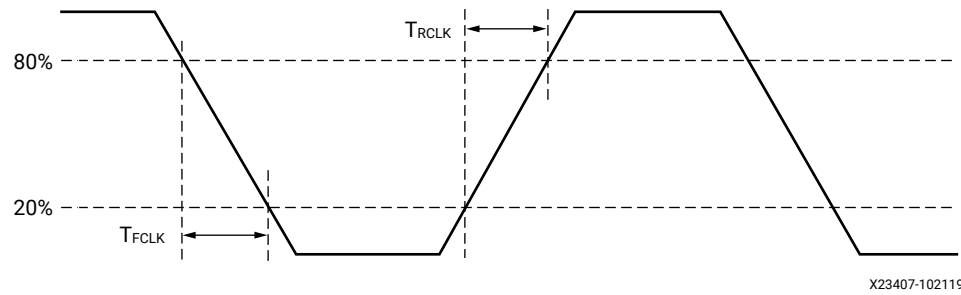
Figure 59: Differential Clock Input Voltage Swing, Peak-to-Peak



X23406-102119

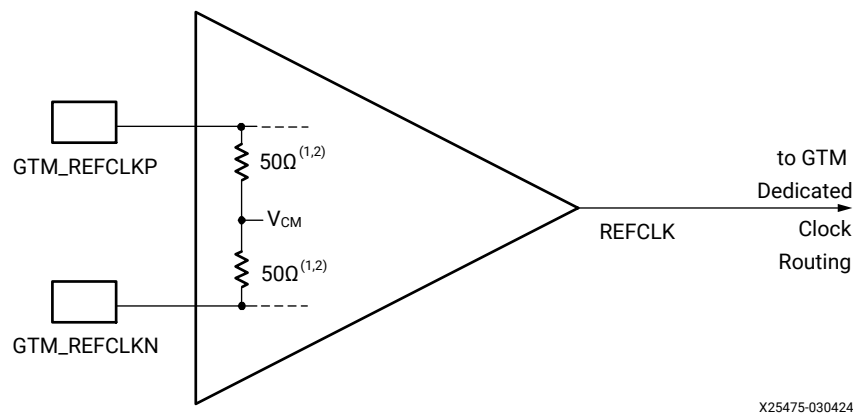
The following figure shows the rise and fall time convention of the reference clock.

Figure 60: Rise and Fall Times



The following figure illustrates the internal details of the IBUFDS. The dedicated differential reference clock input pair GTM_REFCLKP/GTM_REFCLKN is internally terminated with 100Ω differential impedance. The common mode voltage of this differential reference clock input pair is GTM_AVCC. See the [Versal device data sheets](#) for exact specifications.

Figure 61: MGTREFCLK Input Buffer Details



Notes:

1. The resistor values are nominal. See the [Versal device data sheets](#) for exact specifications.
2. AC mode is shown.

GTM Transceiver Reference Clock Checklist

These criteria must be met when choosing an oscillator for a design with GTM transceivers:

- Provide AC coupling between the oscillator output pins and the dedicated GTM transceiver Quad clock input pins.
- Ensure that the differential voltage swing of the reference clock is the range as specified in the [Versal device data sheets](#).

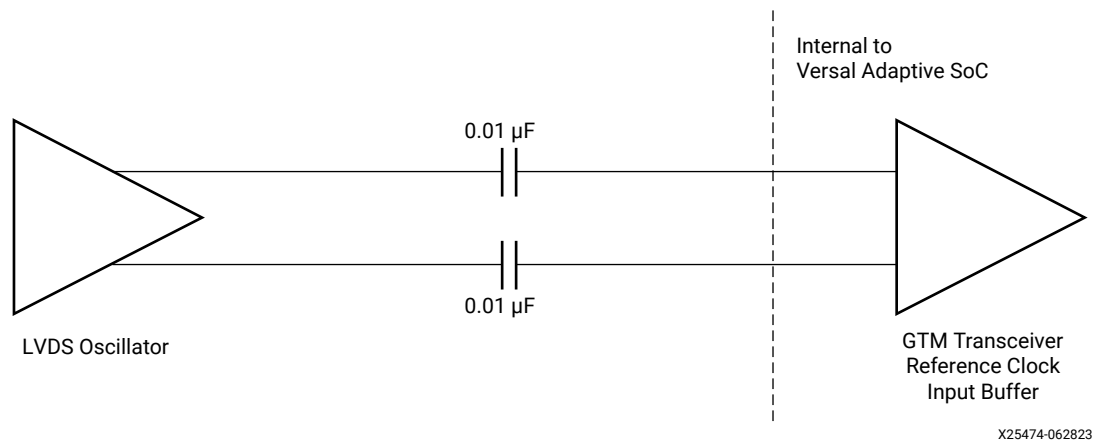
- Meet or exceed the reference clock characteristics as specified in the [Versal device data sheets](#).
- Meet or exceed the reference clock characteristics as specified in the standard for which the GTM transceiver provides physical layer support.
- Fulfill the oscillator vendor's requirement regarding power supply, board layout, and noise specification.
- Provide a dedicated point-to-point connection between the oscillator and GTM transceiver Quad clock input pins.
- Keep impedance discontinuities on the differential transmission lines to a minimum (impedance discontinuities generate jitter).

Reference Clock Interface

LVDS

The following figure shows how an LVDS oscillator is connected to a reference clock input of a GTM transceiver.

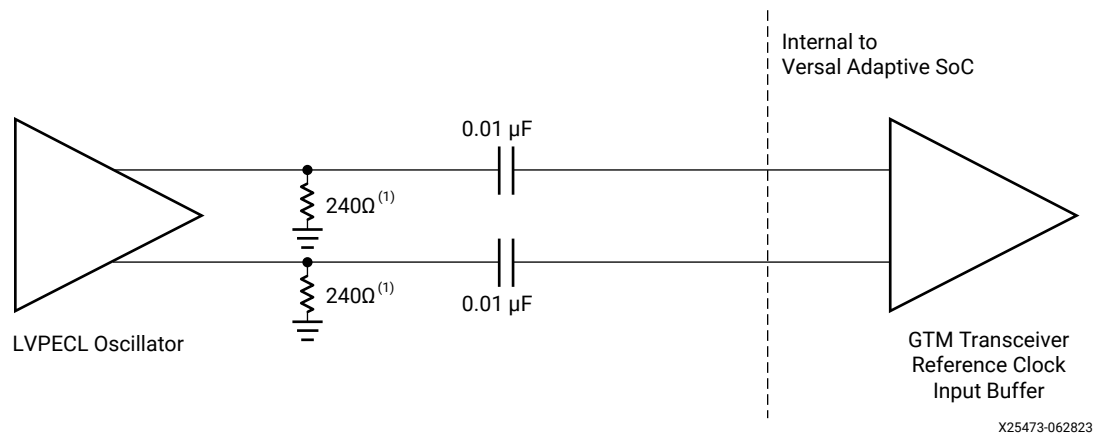
Figure 62: Interfacing an LVDS Oscillator to the GTM Transceiver Reference Clock Input



LVPECL

The following figure shows how an LVPECL oscillator is connected to a reference clock input of a GTM transceiver.

Figure 63: Interfacing an LVPECL Oscillator to the GTM Transceiver Reference Clock Input



Note: These are nominal values only. Refer to the oscillator vendor data sheet for actual bias resistor requirements.

AC Coupled Reference Clock

AC coupling of the oscillator reference clock output to the GTM transceiver Quad reference clock inputs serves multiple purposes:

- Blocking a DC current between the oscillator and the GTM transceiver Quad dedicated clock input pins (which reduces the power consumption of both parts as well)
- Common-mode voltage independence
- The AC coupling capacitor forms a high-pass filter with the on-chip termination that attenuates wander of the reference clock

To minimize noise and power consumption, external AC coupling capacitors between the sourcing oscillator and the GTM transceiver Quad dedicated reference clock input pins are required.

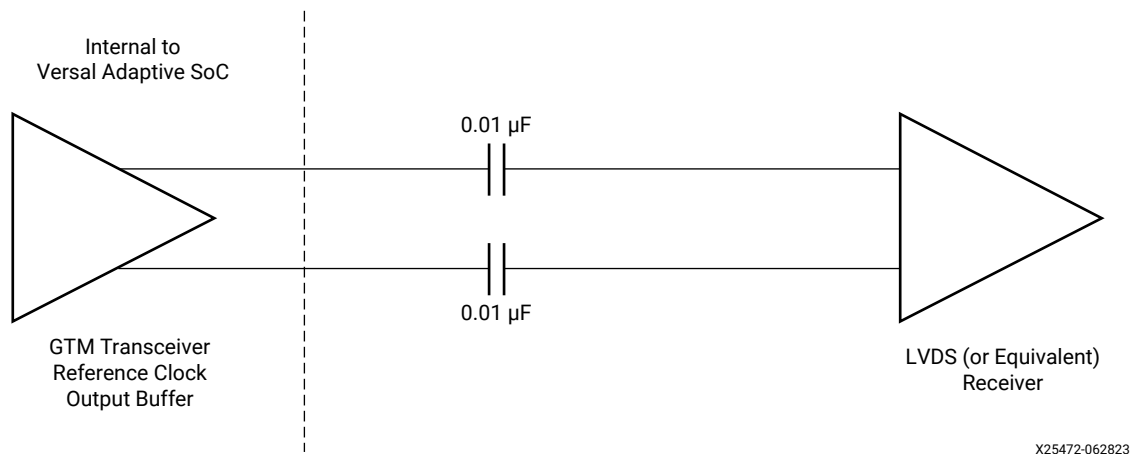
Unused Reference Clocks

If the reference clock input is not used, leave the reference clock input pins unconnected (both GTM_REFCLKP and GTM_REFCLKN).

Reference Clock Output Buffer

The reference clock pins can be configured to be output pins that drive an RX recovered clock from one of the transceivers in the Quad. Operation and configuration of this buffer is discussed in [Reference Clock Input/Output Structure](#). This output is designed to supply a signal through DC blocking capacitors on the PCB. The signal levels are comparable to those of LVDS after the DC blocking capacitors. See the [Versal device data sheets](#) for output levels.

Figure 64: Versal Architecture GTM Transceiver Reference Clock Output Connection



Reference Clock Power

The GTM transceiver reference clock input circuit is powered by GTM_AVCC. Excessive noise on this supply has a negative impact on the performance of any GTM transceiver Quad that uses the reference clock from this circuit.

Power Supply

Analog Power Supply Pins

The GTM transceiver Quad analog power supplies (GTM_AVCC, GTM_AVTT, and GTM_AVCCAUX) have planes inside the package. For some of the packages, there are multiple planes for each analog power supply. The analog power supplies for the transceivers are organized into power supply groups, PSG. Each PSG has a set of power planes. One power plane for each of the transceiver analog power supplies. If there is more than one PSG in the package, the power supply pin names have a suffix (such as _LN, _RN, _LS or _RS) that identifies which pins are associated with which PSG. If all the Quads in a PSG are not used, the associated power pins can be left unconnected or tied to GND. The rules for powering PSGs are as follows:

- Within a package PSG, if no Quads are used, the PSG can be unpowered.
- If any Quads in a PSG are used, the PSG must be powered.
- PSGs on each side (Left or Right) of the package are fully independent. Powering or not powering PSGs on one side of the package does not affect the PSGs on the other sides of the package.
- If a PSG does not have an RCAL master and it is powered, all the PSGs on that side (left or right) of the package must be powered.
- If a PSG with an RCAL master is unpowered, any PSG without an RCAL master on that side of the package must also be unpowered.
- A PSG that does not have an RCAL master can be unpowered without affecting other PSGs.

For each GTM transceiver analog power supply group there are three power supplies (GTM_AVCC, GTM_AVTT, and GTM_AVCCAUX). For example, if there are two PSGs in a package, there are a total of six power supply planes in the package for these groups, with three planes in the package for each PSG. The following table shows the power supply groups for Versal adaptive SoCs.

Note: Refer to *Versal Adaptive SoC Packaging and Pinouts Architecture Manual* ([AM013](#)) for details on the specific package die information.

Table 81: GTM Transceiver Power Supply Groups and RCAL Master by Package

Device	GTM Transceiver																																			
	109	110	111	112	115	116	117	118	121	122	123	124	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	
VM1402VD1760																																				
VP1202VA2785													LC	LC RCL	LC	LC	LN																			

Table 81: GTM Transceiver Power Supply Groups and RCAL Master by Package (cont'd)

Device	GTM Transceiver																																			
	109	110	111	112	115	116	117	118	121	122	123	124	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	
VP1502VA2785	RN	RN RCL	RN	RN									LC	LC RCL	LC	LC		LN	LN RCL	LN	LN	LN	LN													
VP1502VA3340	RN	RN RCL	RN	RN									LLC	LLC RCL	LLC	LLC	LLC	LN	LN RCL	LN	LN	LN	LN													
VP1702VA3340	RUC	RUC RCL	RUC	RUC	RN	RN RCL	RN	RN					LLC	LLC RCL	LLC	LLC	LLC	LUC	LUC RCL	LUC				LN	LN RCL	LN	LN	LN	LN							
VP1802VC4072	RC	RC RCL	RC	RC	RUC	RUC RCL	RUC	RUC	RN	RN RCL	RN	RN	LLC	LLC RCL	LLC	LLC	LLC	LC	LC RCL	LC	LC	LC	LC	LUC	LUC RCL	LUC	LUC	LUC	LUC	LN	LN RCL	LN	LN	LN	LN	LN

- Notes:**
1. In each cell, the top row is the power supply group designator. If the second row contains RCL, that Quad is an RCAL master.

Analog Power Supply Sequencing

The powering up and down of the GTM transceiver analog power supplies must follow a specific sequence specified in the Power Design Manager (PDM) tool (download at www.amd.com/power).

- Versal adaptive SoC GTM power supply power-up sequence:
 1. GTM_AVCC (power-up first)
 2. GTM_AVCCAUX
 3. GTM_AVTT
- Versal adaptive SoC power supply power-down sequence:
 1. GTM_AVTT (power-down first)
 2. GTM_AVCCAUX
 3. GTM_AVCC



IMPORTANT! GTM_AVTT must not be powered on while GTM_AVCC is powered off.

Power Supply and Filtering

The GTM transceiver Quad requires three analog power supplies: GTM_AVCC, GTM_AVCCAUX, and GTM_AVTT. For nominal voltages, see the [Versal device data sheets](#). The pins for each of these analog power supplies are tied to a plane in the package. In some packages, there are two planes (a left plane and a right plane) for each of the analog power supplies. See [Chapter 5: Board Design Guidelines](#) for a discussion of the internal power planes in the GTM transceiver packages.

Noise on the GTM transceiver analog power supplies can cause degradation in the performance of the transceivers. The most likely form of degradation is an increase in jitter at the output of the GTM transmitter and reduced jitter tolerance in the receiver. Sources of power supply noise are:

- Power supply regulator noise
- Power distribution network
- Coupling from other circuits

Each of these noise sources must be considered in the design and implementation of the GTM transceiver analog power supplies. The total peak-to-peak noise as measured at the input pin of the Versal adaptive SoC should not exceed 10 mVpp over the frequency range of 10 kHz to 80 MHz.

Power Up/Down and Reset on Multiple Lanes

The operating state of the GTM transceiver is controlled through the assertion and deassertion of the power down and reset controls (see [Reset and initialization](#) and [Power Down](#)).

When the GTM transceiver's operating state is changed by either changing the power down state or the reset state, the load current as seen by the on-board power distribution network (PDN) and the power supply regulator is also changed. When the load current changes, the power supply regulator must sense the change in the load current and compensate for this change to maintain the design supply voltage. The effect of a delay in the change in the load current can result in a temporary spike or dip in the power supply voltage. When the operating state of the GTM transceiver goes from power down to power up, the load current transient is positive and the voltage from the regulator might dip while the regulator circuit adapts to the new load conditions. Conversely, when the operating state of the GTM transceiver goes from power up to power down, the load current transient is negative and the voltage from the regulator might spike while the regulator circuit adapts to the new load current conditions.

The magnitude and duration of the voltage transient from the power supply regulator depends upon the design of the power supply regulator circuit. In some cases, the voltage might oscillate as the voltage regulator circuit converges to the design voltage setting.

In all of these cases, the important consideration is that the voltage at the input pin of the device must remain within the operating limits as specified in the [Versal device data sheets](#). Use the Power Design Manager (PDM) tool (download at www.amd.com/power) to calculate the amount of power required for the transceivers in your application.

Power Supply Regulators

Normally, the GTM transceiver analog voltage supplies have local power supply regulators that provide a final stage of voltage regulation. Preferably these regulators are placed as close as is feasible to the GTM transceiver power supply pins. Minimizing the distance between the analog voltage regulators and the GTM transceiver power supply pins reduces the opportunity for noise coupling into the supply after the regulator and for noise generated by current transients caused by load dynamics.

Linear versus Switching Regulators

The type of power supply regulator can have a significant impact on the complexity, cost, and performance of the power supply circuit. A power supply regulator must provide adequate power to the GTM transceiver with a minimum amount of noise while meeting the overall system thermal and efficiency requirements. There are two major types of power supply voltage regulators available for regulating the GTM transceiver analog voltage rails, linear regulators, and switching regulators. Each of these types of regulators has advantages and disadvantages. The optimal choice of regulator type depends on system requirements such as:

- Physical size

- Thermal budget
- Power efficiency
- Cost

Linear Regulator

A linear regulator is usually the simplest means to provide voltage regulation for the GTM transceiver analog supply rails. Inherently, a linear regulator does not inject significant noise into the regulated output voltage. In fact, some, not all, linear regulators provide noise rejection at the output from noise present on the voltage input. Another advantage of the linear regulator is that it usually requires a minimal number of external components to realize a circuit on the printed circuit board.

There are potentially two major disadvantages to linear regulators, minimum dropout voltage, and limited efficiency. Linear regulators require an input voltage that is higher than the output voltage. This minimum dropout voltage often is dependent on the load current. Even low dropout linear regulators require a minimum difference between the input voltage and the output voltage of the regulator. The system power supply design must consider the minimum dropout voltage requirements of the linear regulators.

The efficiency of a linear regulator is dependent on the voltage difference between the input and output of the linear regulator. For instance, if the input voltage of the regulator is $2.5 V_{DC}$ and the output voltage of the regulator is $1.2 V_{DC}$, the voltage difference is $1.3 V_{DC}$. Assuming that the current into the regulator is essentially equal to the current out of the regulator, the maximum efficiency of the regulator is 48%. This means that for every watt delivered to the load, the system must consume an additional watt for regulation. This power consumed by the regulator generates heat that must be dissipated by the system. Providing a means to dissipate the heat generated by the linear regulator can drive up the system cost. So even though from a simple component count and complexity cost, the linear regulator appears to have an advantage over the switching regulator, if the overall system cost is considered, including power consumption and heat dissipation, in high current applications, the linear regulator can actually be at a disadvantage.

Switching Regulator

A switching regulator can provide an efficient means to deliver a well-regulated voltage for the GTM transceiver analog power supply. Unlike the linear regulator, the switching regulator does not depend on the voltage drop between the input voltage of the regulator and the output voltage to provide regulation. Therefore the switching regulator can supply large amounts of current to the load while maintaining high power efficiency. It is not uncommon for a switching

regulator to maintain efficiencies of 95% or greater. This efficiency is not severely impacted by the voltage drop between the input of the regulator and the output. It is impacted by the load current in a much lesser degree than that of the linear regulator. Because of the efficiency of the switching regulator, the system does not need to supply as much power to the circuit, and it does not need to provide a means to dissipate power consumed by the regulator.

The disadvantages to the switching regulator are complexity of the circuit and noise generated by the regulator switching function. Switching regulator circuits are usually more complex than linear regulator circuits. This shortcoming in switching regulators has recently been addressed by several switching regulator component vendors. Normally, a switching power supply regulation circuit requires a switching transistor element, an inductor, and a capacitor. Depending on the required efficiency and load requirements, a switching regulator circuit might require external switching transistors and inductors. Besides the component count, these switching regulators require very careful placement and routing on the printed circuit board to be effective.

Switching regulators generate significant noise and therefore usually require additional filtering before the voltage is delivered to the GTM transceiver analog power supply input of the GTM transceiver. As the amplitude of the noise should be limited to less than 10 mVpp over the frequency range 10KHz to 80MHz, the power supply filter should be designed to attenuate the noise from the switching regulator to meet this requirement.

Power Supply Distribution Network Staged Decoupling

Die

The decoupling capacitance on the die filters the highest frequency noise components on the power supplies. The source of this very high frequency noise is the internal on-die circuits.

Package

The Versal architecture package has additional decoupling. Decoupling capacitors in the package provide attenuation for noise in the package power plane, thereby reducing the interaction between GTM transceiver Quads. These capacitors in the package also aid in maintaining a low-impedance, high-frequency path between the power supply, GTM_AVCC, GTM_AVCCAUX, or GTM_AVTT, and GND.

Printed Circuit Board

Because the impedance between the power planes and GND has been kept low on the die and in the package, the board design has a much more relaxed requirement for decoupling on the printed circuit board. The primary purpose of the PCB decoupling capacitors is to provide noise isolation between the transceiver power supply pins and the external noise sources. Some examples of external noise sources are:

- Power supply regulator circuits
- On-board digital switching circuits
- SelectIO™ signals from the Versal adaptive SoC

Decoupling capacitors should be provided on the PCB near the GTM transceiver power pins. These capacitors reduce the impedance of the PCB power distribution network. The reduced impedance of the PDN provides a means to attenuate noise from external sources before it can get into the device package power planes. The noise at the power pins should be less than 10 mVpp over the band from 10 kHz to 80 MHz.

Use the Power Design Manager (PDM) tool (download at www.amd.com/power) to determine the PCB capacitor recommendations. The number and usage of GTM transceivers is used by PDM to calculate the decoupling capacitors required for each of the GTM transceiver analog power supplies. The GTM transceiver Quads are organized into power supply groups in the package. See [Analog Power Supply Pins](#) for the package being used.

PCB Design Checklist

The following table is a checklist of items that can be used to design and review any GTM transceiver PCB schematic and layout.

Table 82: GTM Transceiver PCB Design Checklist

Pins	Recommendation
GTM_REFCLK0P GTM_REFCLK0N GTM_REFCLK1P GTM_REFCLK1N	<p>When configured as an input:</p> <ul style="list-style-type: none"> • Use AC coupling capacitors for connection to oscillator. • For AC coupling capacitors, see Reference Clock Interface. • Reference clock oscillator output must comply with the minimum and maximum input amplitude requirements for these input pins. See the Versal device data sheets. <p>When configured as an output:</p> <ul style="list-style-type: none"> • Use AC coupling capacitors for connection to receiving device. • For AC coupling capacitors use 0.01 µF. • For output signal characteristics, see the Versal device data sheets. • If reference pins are not used, leave the associated pin pair unconnected. However, if the IBUFDS_GTME5 is instantiated in the design but not used, the associated pin pair should be connected to GND.

Table 82: GTM Transceiver PCB Design Checklist (cont'd)

Pins	Recommendation
GTM_RXP[0/1/2/3] GTM_RXN[0/1/2/3]	<ul style="list-style-type: none"> Use AC coupling capacitors for connection to transmitter. The recommended value for AC coupling capacitor is 100 nF. Receiver data traces should be provided enough clearance to eliminate crosstalk from adjacent signals. If a receiver will never be used under any conditions, connect the associated pin pair to GND. If a receiver is not used and not connected to anything under some conditions, but might be connected to something and used under other conditions, then for the conditions when the receiver is unused, either do not instance the GTM transceiver in the Versal adaptive SoC design, or if the GTM transceiver is instanced, set RXP[1:0] to 2'b11. See RX Analog Front End for more details. For 112G long reach (LR) GTM applications, lane selection restrictions are signified in the package file for each TX/RX pin: <ul style="list-style-type: none"> If HIGH performance, lanes are suitable for 112G LR applications. If MEDIUM performance, the lane restrictions vary. Non-LR channels with a loss budget of 20 dB or less (ball to ball), operating at up to 112G, have no restrictions on lane selection. LR channels operating at up to 58G have no restrictions on lane selection. These restrictions do not apply if only TX or RX (not both) are being used within a certain channel. See <i>Versal Adaptive SoC Packaging and Pinouts Architecture Manual (AM013)</i> for package files and Answer Record 35326 for more information.
GTM_TXP[0/1/2/3] GTM_TXN[0/1/2/3]	<ul style="list-style-type: none"> The transmitter should be AC coupled to the receiver. The recommended value for the AC coupling capacitor is 100 nF. Transmitter data traces should be provided enough clearance to eliminate crosstalk from adjacent signals. If a transmitter is not used, leave the associated pin pair unconnected.
GTM_AVTTRCAL	<ul style="list-style-type: none"> Connect to GTM_AVTT and to a 100Ω resistor that is also connected to GTM_RREF. Use identical trace geometry for the connection between the resistor and this pin and for the connection from the other pin of the resistor to GTM_RREF. Also, the DC resistance of the PCB trace should be limited to less than 0.5Ω. See Termination Resistor Calibration Circuit. If an entire PSG is not used by any Quads, tie GTM_AVTTRCAL to ground.
GTM_RREF	<ul style="list-style-type: none"> Connect a 100Ω resistor that is also connected to GTM_AVTTRCAL. Use identical trace geometry for the connection between the resistor and this pin and for the connection from the other pin of the resistor to GTM_AVTTRCAL. Also, the DC resistance of the PCB trace should be limited to less than 0.5Ω. See Termination Resistor Calibration Circuit. If an entire PSG is not used by any Quads, tie MGTRREF to ground.

Table 82: GTM Transceiver PCB Design Checklist (cont'd)

Pins	Recommendation
VCCINT_GT	<ul style="list-style-type: none"> See the Versal device data sheets for nominal voltage and power supply voltage tolerances. The power supply regulator for this voltage should not be shared with non-transceiver loads. Some packages have only one VCCINT_GT package power group and others have two VCCINT_GT package power groups. Information on pin locations for each package can be found in <i>Versal Adaptive SoC Packaging and Pinouts Architecture Manual</i> (AM013). For optimal performance, power supply noise must be less than 10 mVpp over the frequency range 10 kHz to 80 MHz. If any of the GTM transceivers on the VCCINT_GT package power group is used, VCCINT_GT must be powered. If all of the Quads in a power supply group are not used, the associated power pins should be tied to GND. When there are multiple VCCINT_GT package power groups in the package, the pins associated with the power groups have a suffix added, for example, VCCINT_GT_R and VCCINT_GT_L. For power consumption, power supply sequencing and filter capacitor recommendations, refer to the Power Design Manager (PDM) tool (download at www.amd.com/power).
GTM_AVCC	<ul style="list-style-type: none"> See the Versal device data sheets for nominal voltage and power supply voltage tolerances. The power supply regulator for this voltage should not be shared with non-transceiver loads. Many packages have multiple groups of power supply connections in the package for GTM_AVCC. Information on pin locations for each package can be found in <i>Versal Adaptive SoC Packaging and Pinouts Architecture Manual</i> (AM013). For optimal performance, power supply noise must be less than 10 mVpp over the frequency range 10 kHz TO 80 MHz. If all of the Quads in a power supply group are not used, the associated power pins can be left unconnected or tied to GND. For power consumption, power supply sequencing and filter capacitor recommendations, refer to the Power Design Manager (PDM) tool (download at www.amd.com/power).
GTM_AVTT	<ul style="list-style-type: none"> See the Versal device data sheets for nominal voltage and power supply voltage tolerances. The power supply regulator for this voltage should not be shared with non-transceiver loads. Many packages have multiple groups of power supply connections in the package for GTM_AVTT. Information on pin locations for each package can be found in <i>Versal Adaptive SoC Packaging and Pinouts Architecture Manual</i> (AM013). For optimal performance, power supply noise must be less than 10 mVpp over the frequency range 10 kHz TO 80 MHz. If all of the Quads in a power supply group are not used, the associated power pins can be left unconnected or tied to GND. For power consumption, power supply sequencing and filter capacitor recommendations, refer to the Power Design Manager (PDM) tool (download at www.amd.com/power).

Table 82: GTM Transceiver PCB Design Checklist (cont'd)

Pins	Recommendation
GTM_AVCCAUX	<ul style="list-style-type: none">• See the Versal device data sheets for nominal voltage and power supply voltage tolerances.• The power supply regulator for this voltage should not be shared with non-transceiver loads.• Many packages have multiple groups of power supply connections in the package for GTM_AVCCAUX. Information on pin locations for each package can be found in <i>Versal Adaptive SoC Packaging and Pinouts Architecture Manual</i> (AM013).• For optimal performance, power supply noise must be less than 10 mVpp over the frequency range 10 KHz TO 80 MHz.• If all of the Quads in a power supply group are not used, the associated power pins can be left unconnected or tied to GND.• For power consumption, power supply sequencing and filter capacitor recommendations, refer to the Power Design Manager (PDM) tool (download at www.amd.com/power).

Additional Resources and Legal Notices

Finding Additional Documentation

Technical Information Portal

The AMD Technical Information Portal is an online tool that provides robust search and navigation for documentation using your web browser. To access the Technical Information Portal, go to <https://docs.amd.com>.

Documentation Navigator

Documentation Navigator (DocNav) is an installed tool that provides access to AMD Adaptive Computing documents, videos, and support resources, which you can filter and search to find information. To open DocNav:

- From the AMD Vivado™ IDE, select **Help** → **Documentation and Tutorials**.
- On Windows, click the **Start** button and select **Xilinx Design Tools** → **DocNav**.
- At the Linux command prompt, enter `docnav`.

Note: For more information on DocNav, refer to the *Documentation Navigator User Guide* ([UG968](#)).

Design Hubs

AMD Design Hubs provide links to documentation organized by design tasks and other topics, which you can use to learn key concepts and address frequently asked questions. To access the Design Hubs:

- In DocNav, click the **Design Hubs View** tab.
- Go to the [Design Hubs](#) web page.

Support Resources

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

References

These documents provide supplemental material useful with this guide:

1. Versal device data sheets:
 - *Versal Architecture and Product Data Sheet: Overview* ([DS950](#))
 - *Versal Prime Series Data Sheet: DC and AC Switching Characteristics* ([DS956](#))
 - *Versal AI Core Series Data Sheet: DC and AC Switching Characteristics* ([DS957](#))
 2. *Vivado Design Suite User Guide: Logic Simulation* ([UG900](#))
 3. *Versal Adaptive SoC Transceivers Wizard LogiCORE IP Product Guide* ([PG331](#))
 4. *An Attribute-Programmable PRBS Generator and Checker* ([XAPP884](#))
 5. *Versal Adaptive SoC Clocking Resources Architecture Manual* ([AM003](#))
 6. *Versal Adaptive SoC Packaging and Pinouts Architecture Manual* ([AM013](#))
 7. [High-Speed Serial I/O Made Simple](#)
-

Revision History

The following table shows the revision history for this document.

Section	Revision Summary
09/05/2024 Version 1.1	
Features	Updated NRZ and PAM4 modulation rates.
Input Mode	Replaced MGTAVCC with V_{CM} .
Reference Clock Selection and Distribution	Updated line rates.
LC Tank PLL	<ul style="list-style-type: none">• Specified line rates pending characterization.• Added note to Table 16.
Dynamic Frac-N	Added section.
Transceiver RX Component Reset	Added note.
Table 24: Recommended Receiver Resets for Common Situations	Updated PCSRESETMASK bit definitions.

Section	Revision Summary
Figure 29: Transceiver Rate Change with Changes in REFCLK Frequency without Port Change	Added figure.
Ports and Attributes	Added clarification about attributes.
Power Down and PLL Power Down	Updated LCPLLPD to be attribute controlled.
TX and RX Power Down	Added information about half density mode.
Loopback	Added note about BER degradation.
Fabric Configuration Interface	Added information about TXOUTCLK.
Usage Mode	Clarified APB3ADDR settings for reads.
Digital Monitor	Added clarification of digital monitor clock source.
Table 37: TX Fabric Port Connections for Different Data Widths	Added table.
Reading TX FIFO Latency	Added information about TXLATCLK.
TX Pattern Generator	Clarified PRBS patterns for PAM4 and added Table 43: PRBS Patterns for PAM4 .
Figure 40: TX Gray Encoder Bit Flip Sequence and Figure 53: RX Pre-Coder Bit Flip Sequence	Added figures.
Table 51: Transmitter Swing Control	<ul style="list-style-type: none"> Updated swing values. Added Coefficient Units column. Added notes 3 and 4.
Table 52: Transmitter Post-Cursor TX Pre-Emphasis Control , Table 53: Transmitter Pre-Cursor TX Pre-Emphasis Control , Table 54: Transmitter Pre-Cursor 2 TX Pre-Emphasis Control , and Table 55: Transmitter Pre-Cursor 3 TX Pre-Emphasis Control	Updated emphasis values.
Table 54: Transmitter Pre-Cursor 2 TX Pre-Emphasis Control	<ul style="list-style-type: none"> Updated emphasis values. Added note about inversion.
Table 57: TX Configurable Driver Attributes	Added table.
RX Analog Front End	Added sentence about level shifter circuit to first paragraph.
Figure 51: RX Serial and Parallel Clock Divider	Added RXRECCLKOUT to RX PROG DIV.
RX Margin Analysis	Updated sentence about implementing sample eye diagram.
Reading RX FIFO Latency	Added information on RXLATCLK.
Table 77: RX Fabric Port Connections for Different Data Widths	Added table.
Table 80: Transceiver Quad Descriptions	<ul style="list-style-type: none"> Added VCCINT_GT. Updated description of GTM_AVCC, GTM_AVTT, and GTM_AVCCAUX.
Figure 56: GTM Transceiver Power Supply Connections	Added VCCINT_GT.
Reference Clock	<ul style="list-style-type: none"> In Figure 61: MGTREFCLK Input Buffer Details, replaced 4/5 GTM_AVCC with V_{CM}. Updated second note after figure.
GTM Transceiver Reference Clock Checklist	Removed nominal range from second bullet.

Section	Revision Summary
Table 82: GTM Transceiver PCB Design Checklist	<ul style="list-style-type: none">Added 112G LR application to recommendations for GTM_RXP/N[0/1/2/3].Added VCCINT_GT.Removed voltage specifics.
04/27/2022 Version 1.0	
Initial release.	N/A

Please Read: Important Legal Notices

The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions, and typographical errors. The information contained herein is subject to change and may be rendered inaccurate for many reasons, including but not limited to product and roadmap changes, component and motherboard version changes, new model and/or product releases, product differences between differing manufacturers, software changes, BIOS flashes, firmware upgrades, or the like. Any computer system has risks of security vulnerabilities that cannot be completely prevented or mitigated. AMD assumes no obligation to update or otherwise correct or revise this information. However, AMD reserves the right to revise this information and to make changes from time to time to the content hereof without obligation of AMD to notify any person of such revisions or changes. THIS INFORMATION IS PROVIDED "AS IS." AMD MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND ASSUMES NO RESPONSIBILITY FOR ANY INACCURACIES, ERRORS, OR OMISSIONS THAT MAY APPEAR IN THIS INFORMATION. AMD SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT WILL AMD BE LIABLE TO ANY PERSON FOR ANY RELIANCE, DIRECT, INDIRECT, SPECIAL, OR OTHER CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, EVEN IF AMD IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

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

© Copyright 2022–2024 Advanced Micro Devices, Inc. AMD, the AMD Arrow logo, Versal, Vivado, and combinations thereof are trademarks of Advanced Micro Devices, Inc. AMBA, AMBA Designer, Arm, ARM1176JZ-S, CoreSight, Cortex, PrimeCell, Mali, and MPCore are trademarks of Arm Limited in the US and/or elsewhere. PCI, PCIe, and PCI Express are trademarks of PCI-SIG and used under license. Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.