IBERT for UltraScale GTM Transceivers v1.0

LogiCORE IP Product Guide

Vivado Design Suite

PG342 (v1.0) November 15, 2019

XILINX®

Table of Contents

Introduction

The Xilinx® LogiCORE™ IP Integrated Bit Error Ratio Tester (IBERT) core for UltraScale™ architecture GTM transceivers is designed for evaluating and monitoring the GTM transceivers. This core includes pattern generators and checkers that are implemented in the GTM transceivers, and access to ports and the dynamic reconfiguration port attributes of the GTM transceivers.

Communication logic is also included to allow the design to be run-time accessible through JTAG. This core can be used as a self-contained or an open design based on your configuration, and as described in this document.

Features

- Provides a communication path to the Vivado® Serial I/O analyzer feature.
- Provides a user-selectable number of Virtex® UltraScale+™ architecture GTM transceivers.
- Transceivers can be customized for the desired line rate, reference clock rate, and reference clock source.
- Requires a system clock that can be sourced from a pin or one of the enabled GTM transceivers.
- Supports PAM4 and NRZ signal modulations.
- Forward error correction (FEC) mode with PAM4 signaling is supported.

IP Facts

LogiCORE™ IP Facts Table	
Core Specifics	
Supported Device Family[1]	Virtex® UltraScale+™
Supported User Interfaces	NA
Resources	Performance and Resource Use web page
Provided with Core	
Design Files	Register Transfer Level (RTL)
Example Design	Verilog
Test Bench	Not Provided
Constraints File	Xilinx Design Constraints (XDC)
Simulation Model	Not Provided
Supported S/W Driver	Not Provided
Tested Design Flows[2]	
Design Entry	Vivado® Design Suite
Simulation	For supported simulators, see the Xilinx Design Tools: Release Notes Guide.
Synthesis	Vivado Synthesis
Support	
Release Notes and Known Issues	Master Answer Record: 72071
All Vivado IP Change Logs	Master Vivado IP Change Logs: 72775
Xilinx Support web page	

Notes:

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

Overview

Functional Description

The IBERT for UltraScale™ GTM Transceivers core provides a broad-based physical medium attachment (PMA) evaluation and demonstration platform for UltraScale architecture GTM transceivers. Parameterizable to use different GTM transceivers and clocking topologies, the IBERT for UltraScale GTM Transceivers core can also be customized to use different line rates, reference clock rates, and logic widths. Data pattern generators and checkers are included for each GTM transceiver desired, giving several different pseudo-random binary sequence (PRBS) to be sent over the channels.

In addition, the configuration and tuning of the GTM transceivers is accessible through logic that communicates to the dynamic reconfiguration port (DRP) of the GTM transceiver to change attribute settings, as well as registers that control the values on the ports. At run time, the Vivado® serial I/O analyzer communicates to the IBERT for UltraScale GTM transceivers core through JTAG, using the Xilinx® cables and proprietary logic that is part of the IBERT for UltraScale GTM transceivers core.

Feature Summary

The IBERT for UltraScale GTM transceivers core is designed for PMA evaluation and demonstration. All the major PMA features of the GTM transceiver are supported and controllable including the following:

- TX pre-emphasis and post-emphasis
- TX differential swing
- RX equalization
- Decision Feedback Equalizer (DFE)
- Phase-Locked Loop (PLL) divider settings
- Forward Error Correction (FEC) only with PAM4 signaling

- PAM4 and NRZ signaling

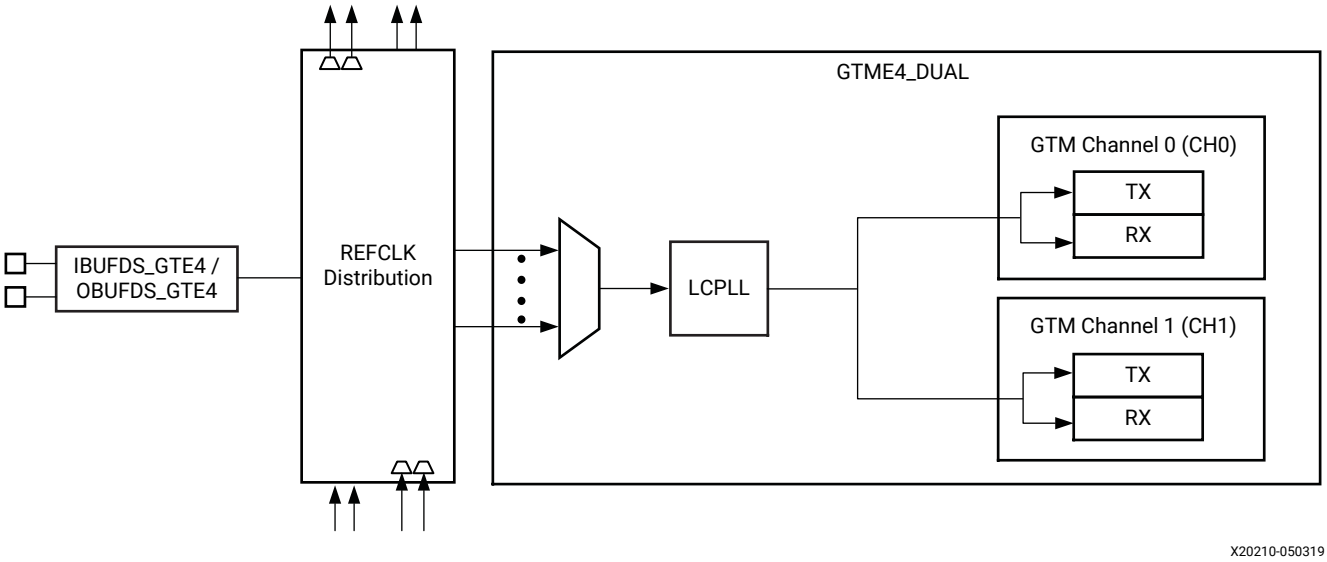
Some of the physical coding sublayer (PCS) features offered by the transceiver are outside the scope of IBERT, including:

- Clock Correction
- Channel Bonding
- 8B/10B, 64B/66B, or 64B/67B encoding
- TX or RX Buffer Bypass

PLL Configuration

The GTM_DUAL primitive contains one LCPLL and two GTM channels. Contrary to other UltraScale+ device transceivers such as the GTH and GTY transceivers, the GTM transceiver does not contain channel/common primitives. All channel ports and attributes are within the GTM_DUAL primitive. The following figure illustrates the clustering of one GTM_DUAL primitive.

Figure 1: **GTM Dual Configuration**



Pattern Generation and Checking

The IBERT for GTM enables pattern generator and checker from GTM_DUAL primitive. Contrary to other IBERT cores such as the GTH and GTY, the GTM transceiver does not contain fabric based pattern generator and checker. Each GTM transceiver enabled in the IBERT design has a pattern generator and a pattern checker. The pattern generator sends data out through the transmitter. The pattern checker accepts data through the receiver and checks it against an internally generated pattern inside GTM_DUAL primitive.

IBERT offers PRBS 7-bits, PRBS 9-bits, PRBS 13-bits, PRBS 15-bits, PRBS 23-bits, PRBS 31-bits and PRBS Disabled with PAM4 and NRZ signaling. Compared to other IBERT core such as the GTH and GTY, Clk 2x (101010...), and Clk 10x (11111111110000000000...) patterns are not supported.

With FEC enabled (only PAM4), the data pattern is non-PRBS and hence it is in FEC pattern. Errors are measured after FEC is enabled. At this stage, the errors are uncorrected. The error counter for both PRBS and FEC-Pattern increments during run-time. The IBERT core uses statistics from FEC to calculate pre-fec BER by adding the errors to the uncorrected error count.

These patterns are optimized for the logic width that was selected at run-time. The TX and RX patterns are individually selected.

Using the pattern checker logic, the incoming data is compared against a pattern that is internally generated. When the checker receives five consecutive cycles of data with no errors, the LINK signal is asserted. If the LINK signal is asserted and the checker receives five consecutive cycles with data errors, the LINK signal is de-asserted. Internal counters accumulate the number of words and errors received.

DRP and Port Access

The GTM transceiver ports and attributes can be changed. The DRP interface logic allows the run-time software to monitor and change any attribute of the GTM transceivers and the corresponding LCPLL. When applicable, readable and writable registers are also included that are connected to the various ports of the GTM transceiver. All are accessible at run-time using the Vivado serial I/O analyzer.

Applications

The IBERT for UltraScale GTM Transceivers core is designed to be used in any application that requires verification or evaluation of UltraScale+™ architecture GTM transceivers.

Unsupported Features

The following features of the standard are not supported in the IBERT for UltraScale GTM Transceivers core:

- Eye scan plot
- Fabric or FPGA based pattern generator and checker
- Receiver Output Clock

- IP integrator

Licensing and Ordering

This Xilinx® LogiCORE™ IP module is provided at no additional cost with the Xilinx Vivado® Design Suite under the terms of the Xilinx End User License.

***Note*:** To verify that you need a license, check the License column of the IP Catalog. Included means that a license is included with the Vivado® Design Suite; Purchase means that you have to purchase a license to use the core.

Information about other Xilinx® LogiCORE™ IP modules is available at the Xilinx Intellectual Property page. For information about pricing and availability of other Xilinx® LogiCORE IP modules and tools, contact your local Xilinx sales representative.

Product Specification

Performance

The IBERT for UltraScale™ GTM transceivers core can be configured to run any of the allowable line rates for the GTM transceivers.

For full details about performance and resource use, visit the Performance and Resource Use web page.

Maximum Frequencies

The core can operate at the maximum user clock frequencies for the FPGA logic width/speed grade selected. The maximum system clock rate is 100 MHz and the generated design divides any incoming system clock to adhere to this constraint.

Resource Use

For full details about performance and resource use, visit the Performance and Resource Use web page.

Port Descriptions

The IBERT for UltraScale GTM Transceivers core interfaces are shown in the following table:

Table 1: **IBERT I/O Signals**

Signal	I/O	Description
gtm_sysclkp_i	I	Clock that clocks all communication logic. This port is present only when an external clock is selected in the generator.
gtm_sysclkn_i		

Table 1: **IBERT I/O Signals** *(cont'd)*

Signal	I/O	Description
d<n>_ch0_rxp_in	O	Receive differential pairs for each of the n GTM transceivers used.
d<n>_ch0_rxn_in		
d<n>_ch1_rxp_in		
d<n>_ch1_rxn_in		
d<n>_ch0_txp_in	I	Transmit differential pairs for each of the n GTM transceivers used.
d<n>_ch0_txn_in		
d<n>_ch1_txp_in		
d<n>_ch1_txn_in		
gtm_refclkp_i[m -1:0]	I	GTM transceiver reference clocks used. The number of MGTREFCLK ports can be equal to or less than the number of transmit and receive ports because some GTM transceivers can share clock inputs.
gtm_refclkp_i[m -1:0]		

Designing with the Core

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

General Design Guidelines

GTM Transceiver Naming Style

There are two conventions for naming the GTM transceiver, based on the location in the serial transceiver tile in the device. In the XmYn naming convention, m and n indicate the X and Y coordinates of the serial transceiver location. In serial transceiver m_n naming convention, m and n indicate serial transceiver number and the associated quad.

Line Rate Support

IBERT supports a maximum of three different line rates in a single design. For each of these line rates, you can select a custom value based on your requirements. Specify the number of serial transceivers for each line rate that is programmed with these settings. The GTM transceiver in the UltraScale+ FPGA is a high performance transceiver, supporting line rates between 9.8 Gb/s and 58 Gb/s. The GTM transceiver are PAM4 and NRZ enabled transceiver that is highly configurable and tightly integrated with the programmable logic resources of the FPGA.

Serial Transceiver Location

Based on the total number of serial transceivers selected, provide the specific location of each serial transceiver that you intend to use. The region shown in the panel indicates the location of serial transceivers in the tile. This demarcation of region is based on the physical placement of serial transceivers with respect to median of BUFGs available for each device.

Use the Example Design

Each instance of the IBERT for UltraScale GTM Transceivers core created by the Vivado design tool is delivered with an example design that can be implemented in a device and then simulated. This design can be used as a starting point for your own design or can be used to sanity-check your application in the event of difficulty. See the Example Design content for information about using and customizing the example designs for the core.

Registering Signals

To simplify timing and increase system performance in a programmable device design, keep all inputs and outputs registered between the user application and the core. This means that all inputs and outputs from the user application should come from, or connect to, a flip-flop. While registering signals might not be possible for all paths, it simplifies timing analysis and makes it easier for the Xilinx® tools to place and route the design.

Recognize Timing Critical Signals

The constraints provided with the example design identify the critical signals and timing constraints that should be applied.

Make Only Allowed Modifications

You should not modify the core. Any modifications can have adverse effects on system timing and protocol compliance. Supported user configurations of the core can only be made by selecting the options in the customization IP dialog box when the core is generated.

Clocking

System Clock

The IBERT for UltraScale GTM transceivers core requires a free-running system clock for communication and other logic that is included in the core. This clock can be chosen at generation time to originate from an FPGA pin, or from a dedicated REFCLK input of one of the GTM transceivers. In order for the core to operate properly, this system clock source must remain operational and stable when the FPGA is configured with the IBERT for UltraScale GTM transceivers core design.

If the system clock is running faster than 100 MHz, it is divided down internally using an Mixed-Mode Clock Manager (MMCM) to satisfy timing constraints. The clock source selected must be stable and free-running after the FPGA is configured with the IBERT design. The system clock is used for core communication and as a reference for system measurements. Therefore, the clock source selected must remain operational and stable when using the IBERT for UltraScale GTM transceivers core.

Reference Clock

The reference clock source should be provided for all the serial transceivers selected. The drop-down list provides you with possible sources based on local clocks in the same quad and shared clocks from north/south quads.

Resets

Run-time resets are available for the IBERT counters and all transceiver resets are available. The reset controller state machine from the UltraScale+™ FPGAs GTM Tranceivers Wizard IP core is used to properly time and sequence resets. See *Virtex UltraScale+ FPGAs GTM Transceivers Wizard LogiCORE IP Product Guide* (PG315) for details.

Design Flow Steps

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

- *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994)
- *Vivado Design Suite User Guide: Designing with IP* (UG896)
- *Vivado Design Suite User Guide: Getting Started* (UG910)
- *Vivado Design Suite User Guide: Logic Simulation* (UG900)

Customizing and Generating the Core

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

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

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

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

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

The IBERT for UltraScale GTM Transceivers core can be found in /Debug & Verification/ Debug/ in the Vivado® IP catalog.

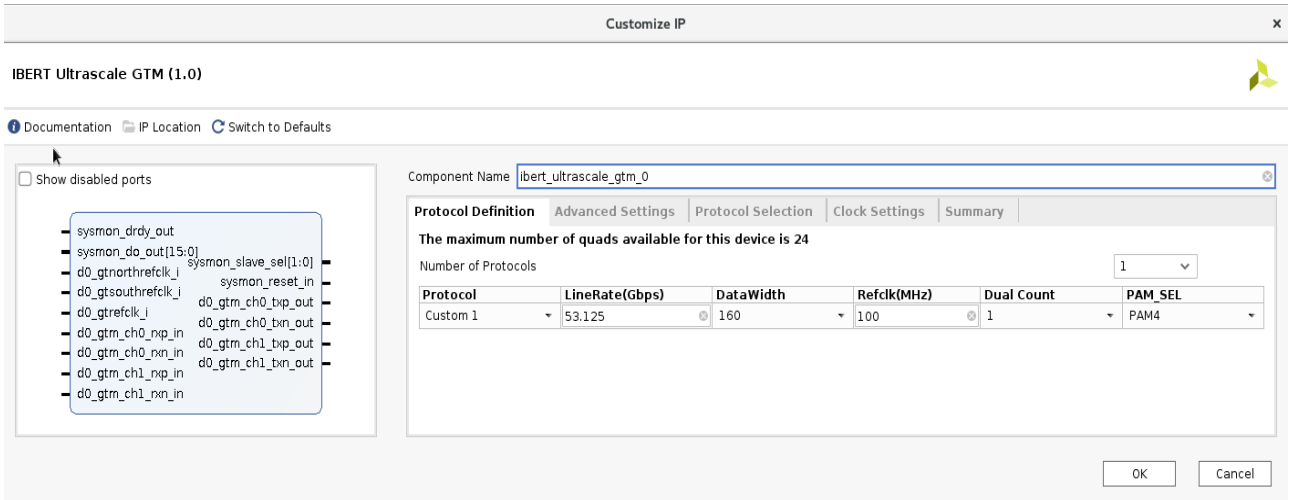
To access the core name, perform the following:

1. Open a project by selecting **File** then **Open Project** or create a new project by selecting **File** then **New Project**.

2. Open the IP catalog and navigate to Debug & Verification/Debug taxonomy.
3. Double-click **IBERT UltraScale GTM** to bring up the IBERT Customize IP dialog box.

- **Entering the Component Name:** The Component Name field can consist of any combination of alpha-numeric characters including the underscore symbol. However, the underscore symbol cannot be the first character in the component name. The following figures show the IBERT Customize IP dialog boxes with information about customizing ports.
- **Protocol Definition:** A protocol is a line rate/data width/reference clock rate combination. Up to three protocols can be defined for an IBERT for UltraScale GTM Transceivers, and any number of available Duals can be designated as any protocol defined.
  1. Choose the number of protocols desired.
  2. In the Protocol combination box, select **Custom**, type in the line rate (the rate appears in red text if outside the range allowed).
     a. Data width is fixed for PAM4 and NRZ. The value cannot be changed.
     b. Enter the `REFCLK` rate.
     c. Choose the number of Duals.
  3. The signaling mode under `PAM_SEL` is set to `PAM4` by default. To select NRZ instead, select NRZ from the combination box named `PAM_SEL`.

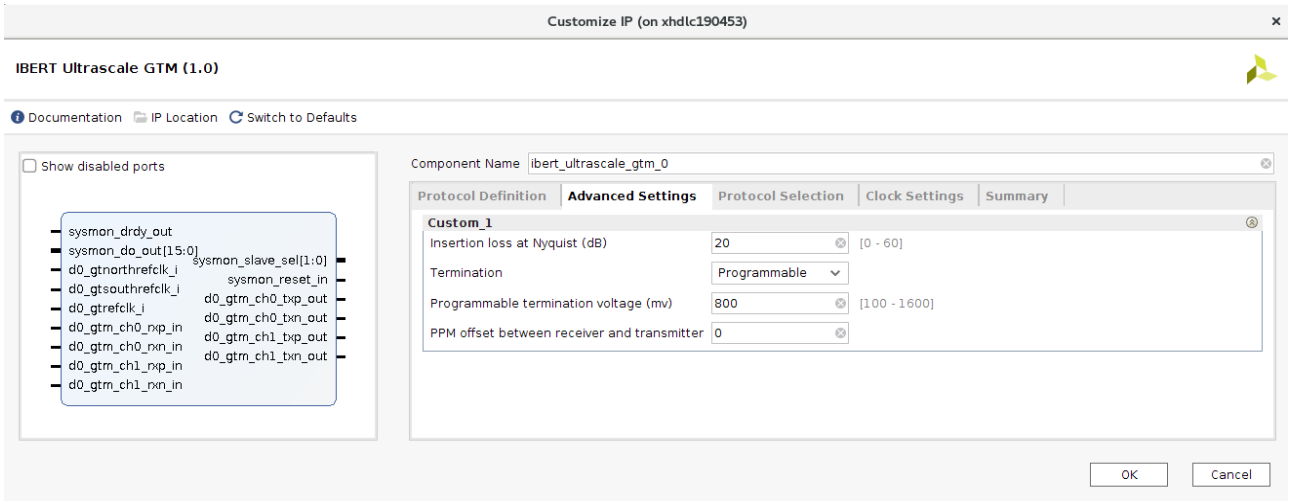
Figure 2: **Vivado Customize IP Dialog Box – Protocol Definition**



- **Advanced Settings:** You can customize advanced settings of the receiver using the customization options (Custom_1, Custom_2, and Custom_3) available in each protocol frame. The number of expandable sections available are equal to the number of protocols selected under protocol definition tab.
  - *Insertion loss at Nyquist (dB)*: Specify the insertion loss of the channel between the transmitter and receiver at the nyquist frequency in dB.

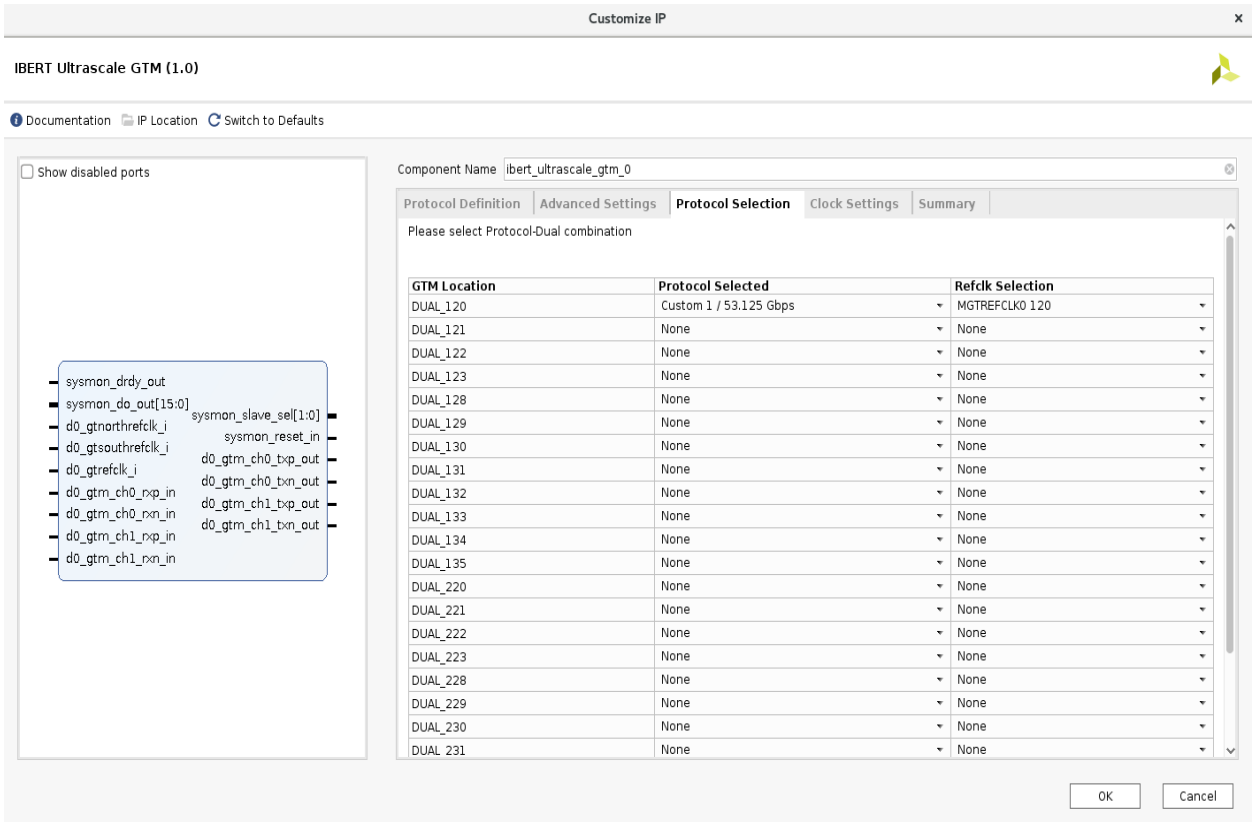
- *Termination*: Select the receiver termination voltage. Your choice of termination should depend on the protocol and its link coupling.
- *Programmable termination voltage (mV)*: When termination is set to programmable, select the termination voltage in mV.
- *PPM offset between receiver and transmitter*: Specify the offset between received data and transmitter data in PPM. For example, if your protocol specifies ±100 ppm, enter 200 as your offset value in this field. This offset affects the receiver CDR settings.

Figure 3: **Vivado Customize IP Dialog Box – Advanced Settings**



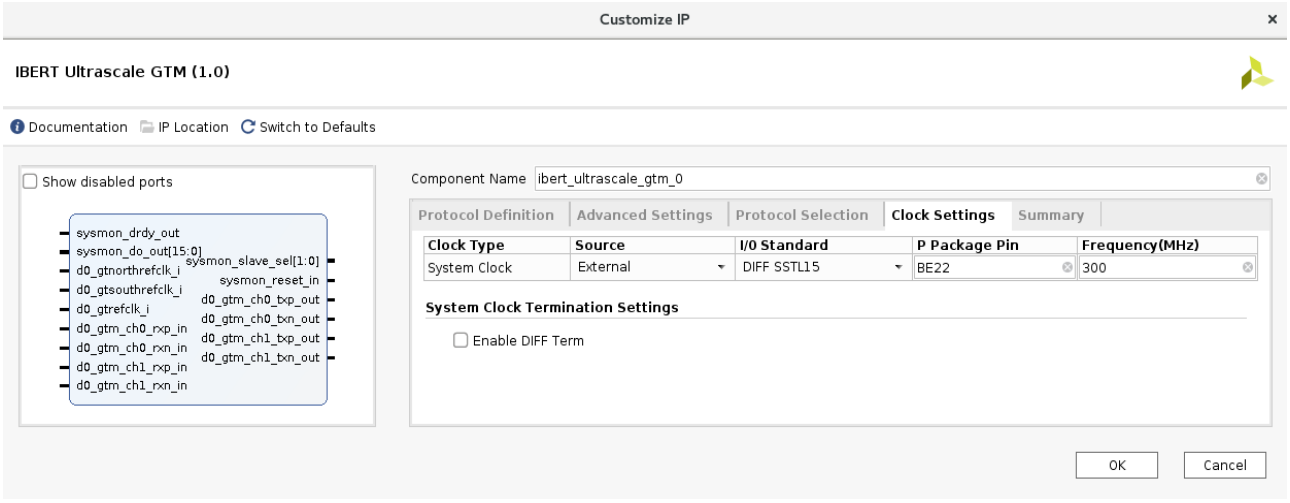
- **Protocol Selection:** In the Protocol Selection tab, the Duals available in the device/package combination are shown. To allocate a Dual to a specific protocol, select it in the Protocol Selection combination. The legal choices for the reference clock input are listed in the `REFCLK` selection combination. Each channel uses its own `TXOUTCLK` as `TXUSRCLK`. The `TXUSRCLK` source selection option is removed (as compared to IBERT 7 Series GTH).

Figure 4: **Vivado Customize IP Dialog Box – Protocol Selection**



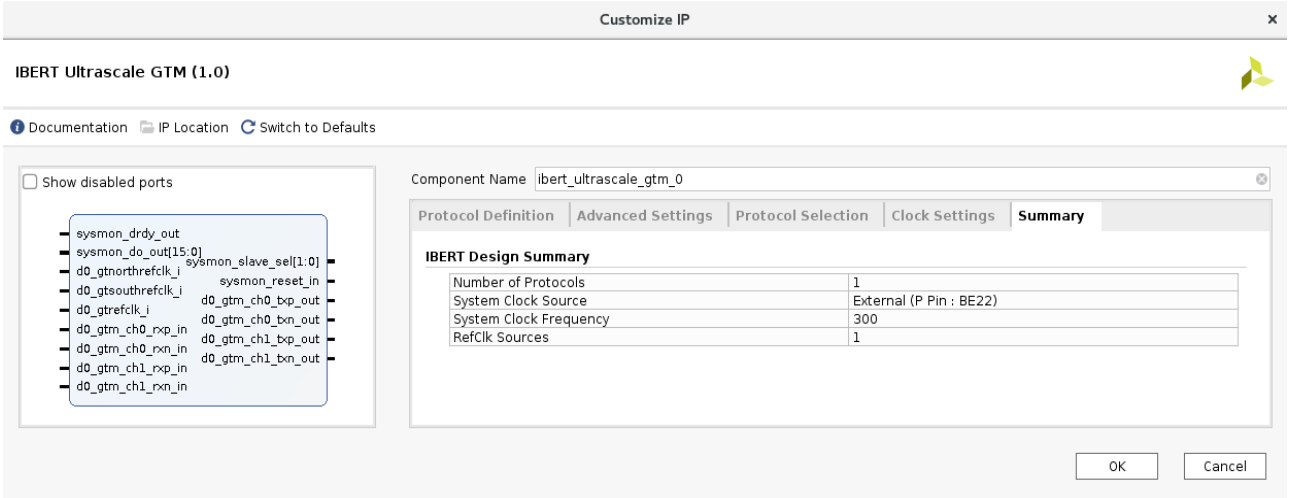
- **Clock Settings:** In the Clock Settings tab, for System Clock, specify an I/O Standard, valid pin locations, and frequency to complete the system clock settings.

Figure 5: **Vivado Customize IP Dialog Box – Clock Settings**



- **Summary:** Review the settings chosen in the summary page and if they are satisfactory, click **OK** to generate IBERT for UltraScale GTM Transceivers core.

Figure 6: **Vivado Customize IP Dialog Box – Summary**



Output Generation

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

Constraining the Core

Required Constraints

This section is not applicable for this IP core.

Device, Package, and Speed Grade Selections

This section is not applicable for this IP core.

Clock Frequencies

This section is not applicable for this IP core.

Clock Management

This section is not applicable for this IP core.

Clock Placement

This section is not applicable for this IP core.

Banking

This section is not applicable for this IP core.

Transceiver Placement

This section is not applicable for this IP core.

I/O Standard and Placement

This section is not applicable for this IP core.

Simulation

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

Synthesis and Implementation

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

Interacting with Tcl Commands

After the design is loaded into the device, a set of `hw_sio` commands interact with the IBERT for UltraScale GTM Transceivers core. For more details see the Generating an IBERT Core using the Vivado® IP Catalog chapter in the *Vivado Design Suite User Guide: Programming and Debugging* (UG908).

Example Design

This chapter contains information about the example design provided in the Vivado® Design Suite. An example design can be generated for any customization of the UltraScale+™ IBERT core. After you have customized and generated a core instance, right-click the generated core and select Open IP Example Design in the Vivado IDE for that instance. A separate Vivado project opens with the IBERT example design as the top-level module. The example design instantiates the customized core. The recommended and supported flow is to use the example design as-is, without modifications outside the Vivado IDE. The purpose of the IBERT IP example design is to:

- Provide a quick demonstration of the customized core instance operating in hardware through the use of a link status indicator based on pseudo-random binary sequence (PRBS) generators and checkers from GTM_DUAL primitive.
- Provide a system which includes reference clock buffers and example system-level constraints.
- Speed up hardware bring-up and debug through the inclusion of a pattern generator and checker.

The example design contains configurable PRBS generator and checker modules per transceiver channel that enable simple data integrity testing, and resulting link status reporting. The example design has a system management wizard IP instance for temperature sensing. The example design is also synthesizable so it can be used to check for data integrity and hardware links, either through loopback or connection to a suitable link partner. All key status signals, driving basic control signals, and hardware I/O interaction can be done using the Serial I/O Analyzer from the Vivado hardware manager after downloading the example design generated bit file.

Appendix A

Debugging

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

If the IP requires a license key, the key must be verified. The Vivado® design tools have several license checkpoints for gating licensed IP through the flow. If the license check succeeds, the IP can continue generation. Otherwise, generation halts with an error. License checkpoints are enforced by the following tools:

- Vivado Synthesis
- Vivado Implementation
- write_bitstream (Tcl command)

IMPORTANT! *IP license level is ignored at checkpoints. The test confirms a valid license exists. It does not check IP license level.*

Finding Help on Xilinx.com

To help in the design and debug process when using the core, the Xilinx Support web page contains key resources such as product documentation, release notes, answer records, information about known issues, and links for obtaining further product support. The Xilinx Community Forums are also available where members can learn, participate, share, and ask questions about Xilinx solutions.

Documentation

This product guide is the main document associated with the core. This guide, along with documentation related to all products that aid in the design process, can be found on the Xilinx Support web page or by using the Xilinx® Documentation Navigator. Download the Xilinx Documentation Navigator from the Downloads page. For more information about this tool and the features available, open the online help after installation.

Answer Records

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

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

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

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

Master Answer Record for the Core

AR 72071.

Technical Support

Xilinx provides technical support on the Xilinx Community Forums for this LogiCORE™ IP product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support if you do any of the following:

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

To ask questions, navigate to the Xilinx Community Forums.

Debug Tools

There are many tools available to address IBERT for UltraScale GTM Transceivers design issues. It is important to know which tools are useful for debugging various situations.

Vivado Design Suite Debug Feature

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

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

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

See the *Vivado Design Suite User Guide: Programming and Debugging* (UG908).

Appendix B

Additional Resources and Legal Notices

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see Xilinx Support.

Documentation Navigator and Design Hubs

Xilinx® Documentation Navigator (DocNav) provides access to Xilinx documents, videos, and support resources, which you can filter and search to find information. To open DocNav:

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

Xilinx 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.
- On the Xilinx website, see the Design Hubs page.

Note: For more information on DocNav, see the Documentation Navigator page on the Xilinx website.

References

These documents provide supplemental material useful with this guide:

1. *Virtex UltraScale+ FPGAs GTM Transceivers Wizard LogiCORE IP Product Guide* (PG315)
2. *Vivado Design Suite User Guide: Designing with IP* (UG896)
3. *Virtex UltraScale+ FPGAs GTM Transceivers User Guide* (UG581)
4. *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994)
5. *Vivado Design Suite User Guide: Getting Started* (UG910)
6. *Vivado Design Suite User Guide: Logic Simulation* (UG900)

Revision History

The following table shows the revision history for this document.

Section	Revision Summary
11/15/2019 Version 1.0	
Customizing and Generating the Core	Updated figures.
Chapter 6: Example Design	Clarifications added.
05/22/2019 Version 1.0	
Initial release.	N/A

Please Read: Important Legal Notices

www.xilinx.com/legal.htm#tos; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx's Terms of Sale which can be viewed at https://www.xilinx.com/legal.htm#tos.

AUTOMOTIVE APPLICATIONS DISCLAIMER

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

Copyright